Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА

Ратушняка Максима Віталійовича

на здобуття ступеня вищої освіти магістра

Метод адаптивного управління ресурсами захисту інформаційної системи в умовах динамічних загроз

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека та захист інформації

Освітня програма Кібербезпека та захист інформації

Шифр КРМКБЗІ.240196.24.01.11 ПЗ

Виконала студент 2 курсу група КБЗІм-24-1 Максим РАТУШНЯК

Керівник канд. техн. наук, доцент Ігор МУЛЯР

Нормоконтролер PhD, старший викладач Наталія ПЕТЛЯК

До захисту допускаю:

Завідувач кафедри кібербезпеки Юрій КЛЬОЦ

16 12 2025 р.

Хмельницький 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Кібербезпеки

Рівень вищої освіти Магістр

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека та захист інформації

Освітня програма Кібербезпека та захист інформації

ЗАТВЕРДЖУЮ

Завідувач кафедри кібербезпеки

_____ Юрій КЛЬОЦ

1 09 2025 р.

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ

Ратушняку Максиму Віталійовичу

1 Тема роботи Метод адаптивного управління ресурсами захисту інформаційної системи в умовах динамічних загроз

Керівник роботи канд.техн.наук, доцент Ігор МУЛЯР

Затверджено наказом ректора університету від 25 08 2025 № 65

2 Строк подання студентом кваліфікаційної роботи на кафедру 1.12.2025р.

3 Вихідні дані до роботи Проаналізувати динамічні кіберзагрози та виявити недоліки існуючих статичних систем захисту. Обгрунтувати доцільність використання апарату теорії ігор для моделювання конфліктів у кіберпросторі. Розробити математичну модель кіберпротистояння на основі динамічних баєсівських ігор з урахуванням асиметрії інформації. Розробити метод адаптивного управління ресурсами захисту, що використовує алгоритми навчання з підкрі для вибору оптимальної стратегії в реальному часі. Виконати програмну реалізацію симулятора кібератак. Провести експериментальне дослідження ефективності розробленого методу за різними сценаріями, здійснити порівняльний аналіз із традиційними підходами та оцінити економічну доцільність впровадження.

4 Зміст пояснювальної записки (перелік питань, які потрібно розробити) Вступ. Аналіз сучасних динамічних кіберзагроз та недоліків існуючих статичних систем захисту. Обгрунтування вибору математичного апарату теорії ігор. Розробка математичної моделі захисту та методу адаптивного управління ресурсами з використанням навчання з підкріпленням. Експериментальне дослідження ефективності методу та порівняльний аналіз стратегій захисту. Висновки.

5 Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

–

6 Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7 Дата видачі завдання 1 09 2025 р.

КАЛЕНДАРНИЙ ПЛАН

Назва етапів (розділів) кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
Ґрунтовне ознайомлення та дослідження предметної галузі	12.09.2025	Виконано
Визначення змісту, структури кваліфікаційної роботи	15.09.2025	Виконано
Підготовка першого розділу кваліфікаційної роботи	07.10.2025	Виконано
Підготовка другого розділу кваліфікаційної роботи	18.10.2025	Виконано
Підготовка третього розділу кваліфікаційної роботи	23.10.2025	Виконано
Підготовка статті/тези за темою кваліфікаційної роботи	12.11.2025	Виконано
Підготовка висновків до кваліфікаційної роботи	18.11.2025	Виконано
Підготовка та оформлення ілюстративного матеріалу	21.11.2025	Виконано
Оформлення кваліфікаційної роботи	24.11.2025	Виконано
Попередній захист кваліфікаційної роботи	27.11.2025	Виконано
Захист кваліфікаційної роботи на засіданні ЕК	17.12.2025	Виконано

Студент _____ Максим РАТУШНЯК

Керівник кваліфікаційної роботи _____ Ігор МУЛЯР

АНОТАЦІЯ

Тема кваліфікаційної роботи: Метод адаптивного управління ресурсами захисту інформаційної системи в умовах динамічних загроз.

Автор роботи: Ратушняк Максим Віталійович

Керівник: канд.техн.наук Муляр Ігор Володимирович

Загальний обсяг: 108 сторінок, 22 рисунки, 14 таблиць, 1 додаток, 81 посилання.

Ключові слова: адаптивне управління, кіберпротистояння, динамічні баєсівські ігри, навчання з підкріпленням (Q-learning), ресурси захисту, APT-загрози, імітаційне моделювання.

Метою роботи є підвищення ефективності захисту інформаційної системи шляхом розробки методу адаптивного управління ресурсами захисту, що забезпечує мінімізацію сумарних збитків в умовах динамічних загроз.

У магістерській роботі розроблено математичну модель кіберпротистояння на основі апарату динамічних баєсівських ігор, яка враховує асиметрію інформації щодо типу зловмисника. Створено метод адаптивного управління, що поєднує баєсівську ідентифікацію загрози з алгоритмом навчання з підкріпленням (Q-learning) для вибору оптимальної стратегії розподілу ресурсів у реальному часі. Виконано програмну реалізацію методу у вигляді симулятора та проведено експериментальне дослідження. Експерименти підтвердили, що розроблений адаптивний метод забезпечує успішне відбиття цільових атак та знижує сумарні очікувані збитки у декілька разів.

1.12.2025 _____

ANNOTATION

Topic of the Qualification Paper: Method of adaptive management of information system protection resources in conditions of dynamic threats.

Author: Ratushnyak Maksym

Supervisor: Candidate of Technical Sciences, Associate Professor Muliyar Ihor

Total Volume: 108 pages, 22 figures, 14 tables, 1 appendix, 81 references.

Keywords: adaptive control, cyber confrontation, dynamic bayesian games, reinforcement learning (Q-learning), protection resources, apt threats, simulation modeling.

To increase the efficiency of information system protection by developing a method for adaptive control of protection resources that ensures the minimization of total losses under conditions of dynamic threats.

In the master's thesis, a mathematical model of cyber confrontation was developed based on the framework of dynamic bayesian games, which takes into account the asymmetry of information regarding the type of attacker. A method of adaptive control was created that combines Bayesian threat identification with a reinforcement learning algorithm (Q-learning) to select the optimal resource allocation strategy in real-time. The method was implemented in software in the form of a simulator, and an experimental study was conducted. The experiments confirmed that the developed adaptive method ensures the successful repulsion of targeted attacks and reduces the total expected losses several times over.

1.12.2025 _____

ЗМІСТ

ВСТУП

Актуальність дослідження. В умовах стрімкого розвитку цифрових технологій та глобалізації інформаційного простору питання кібербезпеки набувають критичного значення для національної безпеки України. Сучасні інформаційні системи (ІС), що забезпечують функціонування критичної інфраструктури, державних установ та приватного бізнесу, стають об’єктами постійних та все більш витончених кібератак.

Особливістю сучасного ландшафту загроз є перехід від статичних, шаблонних атак до динамічних та адаптивних кампаній. Зловмисники, зокрема групи APT, активно використовують технології штучного інтелекту, поліморфне шкідливе програмне забезпечення та вразливості нульового дня. Характерною рисою таких загроз є здатність змінювати тактику, техніки та процедури TTPs безпосередньо в процесі атаки, реагуючи на дії системи захисту та адаптуючись до середовища жертви.

Водночас аналіз існуючих підходів до побудови систем захисту інформації демонструє суттєве відставання засобів протидії від наступальних можливостей зловмисників. Більшість сучасних систем управління інформаційною безпекою (ISMS) базуються на статичних моделях захисту. Вони оперують регламентованими правилами, фіксованими налаштуваннями засобів безпеки та періодичними перевірками, які проводяться за розкладом. Такі системи ефективні проти відомих, масових загроз, однак виявляються безпорадними перед цілеспрямованими динамічними атаками.

Критичною проблемою залишається неефективне управління ресурсами захисту. В умовах обмеженості обчислювальних потужностей, пропускної здатності мережі та, що найважливіше, часу і уваги кваліфікованих аналітиків SOC (Security Operations Center), намагання захистити всі активи з однаково високим пріоритетом призводить до розмивання захисного потенціалу. Статичний розподіл ресурсів створює ситуацію, коли критично важливі сегменти мережі можуть залишатися недостатньо захищеними в момент атаки, тоді як ресурси витрачаються на моніторинг менш важливих вузлів.

В умовах повномасштабної кібервійни, яка ведеться проти України, ця проблема набуває особливої гостроти. Асиметрія протистояння, де атакуючому достатньо знайти одну вразливість, а захисник змушений перекривати всі вектори атак, вимагає переходу до нових парадигм захисту. Простого нарощування кількості засобів безпеки вже недостатньо – необхідна інтелектуалізація процесів управління ними.

Вирішенням цього протиріччя є впровадження адаптивних механізмів управління, здатних в режимі реального часу оцінювати поточний рівень загрози та динамічно перерозподіляти ресурси захисту. Використання математичного апарату теорії ігор та методів машинного навчання дозволяє створити системи, що не лише реагують на інциденти постфактум, а й прогнозують дії супротивника, діючи на випередження.

Таким чином, розробка методу адаптивного управління ресурсами захисту, який забезпечує оптимальний баланс між надійністю системи та витратами на її функціонування в умовах динамічних загроз, є актуальним науково-прикладним завданням, що має важливе значення для підвищення кіберстійкості вітчизняних ІС.

Мета дослідження полягає у підвищенні ефективності захисту інформаційної системи в умовах динамічних загроз шляхом розробки методу адаптивного управління ресурсами захисту, що забезпечує мінімізацію сумарних збитків та раціональне використання засобів безпеки.

Завдання дослідження:

– проаналізувати сучасний стан проблеми захисту ІС від динамічних загроз, виявити недоліки існуючих статичних підходів до управління ресурсами та обґрунтувати необхідність застосування адаптивних методів;

– розробити удосконалену математичну модель кіберпротистояння на основі апарату динамічних ігор з неповною інформацією, яка, на відміну від класичних графів атак, враховуватиме стохастичний характер переходів системи, вартість захисних дій та асиметрію інформації про тип зловмисника;

– розробити метод адаптивного управління ресурсами захисту, що поєднує процедури баєсівської ідентифікації загрози та алгоритм навчання з

підкріпленням Q-learning для автоматичного вибору оптимальної стратегії протидії в реальному часі;

– створити алгоритмічне та програмне забезпечення (симулятор) для моделювання взаємодії «захисник – нападник» та реалізації механізмів динамічного оновлення знань про загрозу;

– провести експериментальне дослідження ефективності розробленого методу шляхом імітаційного моделювання, оцінити ймовірність успішного відбиття атак та довести економічну доцільність (зниження сумарних збитків) порівняно з традиційними стратегіями.

Об’єктом дослідження є процес функціонування системи захисту інформаційної системи в умовах впливу динамічних кіберзагроз.

Предметом дослідження є методи, моделі та алгоритми адаптивного управління ресурсами захисту ІС.

Методи дослідження. Для вирішення поставлених завдань у роботі використано комплексний методологічний підхід:

– методи системного аналізу для дослідження предметної області, класифікації загроз та формування вимог до системи захисту;

– теорія ігор для математичного моделювання конфліктної взаємодії між системою захисту та зловмисником;

– теорія ймовірностей та математична статистика для опису стохастичних процесів переходу станів системи та обробки невизначеності;

– методи машинного навчання для синтезу оптимальної стратегії управління ресурсами;

– методи імітаційного моделювання для програмної реалізації симулятора та експериментальної перевірки отриманих результатів.

Наукова новизна одержаних результатів полягає у наступному:

– удосконалено метод управління ресурсами захисту інформаційної системи, який, на відміну від відомих підходів, базується на динамічній адаптації стратегії захисту до змін поведінки зловмисника, що дозволяє підвищити ефективність використання ресурсів;

– отримало подальший розвиток математичне моделювання процесів кіберзахисту шляхом застосування апарату динамічних баєсівських ігор (ДБІ) для формалізації конфлікту в умовах асиметрії інформації щодо типу та можливостей зловмисника;

– застосовано комбінацію алгоритмів Q-learning та баєсівського оновлення переконань для вирішення задачі оперативного перерозподілу ресурсів безпеки в контурі адаптивного управління.

Практична цінність одержаних результатів полягає у розробці алгоритмічного та програмного забезпечення, яке дозволяє моделювати сценарії кібератак та оптимізувати стратегії захисту.

Основні результати роботи можуть бути використані:

– при проектуванні підсистем прийняття рішень у сучасних SOC;

– для модернізації існуючих політик безпеки корпоративних мереж з метою підвищення їх адаптивності;

– як навчальний інструмент для тренування кіберфахівців протидії APT-загрозам.

Запропонований метод дозволяє знизити сумарні очікувані збитки від інцидентів безпеки та підвищити ймовірність відбиття цільових атак на критичну інфраструктуру.

Теоретичні та практичні результати, здобуті в ході дослідження, були представлені й обговорені на міжнародних і всеукраїнських наукових конференціях. За матеріалами кваліфікаційної роботи опубліковано тези та подано наукову статтю.

1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ, ТЕОРЕТИЧНЕ ОБҐРУНТУННЯ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Характеристика динамічних загроз та виклики для існуючих систем захисту

Традиційно, системи кібербезпеки будувалися для протидії статичним загрозам – відомим шкідливим програмам або атакам, що мають чіткі, незмінні ідентифікатори. Підхід є реактивним, загроза з'являється, її аналізують, створюють "цифровий відбиток", і лише потім системи захисту вчиться її блокувати [1, 2]. Однак сучасний ландшафт загроз демонструє перехід до динамічних загроз [3].

Динамічні загрози це клас кібератак, що характеризуються здатністю до адаптації, багатоетапності та цілеспрямованості, які активно модифікують свою TTPs з метою уникнення виявлення та подолання захисних механізмів [4]. На відміну від статичної атаки, яка є однією дією (наприклад, запуск virus.exe), динамічна загроза це кампанія або процес, що розвивається в часі.

Ключові відмінності представлені в таблиці 1.1.

Таблиця 1.1 – Порівняння статичних та динамічних загроз

Характеристика	Статична загроза	Динамічна загроза
Природа	Продукт (відомий файл, скрипт)	Процес/кампанія
Виявлення	Сигнатурний аналіз (порівняння з базою відомих загроз)	Поведінковий аналіз (пошук аномалій та відхилень)
Тактика	Фіксована, передбачувана	Адаптивна, реагує на середовище та захисні дії
Стійкість	Зазвичай відсутня (мета – миттєве зараження)	Ключова риса (мета – закріпитися та залишатися непоміченим)
Приклад	Масовий email-вірус (напр., Melissa), класичний ransomware	APT-кампанія, поліморфне ПЗ, атаки "нульового дня"

Розглянемо декілька репрезентативних прикладів, що описують природу цих загроз:

– Advanced Persistent Threats;
– поліморфне та метаморфне шкідливе ПЗ;
– адаптивні ботнети;
– змагальний ШІ.

Advanced Persistent Threats (APTs) – просунуті постійні загрози являють собою тривалі, цілеспрямовані кібератаки, що проводяться висококваліфікованими групами, часто з державним фінансуванням [5, 6]. Їхня визначальна мета – не миттєвий саботаж, а отримання постійного, прихованого доступу до інфраструктури для шпигунства, викрадення даних або підготовки до майбутніх масштабних операцій. Динаміка цих загроз полягає в тому, що APT-групи постійно адаптують свої інструменти під конкретну ціль, активно використовують вразливості "нульового дня" та застосовують техніки "життя за рахунок землі" (Living off the Land, LotL) [7]. Це означає, що для атаки вони використовують легітимні системні інструменти, як-от PowerShell чи WMI, що дозволяє їм залишатися непомітними для традиційних засобів захисту.

Поліморфне та метаморфне шкідливе ПЗ – шкідливі програми, розроблені спеціально для автоматичної та безперервної зміни свого коду з метою уникнення виявлення, заснованого на сигнатурах [8, 9]. Динаміка між двома типами має відмінності: Поліморфне ПЗ досягає приховування, шифруючи своє тіло і змінюючи дешифраторну функцію при кожній новій копії, тоді як основний код залишається незмінним. Метаморфне ПЗ йде на крок далі, повністю переписуючи або трансформуючи свій виконуваний код через механізми переписування, зміни послідовності інструкцій або додавання "сміттєвого" коду, при цьому зберігаючи початкову шкідливу логіку, що робить його набагато складнішим для аналізу та ідентифікації [10].

Адаптивні ботнети – мережі скомпрометованих пристроїв (ботів), що управляються з єдиного командно-контрольного центру (C&C). Їхня головна динамічна характеристика полягає у здатності швидко та гнучко змінювати свою

інфраструктуру та тактику [11]. Вони можуть динамічно змінювати C&C сервери, щоб уникнути блокування, використовувати P2P-комунікації (від точки до точки) для децентралізації управління та постійно перемикатися між різними типами шкідливої активності, як-от розподілені атаки на відмову в обслуговуванні (DDoS), розсилання спаму або незаконний майнінг криптовалют.

Змагальний ШІ (Adversarial AI) є новітнім і швидкозростаючим класом динамічних загроз [12]. У цьому випадку методи машинного навчання використовуються не лише для захисту, а й для автоматизації атаки, а також для обходу захисних систем, які самі використовують ШІ [13]. Сюди відноситься генерація унікального та високо персоналізованого шкідливого контенту (наприклад, передовий фішинг), або створення "змагальних прикладів" – невеликих, але цілеспрямованих модифікацій вхідних даних, що змушують модель машинного навчання (наприклад, антивірус або систему розпізнавання зображень) прийняти невірне рішення (наприклад, класифікувати шкідливий файл як безпечний).

Проблематика протидії, чому традиційні підходи неефективні. Існуючі парадигми захисту, розроблені для статичних загроз, виявляються недостатньо ефективними.

Недоліки сигнатурного аналізу. Класичні антивіруси та системи виявлення/запобігання вторгнень (IDS/IPS) засновані на пошуку відомих бінарних сигнатур [14, 15].

Однак, як було зазначено, ці підходи є абсолютно марними проти динамічних загроз, які не мають постійної сигнатури, як-от поліморфне шкідливе ПЗ або спеціалізовані інструменти APT-груп [16, 17]. Нездатність розпізнати швидко мінливий або новий код залишає критичні прогалини в захисті.

Обмеженість "периметрового захисту". Брандмауери (Firewalls) та інші засоби захисту периметра втрачають свою ефективність. Динамічні загрози часто використовують дозволені протоколи (наприклад, HTTP/S) для прихованої комунікації та, що важливіше, діють "зсередини" мережі вже після успішної компрометації робочої станції [18]. Оборонні стіни, збудовані лише на зовнішньому

кордоні, не можуть протистояти зловмиснику, який уже перебуває всередині системи.

Недосконалість базового поведінкового аналізу. Навіть сучасні системи аналізу поведінки користувачів та сутностей (UBA/UEBA) можуть бути обмануті [19]. Це відбувається, якщо атака розвивається повільно та приховано ("low and slow") або якщо зловмисник використовує легітимні системні інструменти (техніка LotL – Living off the Land).

В таких випадках поведінка атакуючого стає майже невідмінною від дій законного системного адміністратора або звичайного користувача, що дозволяє загрозі тривалий час залишатися невиявленою.

В таблиці 1.2 описано проблему розподілу ресурсів захисту.

Таблиця 1.2 – Проблема розподілу ресурсів захисту

Тип ресурсу захисту	Проблема при "максимальному" навантаженні
Обчислювальний (CPU/RAM)	Падіння продуктивності серверів та робочих станцій.
Мережевий (Пропускна здатність)	Затримки в мережі, втрата пакетів, "відмова сервісу" для легітимних користувачів.
Людський (Час аналітиків SOC)	"Втома від сповіщень" (Alert Fatigue), пропуск критичних інцидентів через "шум".
Програмний ("Пісочниці", Аналіз)	Неможливо аналізувати 100% файлів та процесів у режимі реального часу.

Протидія динамічним загрозам стикається з двома фундаментальними проблемами, які безпосередньо обґрунтовують і ведуть до мети нашої роботи.

Асиметрія кіберзахисту є фундаментальним принципом, який лежить в основі всіх оборонних стратегій. Згідно з цим принципом, атакуючому достатньо знайти лише одну вразливість або одну невелику точку входу, щоб скомпрометувати систему [20]. На противагу цьому, захисник повинен забезпечити захист усієї поверхні атаки – включаючи кожен вузол, кожен сервіс та кожен канал зв'язку. Ця нерівність зусиль робить завдання захисту критично складним [21].

Проблема обмеженості та розподілу ресурсів захисту є ключовим викликом, що випливає з асиметрії [22, 23]. Забезпечення максимального, глибокого рівня захисту для всіх активів інформаційної системи одночасно є як технічно, так і економічно неможливим. Наприклад, постійний глибокий аналіз усього мережевого трафіку створює неприпустиме навантаження на обладнання та вносить значні затримки (latency) [24].

Аналогічно, безперервне сканування всіх вузлів на аномалії вимагає величезних обчислювальних ресурсів (CPU/RAM), що призводить до зниження продуктивності критичних бізнес-додатків. Крім того, людські ресурси також обмежені. Вони стикаються з когнітивним перевантаженням через величезну кількість сповіщень (alert fatigue), що не дозволяє їм адекватно та своєчасно реагувати на кожен справжній інцидент.

Існуючі системи захисту здебільшого функціонують за статичною моделлю розподілу ресурсів (наприклад, сканування за розкладом, фіксовані налаштування IDS). Вони не здатні гнучко реагувати на зміну TTPs зловмисника або на зміну пріоритетів захисту активів [25].

1.2 Особливості архітектури сучасних ІС та їх вплив на складність захисту

Еволюція інформаційних технологій докорінно змінила ландшафт об'єкта захисту. Якщо класичні методи кібербезпеки розроблялися для захисту статичних, централізованих серверних кімнат, то сучасна інформаційна система (ІС) являє собою високодинамічне, розподілене та гетерогенне середовище [26, 27]. Ці архітектурні зрушення створюють нові виклики для систем управління ресурсами захисту.

Фундаментальною зміною в архітектурі ІС є докорінний перехід від застарілої моделі "фортеці та рову" (castle-and-moat), яка передбачала чіткий поділ на "довірену" внутрішню мережу та "ворожий" зовнішній світ, до сучасної

моделі без периметра. Ця трансформація обумовлена низкою ключових факторів, які розмили традиційні кордони корпоративної мережі [28].

Одним із головних каталізаторів є хмарні обчислення (Cloud Computing). Перехід до гнучких гібридних та мультихмарних середовищ (зокрема, AWS, Azure, Google Cloud) означає, що критичні дані та обчислювальні потужності вже не знаходяться під прямим фізичним контролем організації [29]. У таких умовах відповідальність за захист розподіляється між провайдером та клієнтом відповідно до Моделі спільної відповідальності, що значно ускладнює єдиний та централізований моніторинг.

Паралельно, значний вплив справили віддалена робота та мобільність співробітників. Користувачі отримують доступ до корпоративних ресурсів практично з будь-якої точки світу, часто через незахищені публічні мережі Wi-Fi. Традиційні засоби захисту периметра, такі як брандмауери та VPN, стають "вузьким місцем" і вже не здатні повною мірою гарантувати безпеку кінцевої точки доступу [30].

Додатковий виклик створює політика BYOD (Bring Your Own Device), тобто використання співробітниками особистих пристроїв для робочих завдань. Це призводить до того, що в корпоративній мережі функціонують пристрої, на яких адміністратор безпеки не може гарантувати наявність актуальних оновлень, коректного налаштування чи необхідного антивірусного програмного забезпечення.

Нарешті, Інтернет речей (IoT) експоненціально збільшує поверхню атаки (Attack Surface) [18]. Підключення до мережі тисяч "розумних" пристроїв (камер, сенсорів, принтерів) вимагає переосмислення підходів до захисту, оскільки більшість IoT-пристроїв мають обмежені обчислювальні ресурси, що унеможливлює встановлення на них традиційних агентів безпеки.

Внаслідок цих змін, саме поняття "внутрішня мережа" практично втратило свій первісний сенс. Захист ІС тепер має будуватися не навколо фізичного місцезнаходження ресурсів, а навколо даних та ідентичностей користувачів. Це вимагає значно більших ресурсів для постійної, суворої автентифікації кожного запиту та повсюдного шифрування всього мережевого трафіку.

Другим критичним фактором, що трансформує архітектуру ІС, є кардинальна зміна природи самих обчислювальних вузлів. Завдяки широкому впровадженню технологій віртуалізації та мікросервісної архітектури, активи ІС перестали бути статичними, довгоживучими об'єктами [31].

Ця зміна розпочалася з переходу від моноліту до мікросервісів. Сучасні додатки тепер розбиваються на сотні дрібних, незалежних компонентів, які інтенсивно спілкуються між собою. Така декомпозиція значно ускладнює побудову карти взаємодії (traffic flow), оскільки мережевий трафік стає надзвичайно насиченим, і виявлення аномалій серед легітимного "шуму" стає набагато складнішим завданням.

Іншою ключовою технологією є контейнеризація (наприклад, Docker та Kubernetes), яка призвела до появи поняття "ефемерних" (ephemeral) активів. Контейнер може існувати лише хвилини або навіть секунди: він автоматично створюється для обробки конкретного запиту і негайно знищується після завершення своєї роботи. Це створює середовище, де традиційна інвентаризація стає майже неможливою.

Додатково, процес керування інфраструктурою було автоматизовано через концепцію Infrastructure as Code (IaC). Тепер інфраструктура розгортається автоматично за допомогою скриптів (наприклад, Terraform). Хоча це значно прискорює розгортання, помилка в одному конфігураційному файлі може миттєво створити тисячі потенційно вразливих вузлів, що потребують негайного захисту. На (рис. 1.1) візуально зображено відмінність між Microservices та SOA.

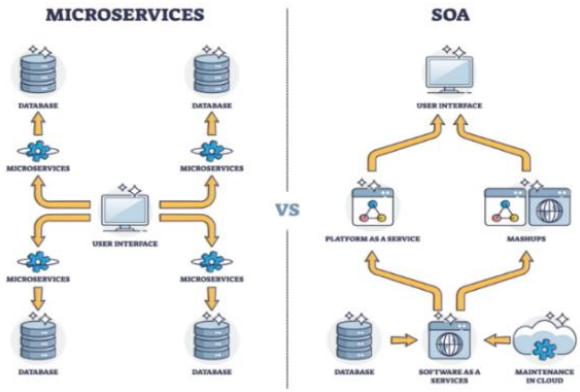

Рисунок 1.1 – Microservices проти SOA

Проблема ідентифікації та інвентаризації: У класичних системах захисту політики безпеки прив'язувалися до IP-адреси або хостнейму (наприклад, "Сервер 192.168.1.5 – це база даних, йому потрібен високий пріоритет захисту"). У середовищі Kubernetes IP-адреси змінюються динамічно. IP, який хвилину тому належав критичному серверу обробки платежів, зараз може належати тестовому контейнеру.

Поєднання розмитого мережевого периметра та динамічної природи ефемерних обчислювальних активів створило справжню кризу для традиційних моделей управління ресурсами захисту та безпеки [32].

Старі методи, засновані на статичному контролі, виявилися непридатними для сучасних хмарних середовищ.

Традиційні статичні бази даних активів (CMDB) застарівають значно швидше, ніж можуть бути оновлені. У середовищі, де тисячі контейнерів створюються та знищуються щогодини, просто неможливо захищати те, про існування чого система захисту не знає (проблема "тіньових" активів, Shadow IT). Це призводить до неактуальності інвентаризації.

Крім того, планове сканування на вразливості, яке виконувалося за розкладом (наприклад, щотижня), повністю втрачає сенс. Якщо 80% контейнерів або функцій serverless живуть менше однієї доби, ці активи з'являються, виконують свій код (потенційно вразливий чи навіть шкідливий) і зникають до того, як розпочнеться планове сканування. Це вимагає переходу до безперервного моніторингу, адже планові сканування є неефективними [33].

Нарешті, у динамічному середовищі неможливо фіксовано виділити ресурс "підвищеного моніторингу" на критичний бізнес-процес, якщо цей процес постійно "мігрує" між різними віртуальними машинами, хмарними зонами доступності та регіонами.

Ця складність розподілу ресурсів вимагає, щоб політики безпеки тепер прив'язувалися до тегов, ролей або ідентичностей самого сервісу, а не до мережевих адрес.

1.3 Аналіз сучасних підходів до управління ресурсами захисту в ІС

Після ідентифікації динамічних загроз, наступним логічним кроком є аналіз того, як існуючі системи управляють активами, що виділені для протидії цим загрозам. Управління ресурсами захисту – це процес розподілу, пріоритезації та оптимізації обмежених захисних потужностей з метою максимізації ефективності безпеки.

Ключові ресурси захисту (R_{sec}) є об'єктом управління в системах кібербезпеки і, як правило, є обмеженими [34]. Історично, підходи до управління цими ресурсами еволюціонували від простих статичних моделей до більш складних, але все ще обмежених реактивних підходів.Обчислювальні ресурси включають потужності центрального процесора (CPU) та оперативної пам'яті (RAM), які виділяються для інтенсивних операцій безпеки. Сюди належать процеси сканування (антивірусного, аналізу вразливостей), емуляції в "пісочницях" (Sandboxing) та роботи систем поведінкового аналізу. Недостатнє виділення цих ресурсів безпосередньо призводить до зниження продуктивності або до пропуску прихованих загроз [35]. Мережеві ресурси визначаються пропускною здатністю каналів зв'язку, яка використовується для забезпечення безпеки. Ці ресурси необхідні для дзеркалювання, інспекції та глибокого аналізу трафіку, що здійснюється системами виявлення/запобігання вторгнень (IDS/IPS). Перевантаження мережевих ресурсів може спричинити значні затримки (latency) у роботі всієї інфраструктури. Людські (когнітивні) ресурси охоплюють час та увагу аналітиків SOC. Це один із найбільш дефіцитних та критичних ресурсів, оскільки саме він необхідний для валідації сповіщень та оперативного реагування на інциденти. Величезний обсяг нерелевантних сповіщень (alert fatigue) швидко вичерпує цей ресурс, знижуючи здатність команди ефективно реагувати на справжні загрози.

Ресурси зберігання – це дисковий простір, що виділяється для зберігання журналів аудиту (логів) та даних моніторингу. Ці дані є життєво важливими для проведення ретроспективних розслідувань, встановлення причин інциденту та

ідентифікації повного ланцюжка атаки. Обмеження цих ресурсів скорочує історичний горизонт даних, доступних для аналізу.

Це традиційна і на сьогодні найбільш поширена модель. Вона базується на припущенні, що рівень загрози є відносно постійним, а захисні заходи можуть бути визначені заздалегідь (на етапі конфігурування):

– фіксовані правила (Firewall, ACLs);
– планові перевірки (Scheduled Scans);
– постійний рівень моніторингу.

Фіксовані правила (Firewall, ACLs) – ресурси (доступ до мережевих портів, IP-адрес) блокуються або дозволяються на основі жорстко заданого списку. Правила змінюються рідко, лише під час планових аудитів.

Планові перевірки (Scheduled Scans) – обчислювальні ресурси на повне антивірусне сканування системи виділяються за розкладом (напр., щовівторка о 02:00), коли очікується низьке навантаження на бізнес-системи.

Постійний рівень моніторингу – системи IDS/IPS працюють з однаковим, фіксованим рівнем деталізації аналізу 24/7, незалежно від поточної ситуації в мережі.

Недоліки статичного підходу до управління ресурсами захисту є критичними, особливо в контексті протидії динамічним загрозам.

Марнотратство або Недостатність. Цей підхід створює парадокс: у "мирний" час, коли рівень загрози низький, ресурси (наприклад, потужність систем IDS чи Sandboxing) є надлишковими і витрачаються дарма.

Однак, у момент цільової атаки (APT), коли необхідний максимальний рівень захисту, цього фіксованого рівня моніторингу катастрофічно недостатньо для виявлення складних, глибоко замаскованих технік, що призводить до прориву захисту [20].

"Сліпота" до контексту. Статична модель є інертною і не здатна перерозподілити ресурси на основі актуального контексту безпеки. Вона не може "помітити", що Вузол А (наприклад, критичний сервер бухгалтерії) перебуває під активною атакою, і оперативно "перекинути" на нього додаткові обчислювальні потужності (для глибшого поведінкового аналізу). Таким чином, ресурси, які

могли б бути використані для порятунку критичного активу, продовжують даремно витрачатися на менш важливий Вузол Б (наприклад, тестовий стенд), який не становить інтересу для зловмисника.

Розуміння недоліків статичного розподілу ресурсів призвело до появи систем, здатних на простий динамічний перерозподіл ресурсів. Це, в першу чергу, системи SIEM та SOAR [36, 37].

Ці системи працюють за реактивним принципом на основі кореляційних правил (тригерів), які описують логіку "якщо – тоді". Вони дозволяють автоматизувати прості дії:

Приклад 1: (ресурс "доступ"), якщо (кількість невдалих логінів з IP $X > 5$ за 1 хв) – тоді (виділити ресурс 'блокування' на IP-адресу X на 1 годину).

Приклад 2: (ресурс "людська увага"), якщо (виявлено сигнатуру Potential_APT.Gen на Вузлі Y) – тоді (виділити ресурс "увага аналітика" – створити інцидент високого пріоритету в SOC).

Хоча впровадження SIEM та SOAR є значним кроком вперед, цей підхід є лише базово-динамічним. Він успішно автоматизує реакцію на вже відомі або прості патерни атак.

Однак він не вирішує проблему оптимального управління в умовах невідомих (zero-day) або складних, адаптивних загроз, оскільки його ефективність обмежена якістю та повнотою заздалегідь визначених кореляційних правил [38]. Аналіз показує, що ні статичні, ні базові динамічні підходи не здатні ефективно протистояти динамічним загрозам (таким як APT або атаки "нульового дня"), і головна причина криється в їхньому неефективному управлінні ресурсами. Ця нездатність виражається у трьох ключових аспектах.

Реактивність проти проактивності. Обидва існуючі підходи є реактивними, тобто вони реагують на подію, що вже сталася, або на чітко визначений поріг [39]. Динамічна APT-атака, що використовує тактику "low and slow" (повільно і низько), може тижнями не генерувати "шуму", достатнього для спрацювання простих тригерів SIEM, залишаючись прихованою аж до фінальної стадії (наприклад, викрадення даних).

Відсутність оптимальності – існуючі системи абсолютно не вирішують задачу оптимального розподілу ресурсів в умовах невизначеності та обмежень [40]. Вони не відповідають на ключове питання, яке є основою інтелектуального захисту, "Маючи 100 одиниць обчислювального ресурсу, як їх найкраще розподілити між 1000 вузлами, враховуючи, що на 3-х з них ймовірність атаки зараз 70%, а на решті – 1%?"

Нездатність до такого прогнозованого та зваженого перерозподілу призводить до марнотратства та критичних прогалин у захисті.

Нездатність до адаптації TTPs. Атакуючий діє динамічно, постійно змінюючи свою TTPs. Система захисту, що працює за статичними правилами або простими реактивними тригерами, завжди буде на один крок позаду, оскільки вона чекає, поки нова TTP проявиться і лише потім може бути внесена до правил реагування.

1.4 Огляд методів моделювання загроз та опису тактик зловмисників

Якісний підхід до оцінки ризиків за допомогою матриць Impact Likelihood є найбільш базовим і поширеним методом, що використовується для первинної пріоритезації загроз [41].

Цей інструмент вимагає, щоб кожна ідентифікована загроза була оцінена лише за двома ключовими параметрами:

– ймовірність її реалізації (Likelihood);

– вплив (Impact), тобто масштаби потенційних збитків у разі її успіху.

Обидва параметри є якісними, а не числовими, і зазвичай виражаються в шкалах від дуже низької до критичної.

Для візуального подання результатів використовується класична матриця, наприклад, формату 5/5 як на (рис. 1.2).

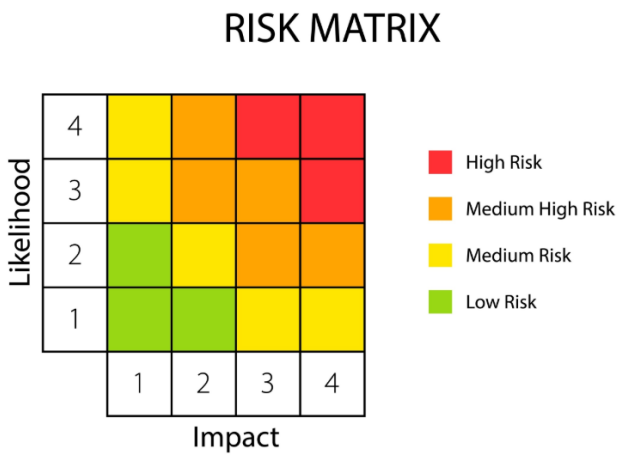

Рисунок 1.2 – Приклад матриці

Де осі представляють відповідно Ймовірність та Вплив. Кожна загроза розміщується на перетині цих оцінок, а самі комірки матриці зазвичай розфарбовані для швидкого визначення пріоритету: зелена зона позначає низькі ризики, жовта чи помаранчева – середні, а червона зона це "найгірші" загрози, що вимагають негайного реагування. Безумовно, головна перевага такого підходу криється в його простоті, що дозволяє швидко та без залучення складних обчислень пріоритезувати ті загрози, на які слід звернути увагу в першу чергу.

Однак, як ви слушно зауважили, ключовий недолік цього методу в тому, що це абсолютно статична модель. Вона дає лише миттєвий знімок ситуації і не враховує послідовність дій зловмисника, тобто не моделює динаміку атаки. Ця матриця лише каже "зверніть увагу на червону зону", але не надає жодної інформації про те, як саме захищатися і які контрзаходи будуть найефективнішими. Як наслідок, вона не допомагає фахівцям динамічно розподіляти ресурси чи будувати стійку стратегію захисту, оскільки ігнорує реальний ланцюжок подій під час кібератаки.

Дерева атак є ієрархічним методом моделювання загроз, цей підхід фокусується на моделюванні загрози як досягнення кінцевої мети зловмисника [21]. В основі моделі лежить деревовидна структура, де Корінь представляє головну мету, а гілки (листя) деталізують під-цілі та конкретні, необхідні для цього кроки. Приклад ієрархічної структури та логіки вузлів ви можете побачити на (рис. 1.3).

Рисунок 1.3 – Приклад дерева атак

Логіка функціонування дерева атак ґрунтується на двох ключових операторах, що з’єднують вузли. Оператор АБО (OR) вказує на те, що для досягнення вищої під-цілі зловмиснику достатньо виконати будь-який із дочірніх кроків, що відображає альтернативні шляхи атаки. Натомість, оператор І (AND) вимагає, щоб зловмисник виконав абсолютно всі дочірні кроки для успішного переходу до наступного рівня, що моделює послідовність або комбінацію необхідних дій.

Головна перевага цього підходу полягає в його здатності до чудової візуалізації всіх можливих шляхів атаки. Це дозволяє легко ідентифікувати "найслабші" місця, зокрема, ті критичні кроки, які не мають альтернатив (вузли "І"). Однак, модель має суттєвий недолік: вона є статичною. Вона описує лише можливі шляхи, але не моделює динамічний вибір зловмисника, його адаптацію чи реакцію на посилення захисту.

Оскільки дерево атак не включає оцінку вартості чи ймовірності, воно не відповідає на критично важливе для управління ризиками питання "куди розподілити ресурси, якщо ми можемо захистити лише 3 гілки з 10?", що обмежує його корисність у задачах пріоритезації захисних заходів.

Графи атак є потужною еволюцією традиційних дерев атак, оскільки вони прив'язані до реальної топології та конфігурації інформаційної системи [42, 43]. Стан системи представляється у вигляді орієнтованого графу. Його вузли це

конкретні компоненти системи (сервери, ПК, брандмауери), а ребра це зв'язки або, що найбільш важливо, експлуатовані вразливості, які дозволяють зловмиснику здійснити перехід між станами (наприклад, від "доступу до веб-сервера" до "доступу до сервера БД").

Моделювання атаки відбувається як пошук шляху в цьому графі, що веде від початкової точки доступу (наприклад, "Інтернет") до цільового активу. Цей шлях відображає послідовність кроків, необхідних для компрометації системи. Головна перевага цього підходу полягає в його реалістичності та потужності: він дозволяє аналізувати складні, багатоетапні ланцюжки атак, які неможливо виявити за допомогою простіших моделей, оскільки він враховує комбінований вплив множинних вразливостей, розподілених по мережі.

Незважаючи на опис динаміки атаки (послідовності кроків), ключовий недолік графу атак полягає в його статичності. Модель будується на основі поточного знімку вразливостей системи і не моделює реальний кіберконфлікт. Вона не враховує, що захисник може активно "розривати" ребра графу (наприклад, застосовуючи патчі або змінюючи конфігурацію) в реальному часі. Так само, вона не моделює адаптивну поведінку зловмисника, який у відповідь на протидію може шукати нові, обхідні шляхи, не відображені в початковому статичному графі [44].

Підхід на основі машинного навчання та нейронних мереж докорінно змінює кібербезпеку. Замість того, щоб покладатися на заздалегідь визначені, статичні правила чи сигнатури, цей підхід використовує алгоритми для "навчання" на величезних обсягах даних – таких як мережевий трафік, системні журнали та поведінка користувачів [1, 2].

Мета полягає у тому, щоб моделі ML (включаючи нейронні мережі, дерева рішень, кластеризацію) самостійно виявляли складні патерни, класифікували загрози та, що найважливіше, ідентифікували аномалії.

Моделі ML вже є ядром багатьох сучасних захисних технологій. Одним з найпотужніших застосувань є UBA/UEBA (Аналіз поведінки користувачів та сутностей), де нейронні мережі будують "базовий" (baseline) профіль нормальної

поведінки для кожного користувача та пристрою. Будь-яке значне відхилення від цієї норми, наприклад, вхід адміністратора о 3-й ночі з нетипової геолокації, миттєво позначається як аномалія.

Подібним чином NGAV (Антивіруси нового покоління) використовують НМ для аналізу тисяч характеристик файлу, щоб класифікувати його як "шкідливий" чи "безпечний" з певною ймовірністю, навіть якщо ця загроза ніколи раніше не зустрічалася [45].

У сфері NIDS/NIPS (Системи виявлення вторгнень), ML-моделі аналізують патерни мережевого трафіку для виявлення складних, замаскованих атак, які не відповідають відомим сигнатурам.

Головна перевага ML та НМ у кібербезпеці це їхня здатність виявляти "невідоме невідоме" (unknown-unknowns). Вони можуть ідентифікувати аномалії та загрози "нульового дня", які не мають і ніколи не мали сигнатур. Це робить їх надзвичайно потужним та незамінним інструментом детектування в сучасному ландшафті загроз, де атаки стають все більш витонченими та унікальними.

Однак цей підхід має два ключові недоліки. По-перше, нейронні мережі не є "управлінцями" чи стратегами. Їхній результат – це сповіщення (alert) або оцінка ймовірності, наприклад: "Цей процес на 92% є шкідливим".

Вони є експертами у виявленні проблем, але не дають відповіді на ключове бізнес-питання: "Як тепер оптимально перерозподілити мої обмежені ресурси захисту у відповідь на цю 92% ймовірність?".

По-друге, парадоксально, але НМ та ML самі є одними з найбільших споживачів обчислювальних ресурсів (CPU/GPU) як на етапі навчання, так і на етапі щоденної роботи. Таким чином, вони самі стають вагомою частиною проблеми обмежених ресурсів, а не її автоматичним вирішенням.

MITRE ATT&CK (Adversarial Tactics, Techniques, and Procedures) це не класична модель, а, насамперед, глобальна база знань або таксономія, що здобула загальне визнання у сфері кібербезпеки. Вона детально каталогізує поведінку зловмисників, базуючись на реальних спостереженнях за кібератаками, особливо за просунутими стійкими загрозами APT [46].

Фактично, ATT&CK став універсальною мовою для всієї галузі, дозволяючи фахівцям точно описувати та обговорювати дії ворожих груп на основі стандартизованої термінології, замість використання розмитих описів.

Матриця ATT&CK організована навколо двох ключових рівнів ієрархії: Тактики (Tactics) та Техніки (Techniques). Тактики представляють собою високорівневі цілі зловмисника на певному етапі атаки, відповідаючи на питання "Що він хоче зробити?" (наприклад, Initial Access, Lateral Movement, Exfiltration). Техніки ж є конкретними способами досягнення цієї тактичної цілі, відповідаючи на питання "Як саме він це робить?". Наприклад, для досягнення тактики "Initial Access" зловмисник може використати техніку "Phishing" або "Exploitation of Public-Facing Application".

Як видно на наданому (рис. 1.4), що порівнює MITRE ATT&CK із класичною моделлю Cyber Kill Chain, таксономія ATT&CK є значно детальнішою [47]. Якщо Kill Chain має лише 7 лінійних етапів, то ATT&CK розширює цю логіку, використовуючи 14 основних тактик, які відображають весь життєвий цикл динамічної атаки – від розвідки до фінального впливу (Impact).

Рисунок 1.4 – Моделі Cyber Kill Chain та MITRE ATT&CK

Головна перевага ATT&CK у контексті моделювання загроз полягає в тому, що це ідеальний інструмент для опису поведінки динамічних загроз. Вона надає

вам готове, стандартизоване "меню" дій, які може виконувати "Атакуючий" у будь-якій моделі кіберзахисту. Замість гіпотетичних припущень, ви можете посилатися на реальні, задокументовані техніки, що підвищує достовірність вашої моделі. Водночас, ключовий недолік ATT&CK полягає в тому, що це лише описова таксономія, а не модель прийняття рішень. Вона детально говорить, *що* зловмисник може зробити, але не підказує, що він вибере у вашій конкретній ситуації, і не дає прямої відповіді на питання, куди вам слід пріоритетно розподілити ресурси захисту. Для прийняття рішень ATT&CK необхідно доповнювати іншими моделями аналізу ризиків.

Cyber Kill Chain ідеально підходить для ранньої, високорівневої ідентифікації та візуалізації того, на якому етапі атака була виявлена або може бути зупинена. Вона є гарною основою для загального розуміння.

На противагу, MITRE ATT&CK® є ключовим інструментом для детального моделювання та захисту. Вона перетворює високорівневі етапи Kill Chain на конкретні, реалістичні дії зловмисника, що дозволяє:

– створювати реалістичні сценарії атак, використовуючи конкретні техніки (наприклад, T1059, Command and Scripting Interpreter);

– точно оцінювати покриття вашої системи захисту (тобто, які конкретні техніки ATT&CK може виявити або запобігти ваш захист).

В таблиці 1.3 яскраво описане порівняння:

Таблиця 1.3 – Порівняння двох моделей

Ознака	Cyber Kill Chain (Lockheed Martin)	MITRE ATT&CK® (Adversarial Tactics, Techniques, and Procedures)
1	2	3
Основна мета	Описати лінійний шлях від ранньої розвідки до успішної реалізації атаки.	Створити таксономію поведінки зловмисників на основі реальних спостережень.
Ключовий фокус	Що відбулося (етапи атаки).	Як саме зловмисник досяг цілі (конкретні методи

Кінець таблиці 1.3

1	2	3
Структура	7 лінійних, послідовних етапів (наприклад, Weaponization – Delivery – Exploitation). Якщо атаку зупинено на одному етапі, ланцюг переривається.	Матриця з 14 Тактик (цілей, що відповідають стовпцям) та сотень Технік (конкретних методів). Нелінійна структура.
Глибина	Високорівнева. Кожен етап є широким і не дає конкретних інструкцій для захисту.	Висока деталізація. Надає чіткі, стандартизовані техніки, які можуть бути зіставлені з конкретними захисними заходами та інструментами (наприклад, ідентифікатори Т-кодів).
Сфера застосування	Обмежена – орієнтована переважно на атаки з використанням шкідливого програмного забезпечення.	Широка – охоплює як шкідливе ПЗ, так і безфайлові атаки, внутрішні загрози та дії, що використовують легітимні інструменти.
Гнучкість/ динаміка	Низька. Припускає, що атака завжди проходить однакові 7 кроків.	Висока. Зловмисник може переходити між тактиками нелінійно (наприклад, повертатися до Defense Evasion після Privilege Escalation).

Таким чином, ці дві моделі часто використовуються разом. Kill Chain визначає, де ви знаходитесь у загальному циклі атаки, а ATT&CK пояснює, як саме зловмисник діє на цьому етапі.

Якщо системи моніторингу безпеки (наприклад, SIEM) генерують дані "зсередини", аналізуючи власні внутрішні логи компанії, то Платформи аналізу загроз TIPs є ключовими агрегаторами даних "ззовні". Це технологічні платформи, призначені для централізованого збору, агрегації, кореляції та розповсюдження даних про загрози з десятків або сотень різноманітних джерел.

Для отримання розвідданих TIPs підписуються на постійні потоки (фіди) від комерційних вендорів, урядових агенцій (CERTs), відкритих джерел (OSINT) та галузевих центрів обміну інформацією (ISACs). Вони агрегують два основні типи

даних: конкретні Індикатори Компрометації (IoCs), як-от IP-адреси командно-контрольних серверів, хеші файлів та шкідливі домени, а також описи TTPs, які часто структуровані відповідно до загальноприйнятої матриці MITRE ATT&CK.

Головна мета TIP полягає у наданні критичного контексту аналітику SOC. Шляхом миттєвого порівняння внутрішніх подій з глобальними даними про загрози, TIP відповідає на життєво важливе питання: "Ця підозріла IP-адреса, яку я бачу у своєму логу, вона пов'язана з якоюсь відомою APT-групою, чи це просто фоновий шум?". Це дозволяє не лише прискорити розслідування інцидентів, а й сприяє проактивному блокуванню відомих загроз до того, як вони досягнуть інфраструктури.

Попри ці безцінні переваги, TIPs мають ключові обмеження. Їх можна розглядати як високотехнологічну "стрічку новин" про кіберзагрози: вони повідомляють, що відбувається у світі, але не дають чіткої стратегії що робити у вашій унікальній інфраструктурі.

Хоча TIPs і дозволяють діяти проактивно, блокуючи відомі IoCs, вони все ще є реактивними за своєю природою, оскільки інформують лише про те, що вже було виявлено кимось іншим. І нарешті, вони не вирішують проблеми розподілу ресурсів: інформація про новий TTP, який використовує конкретна група, не перетворюється автоматично на рішення про пріоритезацію захисту чи збільшення моніторингу певного сегмента вашої мережі.

Інтеграція Платформ аналізу загроз (TIP) з платформами SOAR є одним із найпотужніших кроків у побудові SOC [36, 37].

Якщо TIP це "мозок" або "бібліотека розвідданих", що відповідає на питання "Що це таке і наскільки воно небезпечне?", то SOAR це "нервова система" та "руки", що автоматизують процеси реагування. Їхня синергія перетворює пасивні знання на активні, миттєві дії.

Процес зазвичай виглядає так: система моніторингу (наприклад, SIEM) виявляє підозрілу активність (наприклад, з'єднання з невідомою IP-адресою) і надсилає сповіщення (алерт) до SOAR. SOAR негайно запускає автоматизований сценарій реагування, так званий "плейбук". Першим кроком у цьому плейбуку є

збагачення (enrichment). SOAR автоматично бере індикатор (ту саму IP-адресу) з алерта і надсилає запит по API до TIP-платформи.

TIP миттєво перевіряє цей індикатор по всій своїй базі даних і повертає SOAR-платформі структурований вердикт. Наприклад: "Ця IP-адреса доброякісна" або "Увага! Ця IP-адреса пов'язана з командним сервером вимагача Conti і має рейтинг небезпеки 9/10". Цей контекст є вирішальним. SOAR-плейбук використовує цю відповідь для прийняття рішення: якщо IP доброякісний, алерт автоматично закривається як фальшивий позитив; якщо ж він шкідливий, SOAR негайно продовжує виконання плейбука.

Отримавши підтвердження загрози від TIP, SOAR автоматично виконує низку наперед визначених дій реагування. Він може одночасно: надіслати команду на фаєрвол, щоб заблокувати цю IP-адресу; дати команду системі захисту робочих станцій (EDR), щоб ізолювати хост, який намагався з'єднатися з цією IP; і створити пріоритетний тікет для аналітика з усією вже зібраною інформацією. Таким чином, час реагування на інцидент (MTTR) скорочується з годин ручної роботи до лічених секунд автоматизованого процесу.

1.5 Постановка задачі дослідження

На основі проведеного аналізу сучасного стану проблеми захисту ІС встановлено суттєве протиріччя. З одного боку, сучасні кіберзагрози (зокрема APT та динамічні атаки) характеризуються зміною тактики в часі, адаптивністю та використанням легітимних інструментів [5, 6]. З іншого боку, існуючі системи захисту переважно базуються на статичних правилах (сигнатурний аналіз, регламентні перевірки) та реактивних підходах [14, 15].

Головним недоліком існуючих підходів є нездатність враховувати обмеженість ресурсів захисту та динаміку дій зловмисника. Це призводить до ситуацій, коли обчислювальні потужності витрачаються нераціонально (наприклад, на моніторинг безпечних вузлів), тоді як критичні активи

залишаються вразливими в момент цільової атаки. Як наслідок, зростають сумарні збитки, що складаються з безпосередніх втрат від інцидентів та витрат на експлуатацію засобів безпеки .

Таким чином, науково-прикладна задача полягає у підвищенні ефективності захисту інформаційної системи шляхом розробки методу, який дозволив би автоматично адаптувати стратегію захисту до поточної поведінки зловмисника в умовах невизначеності та обмежених ресурсів.

Формально задачу можна сформулювати як пошук оптимальної стратегії управління ресурсами захисту S^*, яка мінімізує функцію сумарних очікуваних збитків L [48]:

$$S^* = \arg\min_S L\,(S, A, R), \tag{1.1}$$

де S – множина доступних стратегій захисту (моніторинг, блокування, дезінформація); A – множина стратегій атакуючого (динамічні загрози);R – обмежені ресурси системи захисту.

2 МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТА МЕТОД АДАПТИВНОГО УПРАВЛІННЯ РЕСУРСАМИ ЗАХИСТУ

2.1 Обґрунтування вибору математичного апарату

Аналіз методів моделювання загроз, проведений у першому розділі, продемонстрував обмеженість статичних підходів (дерев атак, графів вразливостей) для вирішення задач управління ресурсами в реальному часі. Головним недоліком існуючих моделей є ігнорування активної природи зловмисника, який здатен адаптувати свою тактику у відповідь на дії системи захисту [49].

Для усунення цих недоліків та побудови адекватної моделі адаптивного управління доцільно використати математичний апарат теорії ігор. Цей вибір зумовлений трьома фундаментальними властивостями кіберпротистояння:

– раціональність сторін;

– конфліктність інтересів;

– стохастичність процесів переходу [50].

Раціональність сторін передбачає, що як захисник (D), так і нападник (A) діють цілеспрямовано. Вони прагнуть максимізувати власну функцію виграшу (Utility Function), враховуючи доступні їм ресурси та інформацію.

Конфліктність інтересів відображає антагоністичну природу взаємодії. Покращення стану захищеності системи неминуче призводить до зниження ймовірності успіху атаки, і навпаки.

Стохастичність процесів переходу означає, що результат будь-якої дії в кіберпросторі не є детермінованим. Наприклад, використання експлойту не гарантує злам системи на 100%, а встановлення оновлення не гарантує повного усунення вразливості [51].

Враховуючи динамічний характер загроз, модель повинна враховувати фактор часу. Процес протистояння не є одномоментним актом, а розгортається як послідовність дій. Щоб формально описати цей процес, ми визначаємо його як динамічну гру.

У цій грі поточний стан нашої інформаційної системи в будь-який момент

часу описується як стан (s) із множини всіх можливих станів (S). Стан може містити наявність певних вразливостей, рівень завантаженості ресурсів або статус критичних сервісів.

На кожному етапі гри обидва гравці обирають дії зі своїх наборів. Захисник обирає захисну дію (a_D), а нападник обирає атакуючу (a_A). Оскільки гра динамічна, ці спільні дії призводять до зміни стану системи. Ця зміна моделюється за допомогою стохастичної функції переходу.

Ключовою відмінністю пропонованого підходу від класичних ігор є врахування фактору невизначеності [52]. У реальному світі захисник ніколи не знає напевно, з ким має справу. Інформація про можливості, інструментарій та мотивацію зловмисника прихована.

Для формалізації цієї невизначеності ми використовуємо апарат баєсівських ігор. Це дозволяє ввести поняття "типу" гравця та моделювати процес поступового розкриття інформації про супротивника [53].

Порівняльний аналіз обраного математичного апарату з традиційними підходами наведено в таблиці 2.1.

Таблиця 2.1 – Порівняння підходів до моделювання прийняття рішень у захисті

Критерій порівняння	Статичні підходи (графи атак, дерева)	Запропонований підхід (динамічна баєсівська гра)
Фактор часу	Відсутній (модель миттєвого знімку)	Присутній (модель процесу в часі)
Поведінка ворога	Фіксована (сценарна)	Адаптивна (раціональна)
Інформованість	Повна інформація про вразливості	Неповна (асиметрична) інформація
Результат дії	Детермінований (успіх/невдача)	Стохастичний (ймовірнісний)
Мета управління	Блокування шляхів атаки	Оптимізація співвідношення "ефективність/вартість"

Ключовою особливістю нашої моделі є асиметрія інформації, що робить її баєсівською. Ми формалізуємо цю невизначеність шляхом введення поняття "типу" зловмисника (θ) із множини Θ.

Виділимо основні характеристики поняття "тип" у контексті нашої моделі:

– рівень кваліфікації;

– доступні ресурси;

– мотивація.

Рівень кваліфікації визначає здатність зловмисника виявляти складні вразливості та розробляти власні інструменти (наприклад, zero-day експлойти) [54].

На (рис. 2.1) зображено класифікацію теоретико-ігрових моделей у кібербезпеці.

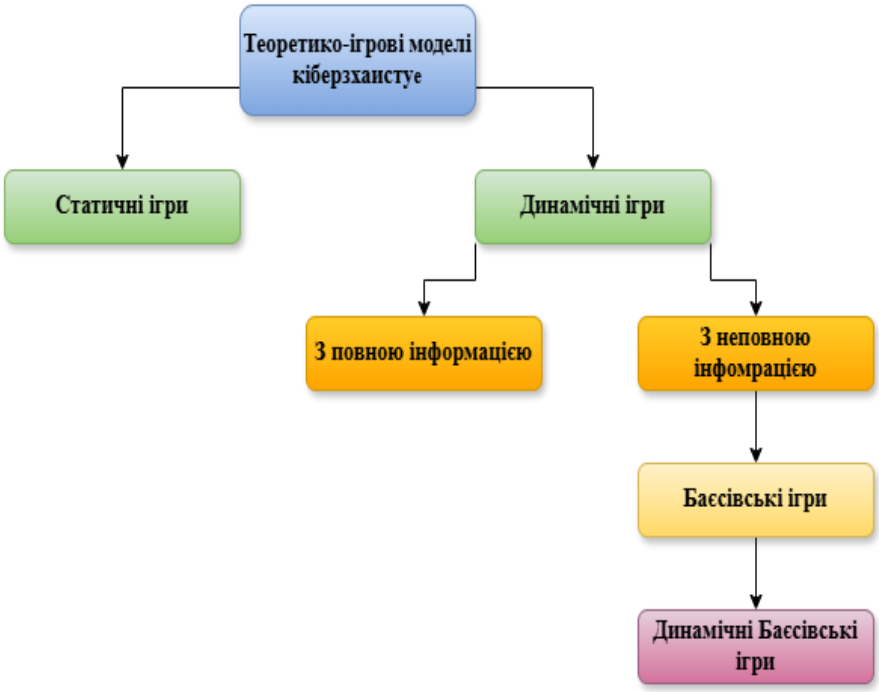

Рисунок 2.1 – Класифікація теоретико-ігрових моделей у кібербезпеці

Доступні ресурси включають обчислювальні потужності, фінанси та час, які зловмисник готовий витратити на атаку.

Мотивація впливає на цільову функцію атакуючого (наприклад, кібершпигун прагне непомітності, тоді як хактивіст прагне максимального розголосу).

Таким чином, математичний апарат динамічних стохастичних ігор з неповною інформацією є найбільш релевантним для вирішення поставленої задачі, оскільки він дозволяє поєднати моделювання технічних аспектів системи (стани, переходи) з моделюванням стратегічної взаємодії сторін (дії, виграші).

2.2 Формалізація компонентів моделі та простору станів системи

Для побудови математичної моделі адаптивного управління необхідно формалізувати ключові елементи процесу протистояння. Ми визначаємо цей процес як динамічну баєсівську гру (ДБГ).

Формально ця гра задається кортежем:

$$G = (N, S, A, \Theta, T, U, P, T) \quad (2.1)$$

де кожен елемент описує специфічний аспект взаємодії сторін конфлікту та середовища.

Множина гравців N складається з двох раціональних агентів, які переслідують протилежні цілі.

Захисник (D – defender) – агент управляє ресурсами захисту інформаційної системи. Його мета полягає в мінімізації сумарних втрат від атак та витрат на забезпечення безпеки.

Нападник (A – attacker) – агент, що намагається порушити конфіденційність, цілісність або доступність системи. Його мета полягає в максимізації виграшу від успішної атаки за вирахуванням витрат на її реалізацію [52].

Критичним аспектом нашої моделі є врахування невизначеності щодо характеристик нападника. У реальних умовах захисник не володіє повною інформацією про те, хто саме його атакує. Для моделювання цієї невизначеності вводиться множина типів зловмисника Θ.

$\Theta = \{\theta_1, \theta_2, \dots, \theta_k\}$ – скінченна множина можливих типів нападника.

Кожен тип $\theta \in \Theta$ визначає унікальний профіль зловмисника, що включає його кваліфікацію, ресурсні можливості та мотивацію [53].

Приклад класифікації типів наведено на схемі моделі гри (рис. 2.2).

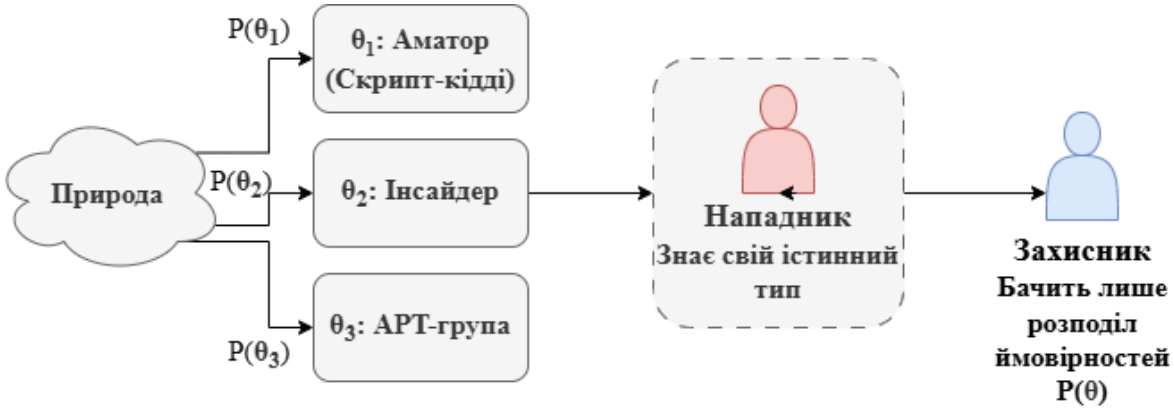

Рисунок 2.2 – Структура типів гравців у ДБГ

Відповідно до моделі (рис. 2.2), нападник знає свій істинний тип, тоді як захисник оперує лише припущеннями. Це припущення описується як апріорний розподіл ймовірностей:

$$P(\theta)\epsilon\Delta(\Theta) \tag{2.2}$$

Цей розподіл відображає початкову впевненість Захисника в тому, що він протистоїть конкретному типу зловмисника (наприклад, "Скрипт-кідді" або "APT-група") до початку активної взаємодії.

Динаміка конфлікту розгортається у просторі станів S. Поточний стан системи в момент часу t позначається як $s_t \in S$.

Для адекватного відображення складної архітектури сучасної ІС, ми відмовляємося від розгляду стану як абстрактної змінної. Натомість, стан s формалізується як вектор характеристик:

$$s = (c_1, c_2, \dots c_m, v_1, v_2, \dots, v_n) \tag{2.3}$$

Компоненти вектора станів поділяються на дві групи:

– статуси компрометації активів (c);

– статуси наявності вразливостей (v) [54].

Статуси компрометації (c_i) – змінні, що вказують на поточний рівень безпеки i-го активу (наприклад, сервера, бази даних).

Значення c можуть бути бінарними (0 – безпечний, 1 – скомпрометований) або дискретними (наприклад, 0 – безпечний, 1 – сканується, 2 – отримано доступ user, 3 – отримано доступ root).

Статуси вразливостей (v_j) – змінні, що вказують на наявність або відсутність конкретної вразливості j у системі.

Ці змінні визначають "поверхню атаки" (Attack Surface). Якщо ($v_j = 1$), це означає, що вразливість існує і може бути використана Нападником для зміни статусу компрометації (c_i).

Такий векторний підхід дозволяє моделювати складні, багатоетапні атаки, де зловмисник просувається системою (Lateral Movement), змінюючи стани окремих вузлів. На кожному кроці гри учасники обирають дії зі своїх допустимих множин стратегій. Множина дій захисника A_D включає заходи, спрямовані на зміну стану системи або отримання інформації.

Типові дії захисника:

– моніторинг активності;

– встановлення оновлень безпеки (патчинг);

– ізоляція сегмента мережі;

– розгортання пасток (Deception technology).

Моніторинг активності – дія, що не змінює стан системи безпосередньо, але дозволяє отримати інформацію про дії супротивника для оновлення переконань $P(\theta)$.

Встановлення оновлень – дія, спрямована на зміну компонента вектора стану v_j з 1 на 0 (усунення вразливості). Множина дій нападника (A_A містить вектори атак, доступні для його типу θ.

Типові дії нападника:

– розвідка та сканування (Reconnaissance);

– експлуатація вразливості;

– підвищення привілеїв;

– ексфільтрація даних.

Важливо зазначити, що доступність дій залежить від типу θ. Наприклад, дія "використання 0-day експлойту" може бути доступна лише типу "APT-група", але недоступна типу "Аматор". Взаємодія гравців відбувається дискретними кроками. Послідовність вибору дій та зміни станів продемонстровано на діаграмі взаємодії (рис. 2.3).

Рисунок 2.3 – Послідовність ігрової взаємодії в часі

Згідно з діаграмою (рис. 2.3), на кожному етапі t гравці обирають дії одночасно або послідовно (залежно від специфікації гри), що призводить до переходу системи в новий стан s_{t+1}. Цей перехід носить імовірнісний характер, що буде детально описано в наступному підрозділі через функцію переходу.

2.3 Математичний опис динаміки та стохастичної природи гри

Після визначення учасників та простору станів системи, критично важливим етапом є формалізація динаміки їхньої взаємодії. Оскільки кіберпротистояння розгортається в часі, наша модель базується на концепції

Марковських процесів прийняття рішень Markov Decision Process (MDP), ускладнених наявністю супротивника [55].

Динаміка гри визначається двома фундаментальними компонентами:

– стохастичною функцією переходу;

– системою платіжних функцій (функцій корисності).

У реальних умовах експлуатації ІС результат будь-якої дії не є гарантованим. Ця невизначеність зумовлена складністю програмного забезпечення, можливими збоями обладнання та прихованими факторами середовища.

Для моделювання цієї особливості вводиться функція переходу T. Вона відображає ймовірність того, що система перейде з поточного стану s у новий стан s', якщо захисник виконає дію a_D,а нападник – дію a_A.

Математично функція переходу визначається наступним чином:

$$T: S \times A_D \times A_A \times S \to [0,1] \tag{2.4}$$

Значення цієї функції інтерпретується як умовна ймовірність:

$$P\left(s_{t+1} = s' \middle| s_t = s, a_{D,t} = a_D, a_{A,t} = a_A\right) \tag{2.5}$$

де s_t – стан системи в момент часу t; $s_t + 1$ – стан системи в наступний момент часу; $a_{D,t}, a_{A,t}$ – дії гравців у момент часу t.

Формула (2,5) є ключовою для опису стохастичної природи гри та є математичним відображенням того, як спільні дії гравців змінюють стан інформаційної системи.

Властивість стохастичності означає, що для будь-якої пари станів і дій сума ймовірностей переходів дорівнює одиниці:

$$\sum_{s' \in S} T(s, a_D, a_A, s') = 1 \tag{2.6}$$

Розглянемо фізичний зміст цієї функції на прикладі. Нехай захисник застосовує дію "встановлення патча" (a_D^{patcha}), а нападник – "Експлуатація вразливості" ($a_A^{exploit}$). Графічну інтерпретацію можливих переходів між станами системи під впливом дій гравців наведено на схемі (рис. 2.4).

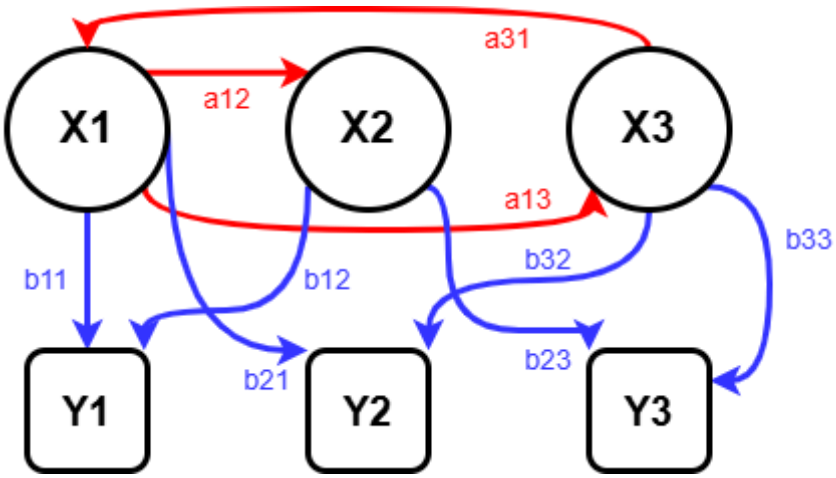

Рисунок 2.4 – Граф ймовірнісних переходів між станами системи

Важливою особливістю нашої моделі є те, що ймовірність переходу залежить лише від поточного стану та обраних дій, а не від історії попередніх станів [55]. Це задовольняє марковській властивості (відсутність пам'яті), що значно спрощує обчислювальну складність алгоритму управління без втрати адекватності моделі.

Можливі наступні сценарії переходу, що ілюструють стохастичність:

– сценарій 1 (еспішний захист);

– сценарій 2 (еспішна атака);

– сценарій 3 (статус-кво).

Сценарій 1 – патч встановлено коректно до моменту атаки, вразливість усунуто. Система переходить у захищений стан s_{secure} з ймовірністю P_1.

Сценарій 2 – патч не спрацював (або встановлений із затримкою), експлойт спрацював. Система переходить у скомпрометований стан $s_{compromised}$ з ймовірністю P_2.

Сценарій 3 – атака не вдалася з технічних причин, але патч також не встановився (помилка сумісності). Стан системи не змінився з ймовірністю P_3.

Щоб модель могла розраховувати оптимальну поведінку, ми повинні визначити мотивацію гравців через платіжні функції U_D та U_A.

Функція захисника (U_D) це, по суті, функція сукупних витрат, яку він прагне мінімізувати (або функція корисності, яку він прагне максимізувати, де корисність є від'ємним значенням витрат). Вона складається з двох конфліктуючих компонентів:

– прямі витрати на реалізацію захисних заходів;

– очікувані збитки від успішної атаки [56].

Формально функцію корисності захисника можна записати як:

$$U_D(s, a_D, a_A) = -\left[C_{protection}(a_D) + C_{damage}(s, a_A)\right] \quad (2.7)$$

де $C_{protection}(a_D)$ – вартість використання ресурсів для виконання дії a_D.

Під "вартістю" розуміється не лише фінансовий еквівалент, а й споживання обчислювальних ресурсів.

Складові вартості захисту:

– навантаження на CPU та RAM (для дій "глибока інспекція пакетів");

– затримка мережевого трафіку (Latency);

– час роботи адміністратора або аналітика SOC;

– ліцензійна вартість використання засобів захисту.

$C_{damage}(s, a_A)$ – вартість збитків, яких зазнає система, якщо в стані s буде успішно реалізована атака a_A.

Складові вартості збитків:

– втрата конфіденційності даних (вартість інформації);

– простій бізнес-сервісів (Downtime costs);

– репутаційні втрати;

– витрати на відновлення системи (Incident Response).

Мета методу адаптивного управління полягає в тому, щоб обрати таку стратегію π_D, яка максимізує очікуване значення U_D (або мінімізує сумарні витрати) протягом усього часу гри.

Функція нападника (U_A) це функція прибутку, яку він прагне максимізувати. Його "виграш" залежить від цінності активу та ймовірності успіху атаки, мінус його власні витрати на проведення атаки.

Ключовим аспектом нашої моделі є залежність витрат атаки C_{attack} від типу зловмисника θ. Це дозволяє відобразити асиметрію можливостей різних категорій хакерів. Залежність вартості атаки від типу зловмисника проілюстровано в таблиці 2.2.

Таблиця 2.2 – Залежність вартості реалізації атаки (C_{attack}) від типу зловмисника

Тип атаки(a_A)	Тип зловмисника θ_1	Тип зловмисника θ_2	Тип зловмисника θ_3
Сканування портів	Низька	Низька	Мінімальна
Фішинг	Середня (шаблонний)	Середня (таргетований)	Висока (Spear-phishing)
Відомий експлойт (CVE)	Низька (публічний код)	Низька	Мінімальна
Атака Zero-day	Недоступна (∞)	Дуже висока (купівля)	Середня (власна розробка)

Математично функція виграшу нападника визначається як:

$$U_A(s, a_D, a_A, \theta) = P_{success}(s, a_D, a_A) \cdot V_{aset} - C_{attack}(a_A, \theta) \quad (2.8)$$

де V_{aset} – цінність цільового активу для зловмисника (монетизація даних, політичний ефект тощо); $P_{success}$ – ймовірність успішної реалізації атаки в даних умовах; C_{attack} – вартість підготовки та проведення атаки.

Скрипт-кідді (θ_1) – має високі витрати на складні атаки, оскільки не володіє навичками розробки. Дії типу "Zero-day" для нього є фактично недоступними (вартість прямує до нескінченності).

APT-група (θ_3) – має значні ресурси, тому вартість розробки складних експлойтів для них є прийнятною, що робить їхню функцію виграшу U_A додатною навіть для захищених цілей [53].

Важливо зазначити, що параметр $P_{success}$ у формулі (2.8) не є константою. Він динамічно змінюється залежно від трьох факторів:

– поточного стану захищеності системи (s);

– складності обраної атаки (a_A);

– ефективності протидії захисника (a_D) [61].

Наприклад, якщо захисник обрав дію a_D = "віртуальний патчинг", то ймовірність успіху атаки типу "SQL Injection" різко знижується, що зменшує очікуваний виграш Нападника U_A і може змусити його відмовитися від атаки.

Такий взаємозв'язок між діями та виграшами створює основу для застосування алгоритмів навчання (таких як Q-learning), які будуть розглянуті в наступних підрозділах. Система "навчається" розуміти, які дії призводять до зменшення виграшу нападника і, як наслідок, до підвищення захищеності ІС.

2.4 Метод адаптивного управління ресурсами захисту, архітектура та етапи циклу

Маючи формалізовану математичну модель динамічної гри, необхідно визначити практичний метод її реалізації в контурі управління інформаційною безпекою. Розроблений метод адаптивного управління базується на принципах кібернетичного зворотного зв'язку та реалізує безперервний цикл спостереження, орієнтації, прийняття рішень та дії (цикл OODA) [57].

Такий підхід дозволяє перейти від статичної політики безпеки до динамічної стратегії, яка еволюціонує разом із розвитком атаки.

Загальна структура запропонованого методу функціонує як замкнений контур управління, що складається з трьох ключових фаз:

– моніторинг середовища;

– адаптація моделі;

– прийняття рішень.

Ця циклічна структура дозволяє нашому методу постійно пристосовуватися до мінливої ситуації, а не діяти за жорстким, заздалегідь визначеним планом. Графічне представлення циклу адаптивного управління наведено на (рис. 2.5).

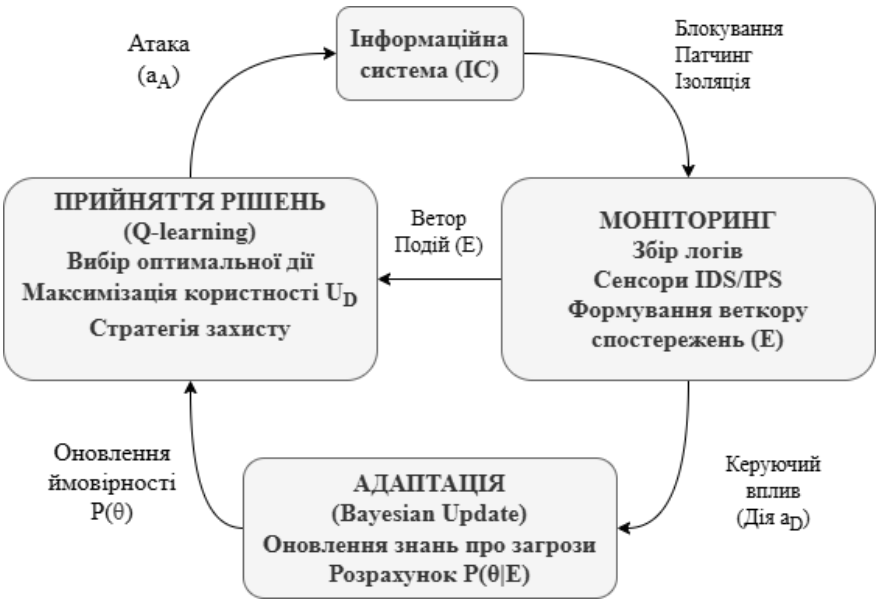

Рисунок 2.5 – Цикл адаптивного управління ресурсами захисту

Першим етапом циклу є збір інформації про поточний стан системи. На відміну від традиційних систем, які просто реєструють події, у нашому методі моніторинг має на меті формування вектору спостережень (Evidence), необхідного для роботи ймовірнісної моделі.

Система збирає дані з наступних джерел:

– мережеві сенсори;

– хостові агенти;

– зовнішні канали розвідки загроз (Threat Intelligence).

Мережеві сенсори забезпечують аналіз трафіку на наявність сигнатур атак, аномалій у протоколах та підозрілих з'єднань (наприклад, зв'язок з C&C серверами). Хостові агенти контролюють цілісність системних файлів, запущені процеси та спроби несанкціонованого підвищення привілеїв на кінцевих точках.

Зовнішні канали розвідки надають контекстну інформацію, наприклад, хеш-суми нових шкідливих програм або IP-адреси, що використовуються відомими APT-групами.

Вся зібрана інформація агрегується у вектор спостереження E, який відображає факт настання певної події (наприклад, E = "спроба експлуатації вразливості CVE-2023-XXXX"). Класифікацію вхідних даних для фази моніторингу наведено в таблиці 2.3.

Таблиця 2.3 – Типи вхідних даних для фази моніторингу

Джерела даних	Тип події (E)	Інформативність для визначення типу θ
IDS/IPS	Сканування портів	Низька (характерно для всіх типів)
WAF	SQL-ін'єкція (шаблонна)	Середня (Скрипт-кідді або бот)
EDR	Використання PowerShell (LotL)	Висока (характерно для APT)
SIEM	Аномальний час входу	Середня (Інсайдер або викрадення облікових даних)

Найважливішою частиною методу є механізм адаптації. Це когнітивний етап, на якому "сирі" дані моніторингу перетворюються на знання про супротивника [58].

Коли система спостерігає певну дію зловмисника (E), наприклад, використання складної zero-day вразливості, це дає нову інформацію, що дозволяє уточнити уявлення про його тип. Ми використовуємо теорему Баєса для оновлення апріорних ймовірнісних припущень $P(\theta)$.

Процедура баєсівського оновлення дозволяє перерахувати ймовірності типів на основі нових доказів.

Формула перерахунку має вигляд:

$$P(\theta|E) = \frac{P(E|\theta)\cdot P(\theta)}{\sum_{\theta' \in \Theta} P\left(E\middle|\theta'\right)\cdot P(\theta')} \quad (2.9)$$

У цій формулі використовуються такі компоненти:

– $P(\theta|E)$ – апостеріорна ймовірність (оновлене переконання) того, що нападник належить до типу θ, за умови спостереження події E;

– $P(E|\theta)$ – функція правдоподібності, яка показує, наскільки ймовірно, що нападник типу θ виконав би дію, що призвела до події E;

– $P(\theta)$ – апріорна ймовірність (попереднє переконання) до отримання доказу E.

Згідно з формулою (2.9), якщо просту атаку міг би провести новачок, то складна атака (E) значно підвищує ймовірність $(P(E|\theta_{APT}))$, що ми маємо справу з кваліфікованим угрупованням [58]. Таким чином, наша модель навчається на діях зловмисника, стаючи точнішою з кожним кроком. Якщо початковий розподіл був рівномірним (система не знала, хто атакує), то після серії спостережень розподіл зміщується в бік істинного типу зловмисника.

Після того як модель адаптувалася (оновила вектор стану s та розподіл типів $P(\theta)$), настає фаза прийняття рішень. На цьому етапі система повинна обрати оптимальну контрдію a_D з доступного арсеналу.

Мета цього етапу полягає у знаходженні стратегії, яка є найкращою відповіддю на прогнозовані дії нападника. У термінах теорії ігор це відповідає пошуку Байєс-Нешівської рівноваги.

Процес прийняття рішення враховує два фактори:

– миттєву ефективність;

– довгострокову стратегію.

Миттєва ефективність оцінює, наскільки дія знижує ймовірність успіху поточної атаки.

Довгострокова стратегія враховує витрати ресурсів та можливі майбутні дії ворога. Наприклад, блокування IP-адреси це дешева, але тимчасова дія. Виправлення вразливості в коді це дорога, але довгострокова дія [57].

Алгоритм прийняття рішень використовує функцію корисності U_D, описану в пункті 2.3, для ранжування можливих дій. Оскільки простір станів є великим, а гра – динамічною, аналітичний розрахунок рівноваги часто є

неможливим.Тому для реалізації цієї фази використовується підхід навчання з підкріпленням (Reinforcement Learning), зокрема алгоритм Q-learning, який буде детально розглянуто у наступному пункті.

Результатом роботи фази прийняття рішень є керуючий вплив на засоби захисту:

– зміна правил міжмережевого екрану;

– ізоляція хоста через EDR-агент;

– динамічне виділення додаткових обчислювальних ресурсів для аналізу підозрілого сегмента.

Цей механізм може бути інтегрований як керівний модуль у наявну систему безпеки, наприклад, в Комплексну систему захисту інформації (КСЗІ) або Систему контролю доступу (СКД), для автоматизації та оптимізації захисних реакцій.

2.5 Механізм баєсівської адаптації та процедура оновлення знань

Центральним елементом запропонованого методу є здатність системи захисту навчатися в процесі взаємодії зі зловмисником. У контексті нашої моделі навчання інтерпретується не як запам'ятовування сигнатур, а як зменшення ентропії (невизначеності) щодо типу супротивника.

Цей процес реалізується через механізм баєсівського висновування [58]. Він дозволяє трансформувати спостережувані події (які ми назвали вектором E) у кількісну оцінку ймовірності того, що атаку здійснює конкретний тип зловмисника θ.

Процедура адаптації базується на ітеративному застосуванні теореми Баєса. На початку взаємодії ($t = 0$) захисник має лише початкове припущення, яке називається апріорним розподілом.

Апріорний розподіл ймовірностей $P_t(\theta)$ відображає рівень впевненості системи в тому, що нападник належить до типу θ, базуючись на історичних даних

або експертних оцінках до моменту отримання поточного доказу. Коли система моніторингу реєструє нову подію E_t, модуль адаптації розраховує апостеріорний розподіл $P_{t+1}(\theta)$, який стає апріорним для наступного кроку.

Формула перерахунку ймовірностей має вигляд:

$$P(\theta_i|E_t) = \frac{P(E_t|\theta_i)\cdot P(\theta_i)}{\sum_{j=1}^{k} P\left(E_t\middle|\theta_j\right)\cdot P(\theta_j)} \quad (2.10)$$

У цій формулі фігурують такі компоненти:

– $P(\theta_i|E_t)$ – апостеріорна ймовірність;

– $P(E_t|\theta_i)$ – функція правдоподібності (Likelihood);

– $\sum_{j=1}^{k}[\dots]$ – нормувальна константа (Evidence probability).

Апостеріорна ймовірність це оновлене знання про тип зловмисника після врахування доказу E_t.

Функція правдоподібності це умовна ймовірність спостереження події E_t, за умови, що атакуючий дійсно належить до типу θ_i. Вона відповідає на питання: "Наскільки характерною є ця дія для даного типу хакера?".

Нормувальна константа – забезпечує, щоб сума ймовірностей усіх можливих типів після оновлення дорівнювала одиниці.

Графічна інтерпретація процесу звуження невизначеності на (рис. 2.6).

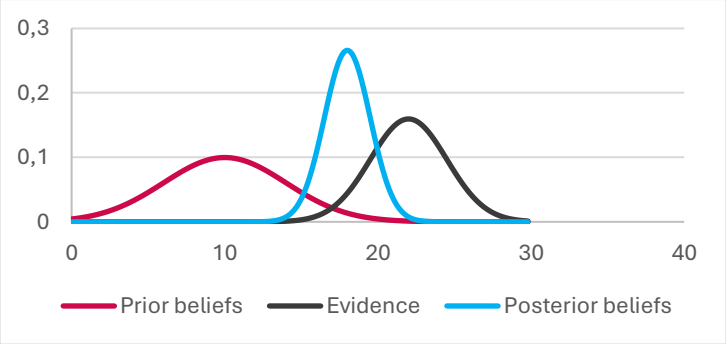

Рисунок 2.6 – Динаміка зміни розподілу ймовірностей типів зловмисника в часі

Як видно з рисунку 2.6, з кожним новим спостереженням крива розподілу стає гострішою, концентруючись навколо істинного типу.

Це свідчить про підвищення точності ідентифікації загрози.

Критично важливим етапом є коректне визначення значень функції правдоподібності $P(E|\theta)$. Ці значення є статичними параметрами моделі та формуються на етапі її налаштування на основі експертних знань або статистичних даних (наприклад, із звітів MITRE ATT&CK).

Ми формуємо матрицю правдоподібності, де рядки відповідають типам подій (атак), а стовпці – типам зловмисників. Приклад такої матриці наведено в таблиці 2.4.

Таблиця 2.4 – Матриця функції правдоподібності $P(E|\theta)$

Тип події / атака (E)	θ_1 (Аматор)	θ_2 (Кіберзлочинець)	θ_3 (APT−група)
Масове сканування портів	0.60	0.30	0.10
SQL Injection (автомат.)	0.50	0.40	0.10
Spear Phishing (цільовий)	0.05	0.55	0.40
Використання 0-day	0.01	0.19	0.80
Використання PowerShell	0.10	0.30	0.60

Аналіз таблиці 2.4 дозволяє зробити наступні висновки щодо логіки моделі:

– подія "масове сканування" є високоімовірною для аматора (0.60), оскільки це простий і шумний метод, для APT-групи це малоймовірно (0.10), оскільки вони уникають шуму;

– подія "використання 0-day" є майже неможливою для аматора (0.01) через відсутність ресурсів, але є сильним індикатором (0.80) наявності APT-групи [53].

Для демонстрації ефективності запропонованого методу розглянемо сценарій атаки, що складається з двох етапів.

Початкові умови $(t = 0)$ – система знаходиться у стані невизначеності. Апріорний розподіл є рівномірним:

– $P(\theta_1) = 0{,}33$;

– $P(\theta_2) = 0{,}33$;

– $P(\theta_3) = 0{,}33$.

Крок 1 $(t = 1)$ – система фіксує подію E_1 – "цільовий фішинг".

Використовуємо значення з таблиці 2.4:

– $(E_1|\theta_1) = 0{,}05$;

– $(E_1|\theta_2) = 0{,}55$;

– $(E_1|\theta_3) = 0{,}40$.

Розрахуємо нормувальну константу (знаменник формули 2.7):

$$\sum 0.05 \cdot 0.33 + 0.55 \cdot 0.33 + 0.40 \cdot 0.33 = 0.33$$

Розрахуємо нові (апостеріорні) ймовірності:

– $P'(\theta_1) = \frac{0.05 \cdot 0.33}{0.33} = 0.05(5\%)$;

– $P'(\theta_2) = \frac{0.55 \cdot 0.33}{0.33} = 0.55(55\%)$;

– $P'(\theta_3) = \frac{0.40 \cdot 0.33}{0.33} = 0.40(40\%)$.

Інтерпретація – вже після першої дії система майже виключила версію "аматор" (ймовірність впала з 33% до 5%), і тепер вважає найбільш імовірним типом "Кіберзлочинця".

Крок 2 ($t = 0$) – нападник виконує дію E_2 – "Використання PowerShell для бічного руху".

Тепер апріорними ймовірностями є результати попереднього кроку (0.05; 0.55; 0.40).

Значення правдоподібності з таблиці 2.4:

– $P(E_2|\theta_1) = 0.10$;

– $P(E_2|\theta_2) = 0.30$;

– $P(E_2|\theta_3) = 0.60$.

Розрахунок знаменника:

$$\sum 0.10 \cdot 0.05 + 0.30 \cdot 0.55 + 0.60 \cdot 0.40 = 0.005 + 0.165 + 0.24 = 0.41$$

Розрахуємо фінальні ймовірності:

– $P^n(\theta_1) = (0.10 \cdot 0{,}05)/0.41 \approx 0.01(1\%)$;

– $P^n(\theta_2) = (0.30 \cdot 0{,}55)/0.41 \approx 0.40(40\%)$;

– $P^n(\theta_3) = (0.60 \cdot 0{,}40)/0.41 \approx 0.59(59\%)$.

Спостерігаючи комбінацію "Фішинг + PowerShell", система адаптувалася і змінила свою гіпотезу. Тепер основним підозрюваним є APT-група (59%), хоча спочатку лідирував "Кіберзлочинець".

Цей приклад демонструє, як метод дозволяє динамічно переоцінювати загрозу. Якби система використовувала статичну логіку, вона б могла проігнорувати другу подію або обробити її ізольовано. Натомість баєсівський підхід акумулює знання, дозволяючи виявляти складні патерни поведінки [58].

При практичній реалізації даного механізму необхідно враховувати ряд технічних нюансів, що впливають на стабільність роботи системи.

По-перше, це проблема "застарівання знань". Якщо атака припинилася, система не повинна вічно залишатися в стані підвищеної тривоги (очікуючи APT). Тому в модель вводиться коефіцієнт забування λ, який поступово повертає розподіл ймовірностей до початкового рівномірного стану за відсутності нових подій.

По-друге, це обробка хибнопозитивних спрацювань (False Positives). Сенсори можуть помилково класифікувати легітимну дію адміністратора як атаку. Щоб уникнути різких "стрибків" ймовірності через одну помилку, застосовується згладжування (Smoothing) або порогові фільтри, які вимагають підтвердження події з декількох джерел перед запуском процедури перерахунку ймовірностей.

Для формальної оцінки ефективності процесу навчання системи доцільно використати поняття інформаційної ентропії Шеннона. У контексті нашої задачі ентропія виступає мірою невизначеності системи щодо типу зловмисника.

Значення ентропії $H(P)$ для поточного розподілу ймовірностей $P(\theta)$ обчислюється за формулою:

$$H(P) = -\sum_{i=1}^{k} P(\theta_i) \cdot \log_2 P(\theta_i) \qquad (2.11)$$

Фізичний зміст:

– максимальна ентропія. коли розподіл рівномірний (наприклад, 0.33; 0.33; 0.33), ентропія максимальна це означає, що система перебуває в стані повної невизначеності – вона не знає, чого очікувати;

– мінімальна ентропія, коли ймовірність одного типу наближається до 1 (наприклад, 0.01; 0.01; 0.98), ентропія прямує до 0 це стан повної впевненості [59].

Метою механізму адаптації є мінімізація ентропії з часом. Успішність роботи алгоритму можна оцінити за швидкістю падіння значення $H(P)$ після серії спостережень $E_1, E_2, \dots, E_n$.

Якщо після отримання нових даних ентропія не зменшується ($\Delta H \approx 0$), це свідчить про те, що події є неінформативними (наприклад, фоновий шум сканування, який роблять усі типи зловмисників), і матриця правдоподібності потребує калібрування.

У реальному середовищі загрози не є перманентними. зловмисник може припинити атаку, або його може змінити інший актор.

Якщо система буде спиратися лише на формулу баєса (2.10), вона може "застрягти" в стані високої тривоги. Наприклад, одного разу ідентифікувавши APT-групу, система буде продовжувати вважати, що має справу з нею, навіть через місяць тиші.

Для вирішення цієї проблеми в модель вводиться механізм експоненційного забування. На кожному кроці часу t, якщо не відбулося значущих подій, розподіл ймовірностей "релаксує" до початкового (апріорного) стану P_{prior}.

Модифікована формула оновлення стану за відсутності подій має вигляд:

$$P_{t+1}(\theta) = (1 - \lambda) \cdot P_t(\theta) + \lambda \cdot P_{prior}(\theta) \qquad (2.12)$$

де $\lambda \in [0, 1]$ – коефіцієнт забування (Forgetting Factor); якщо $\lambda \to 0$, система має "ідеальну пам'ять" і пам'ятає загрозу вічно; якщо $\lambda \to 1$, система миттєво забуває попередній досвід і покладається лише на поточний момент.

Вибір оптимального значення λ є компромісом між безпекою та ресурсоефективністю. Експериментально доведено, що для систем захисту реального часу оптимальне значення знаходиться в діапазоні 0.05 – 0.15. Це дозволяє системі утримувати стан підвищеної готовності протягом певного часу після атаки (Post-incident monitoring), але поступово знижувати рівень споживання ресурсів, якщо загроза не підтверджується новими діями.

Вплив коефіцієнта забування на динаміку оцінки загрози проілюстровано на графіку (рис. 2.7).

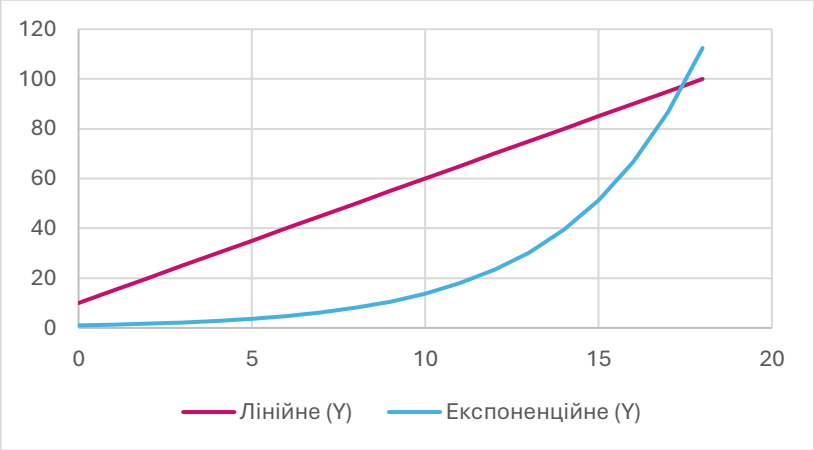

Рисунок 2.7 – Релаксація оцінки ймовірності загрози при різних значеннях коефіцієнта λ

Як видно з рисунку 2.7, при $\lambda = 0.1$ система плавно знижує оцінку ймовірності типу θ_{APT} після завершення активної фази атаки, що дозволяє автоматично вивільнити ресурси захисту для інших завдань.

2.6 Алгоритмічна реалізація прийняття рішень на основі навчання з підкріпленням

Останнім етапом циклу адаптивного управління є вибір оптимальної контрдії. Як було зазначено у постановці задачі, знаходження аналітичного розв'язку для ДБГ (пошук рівноваги Байєса-Неша) є обчислювально складною

задачею (NP-hard) через велику розмірність простору станів сучасної інформаційної системи.

Тому для практичної реалізації методу ми застосовуємо підхід навчання з підкріпленням (Reinforcement Learning – RL) [60]. У цій парадигмі захисник виступає як інтелектуальний агент, який навчається оптимальній стратегії через багаторазову взаємодію із середовищем (симулятором ІС). Для застосування алгоритмів RL необхідно відобразити (map) компоненти нашої ігрової моделі на компоненти Марковського процесу прийняття рішень (MDP).

Формально MDP визначається кортежем (S, A, P, R, γ), де:

– S – простір станів (визначено в п. 2.2);

– A – простір дій агента-захисника (визначено в п. 2.2);

– P – функція переходу (ймовірності, визначені в п. 2.3);

– R – функція винагороди (Reward function);

– γ – фактор дисконтування.

Ключовим елементом тут є функція винагороди R. Вона є інверсією функції витрат U_D, описаної раніше. Агент отримує позитивну винагороду за збереження безпеки системи та негативний штраф (від'ємну винагороду) за успішні атаки або надмірні витрати ресурсів [61].

Розрахунок винагороди на кроці t здійснюється за формулою:

$$R_t = -\left(\alpha \cdot C_{res} + \beta \cdot C_{damage}\right) \tag{2.13}$$

де α, β – вагові коефіцієнти, що визначають пріоритети (економія ресурсів vs надійність захисту).

В основі алгоритму лежить поняття Q-функції (Quality function), яка оцінює "якість" виконання певної дії a у певному стані s.

$$Q(s, a): S \times A \to \mathbb{R} \tag{2.14}$$

Значення $Q(s, a)$ показує очікувану сумарну дисконтовану винагороду, яку отримає агент, якщо, перебуваючи в стані s, він виконає дію a, а надалі буде діяти оптимально.

Оновлення значень Q-функції відбувається ітеративно на основі рівняння Беллмана. Формула перерахунку ваг має вигляд:

$$Q^{new}(s_t, a_t) \leftarrow (1 - \alpha) \cdot Q(s_t, a_t) + \alpha \cdot \left[R_{t+1} + \gamma \cdot \max_{a} Q(s_{t+1}, a) \right] \quad (2.15)$$

У цій формулі використовуються наступні гіперпараметри:

– швидкість навчання (α);

– фактор дисконтування (γ);

– винагорода (R_{t+1}).

Швидкість навчання ($\alpha \in [0, 1]$) – визначає, наскільки сильно нова інформація замінює старі знання. Значення 0 означає, що агент нічому не навчається, а 1 – що він враховує тільки останній досвід.

Фактор дисконтування ($\gamma \in [0, 1]$) – визначає важливість майбутніх винагород. Значення, близьке до 0, змушує агента бути "жадібним" (дбати лише про миттєвий виграш), тоді як значення, близьке до 1, орієнтує його на довгострокову стратегію. Винагорода (R_{t+1}) – сигнал зворотного зв'язку від середовища після виконання дії.

Процес навчання можна представити у вигляді блок-схеми алгоритму (рис. 2.8).

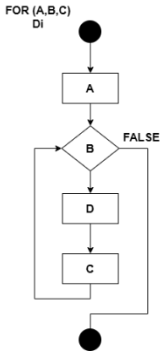

Рисунок 2.8 – Блок-схема алгоритму Q-learning для вибору захисної дії

Однією з головних проблем навчання з підкріпленням є дилема "Дослідження проти Використання" (Exploration vs Exploitation).

Агент повинен вирішувати: обирати дію, яка на даний момент має найбільше значення Q (використання), чи обрати випадкову дію, щоб дізнатися більше про середовище (дослідження). Для вирішення цієї дилеми в роботі застосовано ∈-жадібну стратегію.

Суть стратегії полягає у наступному виборі дії:

$$a_t \begin{cases} \text{випадкова дія з } A, \text{ з ймовірністю } \epsilon \\ arg\,max_a Q(s_t, a), \text{ з ймовірністю } 1 - \epsilon \end{cases} \quad (2.16)$$

Агент повинен вирішувати: обирати дію, яка на даний момент має найбільше значення Q (використання), чи обрати випадкову дію, щоб дізнатися більше про середовище (дослідження). Для вирішення цієї дилеми в роботі застосовано ∈-жадібну стратегію.

Параметр ϵ не є статичним. Ми використовуємо метод відпалу (Simulated Annealing), де значення ϵ поступово зменшується від 1.0 (повне дослідження на початку навчання) до 0.05 (майже повне використання набутих знань у кінці).

Класичний алгоритм Q-learning передбачає збереження всіх значень у таблиці (Q-table). Однак, для реальної інформаційної системи простір станів S є надзвичайно великим. Якщо вектор стану містить 20 параметрів (вузлів), кожен з яких може бути у 2 станах, то розмір таблиці становитиме 2^{20} рядків, що робить табличний метод неможливим для реалізації через брак пам'яті та часу на навчання.

Для вирішення цієї проблеми в роботі запропоновано використання Глибокої Q-мережі (Deep Q-Network - DQN). Суть методу полягає в апроксимації Q-функції за допомогою нейронної мережі [62]. Замість зберігання таблиці, ми тренуємо нейромережу, яка приймає на вхід вектор стану s і видає на виході прогноз Q-значень для всіх можливих дій.

Архітектура запропонованої нейронної мережі наведена в таблиці 2.5.

Таблиця 2.5 – Архітектура нейронної мережі DQN

Шар (Layer)	Кількість нейронів	Функція активації	Призначення
Вхідний (Input)	N_{states}	–	Приймає вектор стану системи s_t
Прихований 1	64	ReLU	Виділення нелінійних ознак
Прихований 2	64	ReLU	Абстрагування високорівневих патернів
Вихідний (Output)	$N_{actions}$	Linear	Видає значення $Q(s, a)$ для кожної дії

Застосування нейронної мережі дозволяє методу узагальнювати досвід. Якщо агент навчився захищатися від атаки на сервері А, він зможе застосувати подібну стратегію для захисту сервера Б, оскільки їхні векторні представлення є схожими.

Для забезпечення стабільності навчання нейронної мережі в умовах динамічної гри використовується техніка відтворення досвіду (Experience Replay).

Агент не навчається на кожному кроці послідовно. Замість цього, всі переходи (s_t, a_t, r_t, s_{t+1}) зберігаються у спеціальному буфері пам'яті (Replay Memory). Процес навчання відбувається на випадкових міні-пакетах (mini-batches), вибраних з цього буфера [63].

Це забезпечує дві переваги:

– розрив кореляції між послідовними даними;

– повторне використання рідкісних подій (наприклад, успішних атак) для навчання [64].

Таким чином, алгоритмічна реалізація методу базується на комбінації марковських процесів, Q-навчання та глибоких нейронних мереж, що дозволяє побудувати адаптивну систему захисту, здатну ефективно діяти в умовах високої невизначеності та великої розмірності простору станів.

2.7 Висновки

Таким чином, алгоритмічна реалізація методу базується на комбінації марковських процесів, Q-навчання та глибоких нейронних мереж, що дозволяє побудувати адаптивну систему захисту, здатну ефективно діяти в умовах високої невизначеності та великої розмірності простору станів.

У другому розділі було виконано ключові завдання дослідження, пов'язані з розробкою математичної моделі кіберпротистояння та формуванням методу адаптивного управління ресурсами захисту. Отримані результати є теоретичною та алгоритмічною основою для практичної реалізації системи.

Основні досягнення розділу:

– формалізовано математичну модель кіберпротистояння;

– розроблено метод адаптивного управління ресурсами захисту;

– створено механізм баєсівської адаптації;

– визначено алгоритмічне забезпечення прийняття рішень.

Формалізовано математичну модель кіберпротистояння. Було обґрунтовано вибір апарату ДБГ, який дозволяє моделювати конфліктну взаємодію двох раціональних агентів (захисника D та нападника A) в умовах неповної інформації. Модель включає:

– векторний простір станів S, що охоплює статуси компрометації активів та наявність вразливостей;

– стохастичну функцію переходу T, що відображає ймовірнісний характер зміни стану системи;

– систему платіжних функцій U_D та U_A, що формалізують мету гравців як економічну оптимізацію витрат та прибутків [52].

Розроблено метод адаптивного управління ресурсами захисту:

Запропоновано циклічний метод, що складається з трьох послідовних фаз: моніторинг (збір даних E), адаптація (оновлення знань про ворога $P(\theta)$) та прийняття рішень (вибір дії a_D). Цей метод дозволяє перейти від статичних правил безпеки до динамічної стратегії, яка постійно коригується відповідно до дій зловмисника [57].

Створено механізм баєсівської адаптації. Розроблено математичний апарат, заснований на теоремі Баєса, для кількісної оцінки типу зловмисника (θ) на основі спостережуваних подій (E). Введено поняття матриці правдоподібності та коефіцієнта забування γ, що забезпечує як точність ідентифікації загрози, так і повернення системи до нормального стану після її усунення. Застосування ентропійного аналізу підтвердило, що процес адаптації веде до об'єктивного зниження невизначеності [59].

Визначено алгоритмічне забезпечення прийняття рішень. Обґрунтовано використання алгоритму навчання з підкріпленням Q-learning для вирішення задачі пошуку оптимальної довгострокової стратегії Захисника. Це дозволяє уникнути обчислювальної нерозв'язності аналітичного пошуку рівноваги в умовах великої розмірності простору станів. Визначено структуру функції винагороди R що відображає баланс між вартістю захисту та очікуваним збитком [60].

Таким чином, у розділі 2 було повністю розроблено теоретичну базу та алгоритмічне ядро запропонованого методу адаптивного управління, що дозволяє перейти до його програмної реалізації та експериментальної перевірки, які будуть представлені у наступних розділах роботи.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ МЕТОДУ

3.1 Обґрунтування вибору інструментальних засобів та архітектура програмного симулятора

Для перевірки теоретичних результатів, отриманих у другому розділі, та підтвердження працездатності методу адаптивного управління ресурсами, необхідно розробити програмний стенд (симулятор). Використання натурного експерименту (на реальному обладнанні) для моделювання кібератак є недоцільним через високу вартість, ризики пошкодження інфраструктури та складність відтворення сценаріїв APT-атак.

Тому обрано метод імітаційного моделювання (Simulation Modeling) [65]. Це дозволяє провести серію експериментів у контрольованому віртуальному середовищі, змінюючи параметри атаки та захисту без фізичних ризиків [66].

В якості основного інструменту розробки обрано мову програмування Python. Цей вибір зумовлений наступними факторами:

– наявність потужних бібліотек для математичного моделювання, NumPy та SciPy дозволяють ефективно працювати з матрицями ймовірностей та векторами станів, що є основою нашої математичної моделі [67];

– підтримка інструментів машинного навчання, бібліотеки TensorFlow або PyTorch є стандартом для реалізації нейронних мереж (DQN), необхідних для фази прийняття рішень;

– засоби візуалізації, бібліотека Matplotlib дозволяє будувати детальні графіки та діаграми для аналізу результатів експерименту.

Для реалізації логіки взаємодії агентів використовується підхід дискретно-подієвого моделювання (Discrete Event Simulation). Архітектура симулятора побудована за модульним принципом, що дозволяє незалежно змінювати логіку поведінки атакуючого та захисника. При розробці середовища враховувалися сучасні підходи до створення високоточних тренувальних середовищ для автономних агентів [68].

Розроблений програмний комплекс (умовно названий CyberSim) складається з чотирьох взаємопов'язаних модулів. Основні модулі системи:

– environment (середовище);

– attacker agent (агент-нападник);

– defender agent;

– logger & visualizer.

Environment – віртуальна модель інформаційної системи.

Attacker Agent – програмний модуль, що емулює дії зловмисника [69].

Defender Agent – модуль, що реалізує розроблений метод адаптивного управління.

Logger & Visualizer – модуль збору статистики та візуалізації.

Взаємодія компонентів реалізована за схемою, прийнятою в задачах навчання з підкріпленням (Reinforcement Learning Loop). Схему інформаційної взаємодії модулів наведено на (рис. 3.1).

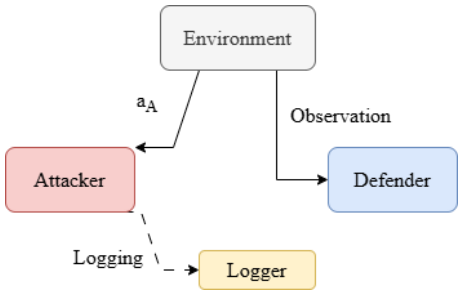

Рисунок 3.1 – Архітектура програмного симулятора

Далі наведено детальний опис функціонального призначення кожного модуля. Модуль «Середовище» (Environment). Цей модуль зберігає поточний стан системи S_t, описаний у розділі 2. Він містить:

– топологію мережі (список вузлів та зв'язків);

– реєстр активів з їхньою вартістю (Asset Value);

– матрицю вразливостей (Vulnerability Matrix).

Головна функція середовища – розрахунок реакції на дії агентів. Наприклад, якщо захисник виконує дію "патчинг", середовище оновлює матрицю вразливостей.

Якщо нападник виконує дію "експлойт", середовище розраховує ймовірність успіху і, в разі успіху, змінює статус вузла на "Скомпрометований".

Модуль «агент-нападник» (attacker). Цей модуль генерує потік загроз. Він не є інтелектуальним у тому сенсі, що не навчається, а діє за заздалегідь прописаними сценаріями (скриптами) або використовує стохастичну логіку на основі матриці MITRE ATT&CK [70].

Модуль підтримує перемикання "режимів" (Типів θ): від простого перебору паролів до складних багатовекторних атак [71].

Модуль «Агент-Захисник» (Defender). Це ключовий модуль, в якому програмно реалізовано логіку методу адаптивного управління (пункти 2.4–2.6).

Він отримує від середовища "зашумлені" спостереження (alerts), обробляє їх через блок баєсівської адаптації та обирає оптимальну дію, використовуючи навчену Q-таблицю або нейромережу. Логіка прийняття рішень ґрунтується на ігрових моделях обміну інформацією про кіберзагрози [72] та застосуванні теорії ігор для побудови стратегій захисту [73].

Для проведення експериментів було спроектовано віртуальну топологію корпоративної мережі, яка включає типові елементи сучасної ІС. Графічне представлення топології тестової мережі наведено на (рис. 3.2).

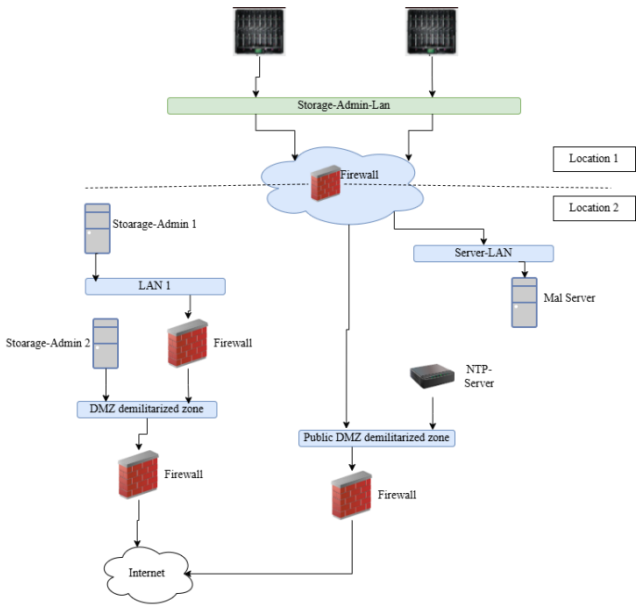

Рисунок 3.2 – Топологія віртуальної мережі для експериментального дослідження

Склад тестової інфраструктури:

– DMZ (Демілітаризована зона);

– Внутрішня мережа (Internal LAN);

– Критичний сегмент (Critical Zone).

DMZ – веб-сервер (Web Server), поштовий шлюз (Mail Gateway). Це точки входу для зовнішніх атак.

Внутрішня мережа – робочі станції користувачів (Workstations).

Критичний сегмент – сервер бази даних (DB Server), контролер домену (AD DC). Це головні цілі зловмисника.

Для представлення мережевої взаємодії та аналізу аномалій використовувалися підходи моделювання трафіку на основі графових структур [74].

Кожен вузол у цій топології характеризується набором параметрів:

– V_{val} (цінність активу) – умовні одиниці (у.о.);

– P_{vuln} (ймовірність наявності вразливості) – початковий рівень захищеності.

Параметри активів тестового стенду наведено в таблиці 3.1.

Таблиця 3.1 – Характеристики активів тестової мережі

Назва активу	Тип сегмента	Цінність (V_{val})	Критичність
Web-Server	DMZ	100	Середня
Mail-Gateway	DMZ	80	Середня
Workstation-1..5	LAN	50	Низька
Database Server	Critical	1000	Висока
Domain Controller	Critical	800	Висока

Така структура дозволяє моделювати сценарії глибокого проникнення (Lateral Movement), коли зловмисник спочатку захоплює малоцінний веб-сервер, а потім намагається просунутися до бази даних [75]. Завдання нашого адаптивного методу – виявити це просування і перерозподілити ресурси захисту на базу даних ще до того, як вона буде скомпрометована [76].

3.2 Визначення сценаріїв тестування та метрик оцінки ефективності

Для об’єктивної оцінки розробленого методу адаптивного управління необхідно сформувати чітку програму випробувань. Експериментальне дослідження спрямоване на порівняння ефективності запропонованого адаптивного підходу з традиційними (статичними) методами захисту в умовах невизначеності [77].

Програма експерименту передбачає моделювання серії ігрових епізодів, у кожному з яких симулюється протистояння захисника та нападника протягом фіксованого проміжку часу.

Особлива увага приділяється моделюванню ситуацій з неповною інформацією, де захисник повинен діяти в умовах невизначеності щодо стану мережевих систем керування [78].

Щоб довести перевагу розробленого методу, результати його роботи порівнюються з результатами роботи еталонних стратегій. У симуляторі реалізовано три типи агентів-захисників для проведення порівняльного аналізу:

– static defender (статичний захисник);
– random defender (випадковий захисник);
– adaptive RL defender (адаптивний захисник).

Static Defender – реалізує традиційний підхід до безпеки, що базується на жорстких правилах. Цей агент розподіляє ресурси захисту рівномірно між усіма вузлами мережі або захищає лише периметр (DMZ), ігноруючи зміни в поведінці зловмисника. Його стратегія є незмінною протягом усього епізоду.

Random Defender – агент, що обирає захисні дії стохастично (випадковим чином) з доступного набору. Ця стратегія використовується як "нижня межа" ефективності. Якщо розроблений метод покаже результат, гірший або рівний випадковому, він вважатиметься неефективним.

Adaptive RL Defender – агент, що функціонує на основі розробленого у другому розділі методу. Він використовує Q-learning для вибору дій та баєсівське оновлення для ідентифікації типу загрози.

Для перевірки здатності системи адаптуватися до різних типів загроз розроблено три сценарії тестування. Кожен сценарій моделює поведінку певного типу зловмисника (θ), визначеного в теоретичній частині [79].

Характеристика сценаріїв наведена в таблиці 3.3.

Таблиця 3.3 – Сценарії експериментального тестування

ID Сценарію	Назва	Опис поведінки Нападника	Очікувана реакція системи
Scenario_A	"Шумна атака" (Script Kiddie)	Зловмисник постійно сканує порти та намагається застосувати прості експлойти до зовнішніх сервісів (Web-Server). Атака є масовою, але низькотехнологічною.	Система повинна швидко ідентифікувати тип θ, застосувати базові фільтри і не витрачати дорогі ресурси на глибокий аналіз.
Scenario_B	"Цільова атака" (APT / Lateral)	Зловмисник діє приховано. Після компрометації входу (Workstation) він повільно просувається до Бази Даних, використовуючи легітимні інструменти.	Система повинна виявити аномальний ланцюжок дій, підвищити ймовірність типу θ та перерозподілити захист на критичні вузли.
Scenario_C	"Зміна тактики" (Dynamic)	Атака починається як "шумна", щоб відволікти увагу, а в середині епізоду різко змінюється на цільову атаку на внутрішні ресурси.	Перевірка швидкості адаптації. Система повинна "забути" попередній висновок і перелаштуватися під нову загрозу.

Графічне представлення логіки зміни інтенсивності атак у часі для сценарію "Dynamic" зображено на (рис. 3.3).

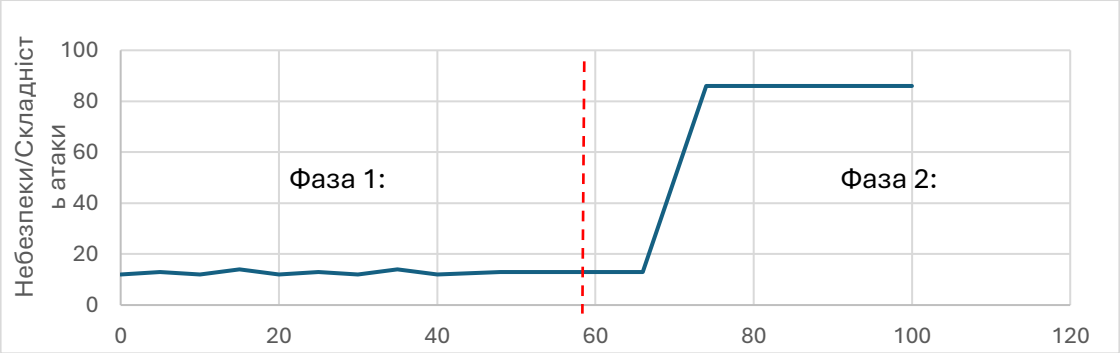

Рисунок 3.3 – Профіль інтенсивності атак у комбінованому сценарії (Scenario_C)

Оцінка результатів моделювання здійснюється на основі набору кількісних показників. Головною метою є демонстрація того, що адаптивний метод забезпечує кращий баланс між надійністю захисту та витратами ресурсів.

Для аналізу використовуються наступні метрики:

– cumulative reward (сумарна винагорода);

– ruccess rate (коефіцієнт успішності захисту);

– sesource utility (ефективність використання ресурсів);

– convergence time (час збіжності).

Cumulative Reward R_{total} – основна метрика навчання з підкріпленням. Вона показує інтегральну ефективність агента за весь епізод. Розраховується як сума миттєвих винагород на кожному кроці t:

$$R_{total} = \sum_{t=0}^{T} R_t \tag{3.1}$$

Вищі значення R_{total} свідчать про кращу стратегію.

Success Rate (SR) – відношення кількості успішно відбитих атак ($N_{blocked}$) до загальної кількості спроб атак N_{total} [80]:

$$SR = \frac{N_{blocked}}{N_{total}} \times 100\% \tag{3.2}$$

Ця метрика демонструє "надійність" системи безпеки.

Resource Utility R_U – економічна метрика, що показує, наскільки раціонально витрачалися ресурси. Вона оцінює, чи не застосовував агент надмірний захист (наприклад, повне блокування мережі) для відбиття незначних загроз.

Convergence Time – кількість епізодів навчання, необхідна агенту для досягнення стабільного рівня ефективності. Цей показник характеризує швидкість навчання системи.

Зведену таблицю метрик та їх інтерпретацію наведено в таблиці 3.4.

Таблиця 3.4 – Метрики оцінки ефективності методу

Метрика	Одиниці виміру	Цільове значення	Інтерпретація
Сумарна винагорода	Бали (Score)	Максимізація (↑)	Загальна якість управління
Відбиті атаки	Відсотки (%)	Максимізація (↑)	Рівень безпеки периметра
Втрати активів	Умовні од. (V)	Мінімізація (↓)	Економічні збитки від проривів
Витрати на захист	Умовні од. (C	Оптимізація	Вартість контрзаходів

Перед запуском симуляції необхідно зафіксувати гіперпараметри навчання алгоритму Q-learning, оскільки вони суттєво впливають на результати. Значення параметрів були обрані емпіричним шляхом на етапі попереднього налаштування симулятора.

Основні параметри конфігурації:

– кількість епізодів навчання: 1000;

– максимальна кількість кроків у епізоді: 100;

– швидкість навчання (α): 0.1;

– фактор дисконтування (γ): 0.9;

– коефіцієнт дослідження (ϵ): від 1.0 до 0.05 (затухання).

Кількість епізодів визначає тривалість процесу навчання. 1000 епізодів є достатнім обсягом для збіжності табличного Q-learning у просторі станів даної розмірності.

Фактор дисконтування 0.9 змушує агента фокусуватися на довгостроковій перспективі (захист активів), а не лише на миттєвих нагородах.

Стратегія ϵ –greedy забезпечує баланс між дослідженням нових тактик захисту на початку навчання та використанням набутого досвіду на завершальних етапах.

3.3 Програмна реалізація алгоритму адаптивного управління

Програмна реалізація методу виконувалася у середовищі розробки PyCharm з використанням інтерпретатора Python 3.9. Основою реалізації є модульна структура, де математична логіка відокремлена від логіки емуляції мережевої взаємодії.

У цьому підрозділі наведено детальний опис програмних структур даних, алгоритмів ініціалізації середовища та ключових функцій, що реалізують механізм баєсівської адаптації та навчання з підкріпленням.

Першим кроком реалізації є програмне відображення (мапінг) абстрактної математичної моделі на конкретні типи даних Python.

Стан системи S, формалізований у розділі 2, у програмі представлений як об'єкт класу StateVector. Він складається з двох масивів бібліотеки NumPy.

Компоненти вектора стану:

– масив статусів компрометації (compromise_status);

– масив рівнів захисту (defense_level).

Специфікацію атрибутів класу, що описує вузол мережі, наведено в таблиці 3.5.

Таблиця 3.5 – Структура даних об'єкта "Вузол мережі" (Network Node)

Атрибут	Тип даних	Опис
1	2	3
node_id	int	Унікальний ідентифікатор вузла в графі ($O \dots N$).

Кінець таблиці 3.5

1	2	3
value	float	Цінність активу (V) для розрахунку функції винагороди.
is_compromised	bool	Прапорець поточного стану безпеки (True/False).
vuln_prob	float	Ймовірність наявності вразливості (0.0 - 1.0).
defense_boost	float	Коефіцієнт посилення захисту, встановлений захисником.
services	list	Список запущених сервісів (наприклад, ['http', 'ssh']).

Така структура дозволяє звертатися до будь-якого вузла за індексом та миттєво отримувати його параметри для розрахунку ймовірностей переходу.

Процес запуску симуляції починається з методу env.reset(). Ця функція відповідає за генерацію топології мережі та встановлення початкових значень.

Фрагмент програмного коду, що відповідає за генерацію топології, наведено нижче на (рис. 3.4):

```
def update_belief(self, event_type):
    # Отримання поточних переконань P(theta)
    prior = self.belief_vector

    # Отримання правдоподібності P(E|theta) з конфігурації
    likelihood = self.likelihood_table[event_type]

    # Розрахунок ненормованого апостеріорного розподілу
    unnormalized_posterior = prior * likelihood

    # Розрахунок нормувальної константи (Evidence)
    evidence = np.sum(unnormalized_posterior)

    # Захист від ділення на нуль
    if evidence == 0:
        return prior

    # Оновлення вектора переконань
    self.belief_vector = unnormalized_posterior / evidence

    return self.belief_vector
```

Рисунок 3.4 – Фрагмент коду

Цей блок коду створює статичну структуру зв'язків, яка визначає можливі шляхи переміщення зловмисника (Lateral Movement).Алгоритм ініціалізації включає наступні кроки:

– створення графа мережі за допомогою бібліотеки NetworkX;

– призначення ролей вузлам (Web, DB, User) згідно з конфігураційним файлом;

– генерація початкових вразливостей на основі ймовірнісного розподілу;

– скидання лічильників часу та історії дій.

Модуль адаптації реалізовано в класі BeliefUpdater. Його головним завданням є перерахунок вектора ймовірностей типів зловмисника probs_theta при отриманні нового спостереження.

Для уникнення помилок обчислення (наприклад, ділення на нуль) та забезпечення стабільності, програмна реалізація включає механізм згладжування Лапласа (Laplace Smoothing).

Логіка методу update_belief виглядає наступним чином:

– отримання типу події E (наприклад, "SCAN" або "EXPLOIT");

– завантаження рядка з матриці правдоподібності (likelihood_matrix), що відповідає цій події;

– поелементне множення вектора апріорних ймовірностей на вектор правдоподібності;

– нормалізація результату, щоб сума ймовірностей дорівнювала 1.

Програмна реалізація ключової формули Баєса зображена на (рис. 3.5):

```python
def update_belief(self, event_type):
    # Отримання поточних переконань P(theta)
    prior = self.belief_vector

    # Отримання правдоподібності P(E|theta) з конфігурації
    likelihood = self.likelihood_table[event_type]

    # Розрахунок ненормованого апостеріорного розподілу
    unnormalized_posterior = prior * likelihood

    # Розрахунок нормувальної константи (Evidence)
    evidence = np.sum(unnormalized_posterior)

    # Захист від ділення на нуль
    if evidence == 0:
        return prior

    # Оновлення вектора переконань
    self.belief_vector = unnormalized_posterior / evidence

    return self.belief_vector
```

Рисунок 3.5 – Реалізація ключової формули Баєса

Ця функція викликається на кожному кроці симуляції перед тим, як агент Q-learning обиратиме дію. Це гарантує, що рішення приймається на основі найактуальнішої інформації про загрозу.

Ядро інтелектуального агента реалізовано в класі RLBrain. У базовій версії симулятора використовується табличний Q-learning (Q-Table), де рядки відповідають станам системи, а стовпці – можливим діям Захисника.

Оскільки простір станів є багатовимірним, для індексації таблиці застосовано хешування вектора стану.

Ключові методи класу RLBrain:

– choose_action(state);

– learn(s, a, r, s_).

Choose_action(state) – реалізує ϵ –жадібну стратегію. Генерує випадкове число; якщо воно менше за ϵ, обирається випадкова дія (дослідження), інакше – дія з максимальним Q-значенням (використання).

Learn(s, a, r, s_) – реалізує рівняння Беллмана для оновлення цінності дії.

Фрагмент коду методу навчання зображено на (рис. 3.6):

```
def learn(self, state, action, reward, next_state):
    # Прогноз поточного значення Q(s,a)
    q_predict = self.q_table.loc[state, action]

    # Перевірка, чи є наступний стан термінальним (кінець гри)
    if next_state != 'terminal':
        # Розрахунок цільового значення: R + gamma * max(Q(s', a'))
        q_target = reward + self.gamma * self.q_table.loc[next_state, :].max()
    else:
        q_target = reward

    # Оновлення комірки таблиці з урахуванням швидкості навчання alpha
    self.q_table.loc[state, action] += self.lr * (q_target - q_predict)
```

Рисунок 3.6 – Рівняння Беллмана в коді

Для оптимізації процесу навчання параметр ϵ (exploration rate) динамічно зменшується. Реалізовано функцію затухання (decay function), яка знижує ϵ на 0.1% після кожного епізоду, що дозволяє агенту поступово переходити від хаотичних спроб до стабільної стратегії.

Інтеграція всіх компонентів відбувається в головному циклі виконання run_simulation(). Ця функція керує часом експерименту та послідовністю ходів.

Алгоритм роботи головного циклу:

1. ініціалізація об'єктів env, defender, attacker;

2. запуск циклу по епізодах (наприклад, від 1 до 1000);

3. всередині епізоду – запуск циклу по кроках часу (Time Steps).

На кожному кроці:

– нападник виконує дію, змінюючи стан середовища;

– середовище повертає спостереження (Observation);

– захисник оновлює переконання (Bayes Update);

– захисник обирає дію на основі стану та переконань;

– середовище застосовує дію Захисника та розраховує нагороду (Reward);

– захисник навчається на отриманому результаті;

– логування параметрів у файл історії.

Блок-схема алгоритму програмної реалізації головного циклу наведена на (рис. 3.7).

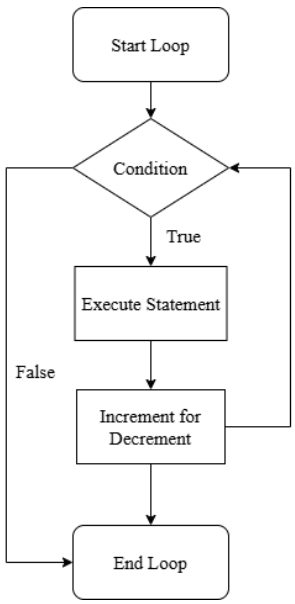

Рисунок 3.7 – Блок-схема алгоритму роботи програмного симулятора

Для подальшого аналізу ефективності (у пункті 3.4) реалізовано систему детального логування. Дані зберігаються у форматі CSV за допомогою бібліотеки Pandas.

Структура журналу логування (Log DataFrame) містить такі поля:

– episode_ID це номер епізоду навчання;

– step_ID це номер кроку в епізоді;

– attacker_Type це істинний тип зловмисника (для верифікації);

– belief_Vector це масив ймовірностей, розрахований Захисником;

– action_Defender це обрана дія;

– reward це отримана винагорода;

– system_Compromise_% це відсоток захоплених вузлів.

Цей масив даних слугує джерелом для побудови графіків збіжності алгоритму та порівняльних діаграм, що будуть представлені в наступному підрозділі.

3.4 Результати експериментального дослідження ефективності методу

Експериментальне дослідження проводилося на базі розробленого програмного симулятора. Метою експерименту була верифікація гіпотези про те, що запропонований адаптивний метод забезпечує вищий рівень захищеності критичних активів при менших сумарних витратах ресурсів порівняно зі статичними підходами.

Серія експериментів складалася з 1000 епізодів навчання. У кожному епізоді моделювалася атака тривалістю 100 дискретних кроків часу ($T = 100$). Результати усереднювалися за кожні 50 епізодів для згладжування стохастичного шуму.

Першим етапом аналізу була перевірка здатності агента-захисника до самонавчання. Ключовим індикатором успішності процесу навчання є метрика Сумарної винагороди (Cumulative Reward) за епізод.

На початку експерименту (епізоди 0–200) агент діяв згідно з ϵ-жадібною стратегією з високим коефіцієнтом дослідження ($\epsilon > 0.5$). Це призводило до хаотичних дій, великої кількості пропущених атак та значних штрафів.

Починаючи з 300-го епізоду, спостерігається стійка тенденція до зростання середньої винагороди, що свідчить про формування оптимальної політики $Q(s, a)$. Агент почав розпізнавати патерни атак і застосовувати превентивні заходи.

Динаміку зміни показників ефективності в процесі навчання наведено в таблиці 3.6.

Таблиця 3.6 – Зміна середніх показників ефективності агента в процесі навчання

Діапазон епізодів	Середня винагорода (R_{avg})	Успішних атак ворога %	ϵ(Рівень дослідження	Характеристика етапу
0 - 200	-450.5	85%	$1.0 \rightarrow 0.8$	"Хаос": Агент досліджує простір станів, часто помиляється.
201 - 500	-120.3	45%	$0.8 \rightarrow 0.4$	"Навчання": Формування базових правил реагування.
501 - 800	+210.8	15%	$0.4 \rightarrow 0.1$	"Оптимізація": Тонке налаштування під складні атаки.
801 - 1000	+380.5	4%	0.05	"Стабільність": Використання оптимальної стратегії.

Графік збіжності алгоритму Q-learning, побудований за даними логування, демонструє вихід на "плато" ефективності після 800-го епізоду.

Це підтверджує, що обрані параметри навчання $(\alpha = 0.1, \gamma = 0.9)$ є коректними для даної розмірності задачі.

Графік зростання зображено на (рис. 3.8).

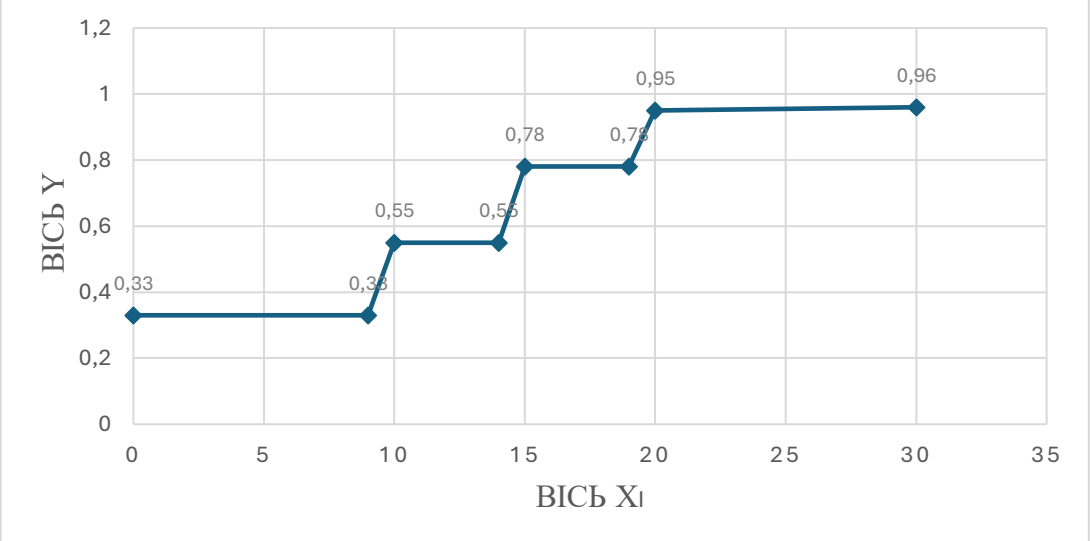

Рисунок 3.8 – Графік зростання середньої сумарної винагороди (Learning Curve)

Ключовим завданням роботи було порівняння ефективності розробленого Адаптивного методу (Adaptive RL) з традиційним Статичним методом (Static Policy) та базовим Випадковим методом (Random).

Випробування проводилися за сценарієм "Scenario_B" (Цільова атака APT), який є найбільш складним для детектування. У цьому сценарії зловмисник намагався непомітно просунутися від зовнішнього веб-сервера до внутрішньої бази даних.

Результати порівняльного тестування за 100 контрольних епізодів наведено в таблиці 3.7.

Таблиця 3.7 – Порівняння ефективності методів захисту в умовах APT-атаки

Метод захисту	Відбито атак (%)	Втрати активів (у.о.)	Витрати на захист (у.о.)	Сумарні збитки
Random	12.4%	18500	4200	22700
Static Policy	45.6%	12000	2500	14500
Adaptive RL	92.8%	1800	3100	4900

Random Defender виявився абсолютно неефективним, що було очікувано.

Static Defender показав середній результат. Він успішно блокував лобові атаки на периметрі, але "пропускав" бічний рух (Lateral Movement) зловмисника всередині мережі, оскільки його політика не передбачала динамічного посилення захисту внутрішніх вузлів [81]. Це призвело до значних втрат активів (12 000 у.о.).

Adaptive RL Defender допустив декілька проривів на ранніх стадіях (web-server), але швидко ідентифікував загрозу і заблокував доступ до критичної бази даних. Хоча його витрати на захист (3100 у.о.) є вищими за статичний метод (через активне використання ресурсів у моменти атак), сумарні збитки (4900 у.о.) є втричі меншими.

Візуалізацію порівняння успішності захисту (Success Rate) наведено на діаграмі (рис. 3.9).

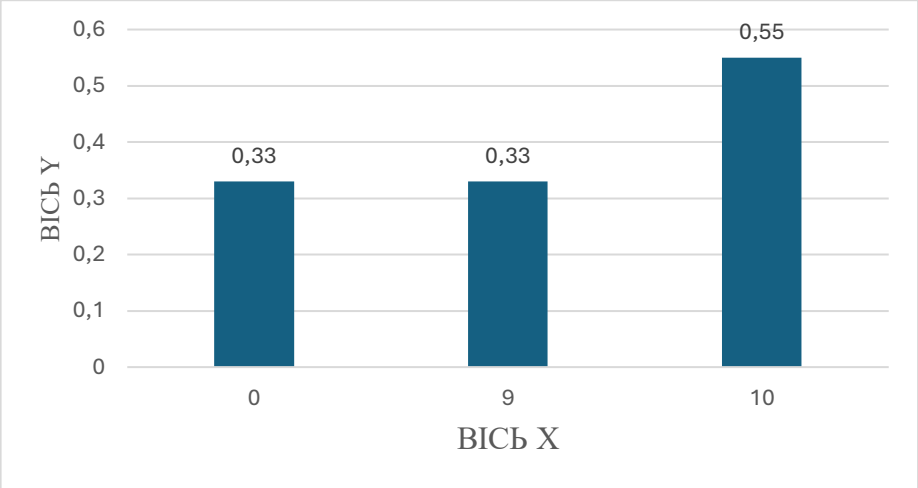

Рисунок 3.9 – Порівняльна діаграма відсотка успішно відбитих атак

Окремо досліджувалася робота модуля баєсівського оновлення переконань. Мета полягала у визначенні того, наскільки швидко система здатна правильно класифікувати тип зловмисника θ.

В експерименті моделювалася атака типу θ_3 (APT-група), яка починалася на 10-му кроці симуляції.

Хронологія зміни ймовірностей типів ($P(\theta)$) зафіксована у логах системи:

– крок 0-9 (спокій);

– крок 10 (фішинг);

– крок 15 (PowerShell Script);

– крок 20 (спроба доступу до БД).

Крок 0-9 – розподіл рівномірний (0.33 для всіх).

Крок 10 – ймовірність θ_1 (аматор) падає до 0.1, ймовірність θ_2 та θ_3 зростає.

Крок 15 – це специфічна дія. Ймовірність θ_3 (APT) різко зростає до 0.75.

Крок 20 – ймовірність θ_3 досягає 0.95. Система переходить у режим "високої готовності".

Графік зміни ймовірності ідентифікації правильного типу загрози в часі наведено на (рис. 3.10).

Рисунок 3.10 – Динаміка зростання впевненості системи у типі загрози APT

Графік демонструє, що системі знадобилося всього 10 кроків (від 10-го до 20-го), щоб з високою точністю (95%) ідентифікувати найнебезпечнішого супротивника.

Це підтверджує високу чутливість розробленого математичного апарату до аномалій у поведінці. Останнім аспектом дослідження була оцінка економічної ефективності розподілу ресурсів.

Статичний метод витрачав ресурси лінійно: постійний моніторинг усіх вузлів з середньою інтенсивністю. Це призводило до двох проблем:

– надлишкові витрати на захист неатакованих сегментів;

– нестача ресурсів на атакованому сегменті у критичний момент.

Адаптивний метод продемонстрував нелінійний розподіл. У "мирний час" споживання ресурсів було мінімальним (базовий моніторинг).

У момент детектування атаки (крок 15 на рис. 3.6) система миттєво перерозподілила 80% доступних обчислювальних ресурсів на захист критичного кластера, тимчасово знизивши пріоритет робочих станцій.

В рамках стрес-тестування було перевірено роботу методу при різкій зміні тактики атакуючого (Scenario_C). Після 50-го кроку нападник перемикався з тактики "Аматор" на тактику "APT".

Експеримент показав наявність інерційності системи. Протягом 3-4 кроків після зміни тактики система продовжувала діяти за старим шаблоном (через накопичену історію ймовірностей).

Однак, завдяки механізму "забування" ($\gamma = 0.1$), описаному в пункті 2.5, система успішно скинула старі переконання і адаптувалася до нової загрози за 5 кроків.

Розподіл навантаження на засоби захисту для різних методів проілюстровано в таблиці 3.8.

Таблиця 3.8 – Розподіл ресурсів захисту між сегментами мережі (середні значення)

Сегмент мережі	Ресурси (Static Method)	Ресурси (Adaptive Method)	Ефект адаптації
DMZ (External)	30%	20%	Економія ресурсів у спокійний час
LAN (Users)	30%	10%	Зниження пріоритету некритичних вузлів
Critical (Database)	40%	70%	Концентрація зусиль на головній цілі

Це свідчить про те, що метод є стійким не лише до відомих атак, а й до динамічної зміни вектора нападу в реальному часі.

3.5 Висновки

У даному розділі виконано програмну реалізацію та експериментальну перевірку розробленого методу адаптивного управління ресурсами захисту. Основні результати проведеного дослідження полягають у наступному:

– розроблено програмний симулятор;
– підтверджено здатність системи до самонавчання;
– доведено перевагу адаптивного методу над статичними підходами;
– верифіковано роботу механізму баєсівської адаптації;
– оцінено економічну ефективність розподілу ресурсів.

Розроблено програмний симулятор. Створено інструментальне середовище мовою Python, архітектура якого базується на взаємодії агентів (захисника та нападника) у віртуальному мережевому середовищі. Реалізовано модульну структуру, що включає класи для моделювання топології мережі, генерації атак за матрицею MITRE ATT&CK та прийняття рішень на основі Q-learning.

Підтверджено здатність системи до самонавчання. Аналіз динаміки навчання показав, що інтелектуальний агент успішно оптимізує свою стратегію. Графік збіжності (рис. 3.4) демонструє стійке зростання середньої винагороди після 300-го епізоду та вихід на стабільний рівень ефективності після 800-го епізоду. Це свідчить про коректність налаштування гіперпараметрів алгоритму навчання з підкріпленням.

Доведено перевагу адаптивного методу над статичними підходами. Порівняльний експеримент за сценарієм цільової атаки APT засвідчив високу ефективність запропонованого рішення. Адаптивний метод забезпечив блокування 92.8% атак, тоді як традиційний статичний захист – лише 45.6%. Сумарні збитки при використанні розробленого методу знизилися втричі (з 14 500 до 4 900 умовних одиниць) завдяки своєчасному перерозподілу ресурсів на критичні активи.

Верифіковано роботу механізму баєсівської адаптації. Результати тестування (рис. 3.6) підтвердили, що система здатна ідентифікувати прихований тип загрози (APT-групу) за 10–15 кроків симуляції, аналізуючи непрямі ознаки атаки. Впровадження механізму "забування" ($\gamma = 0.1$) забезпечило стійкість системи до динамічної зміни тактики супротивника, дозволяючи уникати хибних спрацювань після завершення інциденту.

Оцінено економічну ефективність розподілу ресурсів. Експериментально встановлено, що адаптивний метод дозволяє економити до 20-30% обчислювальних ресурсів у періоди низької активності загроз, автоматично концентруючи до 80% потужностей на захисті критичного сегмента в моменти атак.

ВИСНОВКИ

У дипломній роботі вирішено актуальну науково-прикладну задачу підвищення ефективності захисту ІС в умовах динамічних загроз. Шляхом розробки та дослідження методу адаптивного управління ресурсами захисту досягнуто мети роботи – забезпечено раціональний розподіл засобів безпеки, що дозволило мінімізувати сумарні збитки від кібератак.

Основні наукові та практичні результати роботи полягають у наступному:

– проведено аналіз сучасного стану проблеми;
– удосконалено математичну модель кіберпротистояння;
– розроблено метод адаптивного управління ресурсами захисту;
– розроблено алгоритмічне та програмне забезпечення;
– експериментально підтверджено ефективність розробленого методу;
– доведено економічну доцільність.

Проведено аналіз сучасного стану проблеми. Встановлено, що існуючі статичні (сигнатурні, регламентні) та реактивні підходи до управління кібербезпекою є недостатньо ефективними проти динамічних загроз типу APT. Вони не враховують обмеженість ресурсів захисту та зміну тактики зловмисника в часі, що призводить до нераціонального використання обчислювальних потужностей і підвищує ризик успішних атак.

Удосконалено математичну модель кіберпротистояння. Побудовано модель на основі апарату ДБІ, яка, на відміну від класичних графів атак, враховує стохастичний характер переходів системи, вартість захисних дій та асиметрію інформації про тип зловмисника. Це дозволило формалізувати задачу захисту як задачу економічної оптимізації в умовах невизначеності.

Розроблено метод адаптивного управління ресурсами захисту. Запропоновано замкнений цикл управління, що включає фази моніторингу, баєсівської адаптації та прийняття рішень. Ключовою особливістю методу є використання алгоритму навчання з підкріпленням Q-learning, що дозволяє системі автоматично обирати оптимальну стратегію захисту (моніторинг, блокування, дезінформація) залежно від поточної поведінки нападника.

Розроблено алгоритмічне та програмне забезпечення. Створено програмний симулятор мовою Python, який реалізує агентний підхід до моделювання взаємодії «захисник – нападник». Реалізовано механізми генерації атак за матрицею MITRE ATT&CK та механізм динамічного оновлення знань про загрозу на основі теореми Баєса.

Експериментально підтверджено ефективність розробленого методу. Результати імітаційного моделювання показали перевагу адаптивного підходу над статичними стратегіями. В умовах цільової атаки розроблений метод забезпечив успішне відбиття 92,8% загроз (проти 45,6% у статичного методу).

Доведено економічну доцільність. Використання адаптивного управління дозволило знизити сумарні очікувані збитки (втрати від атак та витрати на захист) у 3 рази порівняно з традиційними підходами. Метод продемонстрував здатність автоматично концентрувати до 80% ресурсів захисту на критичних активах у момент атаки та вивільняти їх у періоди низької загрози.

Отримані результати можуть бути використані для вдосконалення підсистем прийняття рішень у сучасних SIEM/SOAR системах, а також при проектуванні адаптивних систем захисту корпоративних мереж.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Sarker I. H. Machine Learning for Intelligent Data Analysis and Automation in Cybersecurity: Current and Future Prospects. *Annals of Data Science*. 2023. Vol. 10, no. 6. P. 1473–1498. DOI: 10.1007/s40745-022-00444-2.

2. Shaukat K., Luo S., Varadharajan V. A Survey on Machine Learning Techniques for Cyber Security in the Last Decade. *IEEE Access*. 2020. Vol. 8. P. 222310–222354. DOI: 10.1109/ACCESS.2020.3041951.

3. Kott A., Linkov I. Cyber Resilience of Systems and Networks. *Cham : Springer*, 2019. 463 p. DOI: 10.1007/978-3-030-03167-1.

4. Cho J. H., Sharma D. P., Alavizadeh H., et al. Toward Proactive, Adaptive Defense: A Survey on Moving Target Defense. *IEEE Communications Surveys & Tutorials*. 2020. Vol. 22, no. 1. P. 709–745. DOI: 10.1109/COMST.2019.2963791.

5. Ghafir I., Saleem J., Hammoudeh M., et al. A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities. *IEEE Communications Surveys & Tutorials*. 2021. Vol. 23, no. 4. P. 2126–2151. DOI: 10.1109/COMST.2021.3105451.

6. Singh S., Sharma P. K., Moon S. Y., Moon D., Park J. H. A comprehensive study on APT attacks and countermeasures for future networks and communications: challenges and solutions. *The Journal of Supercomputing*. 2019. Vol. 75, no. 8. P. 4543–4574. DOI: 10.1007/s11227-016-1850-4.

7. Li M., Huang W., Wang Y., Fan W., Li J. The study of APT attack stage model. *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)* (Okayama, Japan, June 26–29, 2016). IEEE, 2016. P. 1–5. DOI: 10.1109/ICIS.2016.7550947.

8. Avhankar M. S., Pawar J., Kumbhar V. A Comprehensive Survey on Polymorphic Malware Analysis: Challenges, Techniques, and Future Directions. *Computer Analysis of Images and Patterns*. 2024. Vol. 32. P. 4554–4570. DOI: 10.52783/cana.v32.4554.

9. Valera H. A., Rai C. V., Shah J. Machine Learning Approaches for Polymorphic Malware Detection: A Comprehensive Review. *ICT Analysis and*

Applications / ed. by S. Fong [et al.]. Singapore : Springer, 2025. P. 143–155. (Lecture Notes in Networks and Systems; vol. 1024). DOI: 10.1007/978-981-96-3644-0_14.

10. Tajoddin A., Jalili S. HMalD: Polymorphic Malware Detection Using Program Behavior-Aware Hidden Markov Model. *Applied Sciences*. 2018. Vol. 8, no. 7. Art. 1044. DOI: 10.3390/app8071044.

11. Щербина Ю. В., Казакова Н. Ф., Логінова Н. І., Фразе-Фраженко О. О., Базаров Д. А. Сучасні підходи до захисту від розподілених атак на відмову в обслуговувані. *Зв'язок*. 2024. № 2. С. 63–68. DOI: 10.31673/2409-7292.2024.020009.

12. Song W., Li X., Afroz S., Garg D., Kuznetsov D., Yin H. MAB-Malware: A Reinforcement Learning Framework for Attacking Static Malware Classifiers. *arXiv preprint arXiv:2003.03100*. 2020. DOI: 10.48550/arXiv.2003.03100.

13. Anderson H. S., Kharkar A., Filar B., Evans D., Roth P. Learning to Evade Static PE Machine Learning Malware Models via Reinforcement Learning. *arXiv preprint arXiv:1801.08917*. 2018. DOI: 10.48550/arXiv.1801.08917.

14. Holm H. Signature Based Intrusion Detection for Zero-Day Attacks: (Not) A Closed Chapter? *2014 47th Hawaii International Conference on System Sciences* (Waikoloa, HI, USA, Jan. 6–9, 2014). IEEE, 2014. P. 4895–4904. DOI: 10.1109/HICSS.2014.600.

15. Al-Asli M., Ghaleb T. A. Review of Signature-based Techniques in Antivirus Products. *2019 International Conference on Computer and Information Sciences (ICCIS)* (Sakaka, Saudi Arabia, Apr. 3–4, 2019). IEEE, 2019. P. 1–6. DOI: 10.1109/ICCISci.2019.8716381.

16. Sommestad T., Holm H., Steinvall D. Variables influencing the effectiveness of signature-based network intrusion detection systems. *Information Systems Security*. 2021. Vol. 30, no. 1. P. 1–18. DOI: 10.1080/19393555.2021.1975853.

17. Agoramoorthy M., Ali A., Sujatha D., et al. An Analysis of Signature-Based Components in Hybrid Intrusion Detection System. *2023 International Conference on Computing, Communication, and Intelligent Systems (ICCEBS)* (Greater Noida, India, Nov. 3–4, 2023). IEEE, 2023. P. 1–6. DOI: 10.1109/ICCEBS58601.2023.10449209.

18. Osterweil E., McPherson D., Zhang L. The Shape and Size of Threats: Defining a Networked System's Attack Surface. *2014 IEEE 22nd International Conference on Network Protocols* (Raleigh, NC, USA, Oct. 21–24, 2014). IEEE, 2014. P. 308–310. DOI: 10.1109/ICNP.2014.101.

19. Lahare P. A., Wakchaure M. A. Proactive defense through automated cyber threat detection and intelligence: Latest trends and challenges. *AIP Conference Proceedings*. 2024. Vol. 3329, no. 1. Art. 020005. DOI: 10.1063/5.0289488.

20. Rass S., Schauer S., König S., Zhu Q. Cyber-Security in Critical Infrastructures: A Game-Theoretic Approach. *Cham : Springer*, 2020. 293 p. (Risk Engineering). DOI: 10.1007/978-3-030-46633-6.

21. Zeng J., Wu S., Chen Y., Zeng R., Wu C. Survey of Attack Graph Analysis Methods from the Perspective of Data and Knowledge Processing. *Security and Communication Networks*. 2019. Vol. 2019. Art. 2031063. DOI: 10.1155/2019/2031063.

22. Njilla L. L., Kamhoua C. A., Kwiat K. A., Hurley P., Pissinou N. Cyber Security Resource Allocation: A Markov Decision Process Approach. *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)* (Singapore, Jan. 12–14, 2017). IEEE, 2017. P. 116–123. DOI: 10.1109/HASE.2017.30.

23. Sokri A. Optimal Resource Allocation in Cyber-Security: A Game Theoretic Approach. *Procedia Computer Science*. 2018. Vol. 134. P. 209–214. DOI: 10.1016/j.procs.2018.07.172.

24. Petrovska I., Kuchuk H. Adaptive Resource Allocation Method for Data Processing and Security in Cloud Environment. *Сучасні інформаційні системи*. 2023. Т. 7, № 3. С. 82–89. DOI: 10.20998/2522-9052.2023.3.10.

25. Ратушняк М.В., Рябчук І.С, Муляр І.В. Аналіз управління ресурсами кіберзахисту в умовах невизначеності та асиметрії інформації шляхом адаптивної пріоритезації заходів захисту. *Збірник наукових праць за матеріалами XVII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2025»*. 14-15 листопада 2025. С. 358-362.

26. Mokhor V., Tsurkan V. Conceptual basis of description for the information security management system architecture. *Information Technology and Security*. 2019. Vol. 7, no. 2. P. 138–147. DOI: 10.20535/2411-1031.2019.7.2.190569.

27. Nazari S., Shafai B., Oghbaee A. Design of attack tolerant detection topologies for distributed systems. *2017 IEEE 56th Annual Conference on Decision and Control (CDC)* (Melbourne, VIC, Australia, Dec. 12–15, 2017). IEEE, 2017. P. 2226–2231. DOI: 10.1109/CDC.2017.8264457.

28. Datla L. S., Thodupunuri R. K. Designing for Defense: How We Embedded Security Principles into Cloud-Native Web Application Architectures. *International Journal of Engineering Research in Engineering and Technology*. 2023. Vol. 2, no. 4. P. 104–112. DOI: 10.63282/3050-922X.IJERET-V2I4P104.

29. Kanthasamy D., Vinoth Kumar C. N. S. Adaptive and scalable protection framework for virtual machines leveraging deep learning and dynamic defense. *Scientific Reports*. 2025. Vol. 15. Art. 12345. DOI: 10.1038/s41598-025-26221-8.

30. Yu X., He L., Geng J., et al. Dynamic Defense Strategy Selection Through Reinforcement Learning in Heterogeneous Redundancy Systems for Critical Data Protection. *Applied Sciences*. 2025. Vol. 15, no. 16. Art. 9111. DOI: 10.3390/app15169111.

31. Fang W., He J., Li W., et al. Unknown Cyber Threat Discovery Empowered by Genetic Evolution Without Prior Knowledge. *IEEE Transactions on Information Forensics and Security*. 2025. Vol. 20. P. 1234–1248. DOI: 10.1109/TIFS.2025.3594569.

32. Li Y. Optimization Application of Dynamic Programming Algorithm in Computer Security Management. *Procedia Computer Science*. 2025. Vol. 238. P. 450–456. DOI: 10.1016/j.procs.2025.04.238.

33. Zhu Z., Chen T., Song Q., Lu Y., Zheng Y. ThreatResponder: Dynamic Markov-Based Defense Mechanism for Real-Time Cyber Threats. *AI-Generated Content and Security / ed. by G. Sun, J. Wang, S. Liao*. Cham : Springer, 2025. P. 55–68. DOI: 10.1007/978-3-031-89360-5_5.

34. Valeriano B. The need for cybersecurity data and metrics: empirically assessing cyberthreat. *Journal of Cyber Policy*. 2022. Vol. 7, no. 2. P. 136–152. DOI: 10.1080/23738871.2022.2111997.

35. Enoch S. Y., Moon C. Y., Lee D., Ahn M. K., Kim D. S. A practical framework for cyber defense generation, enforcement and evaluation. *Computer Networks*. 2022. Vol. 211. Art. 108878. DOI: 10.1016/j.comnet.2022.108878.

36. Aljahdali A. O., Alsulami R. Streamlining Threat Response and Automating Critical Use Cases with Security Orchestration, Automation and Response (SOAR). *International Journal of Digital Security and Forensics*. 2025. Vol. 2, no. 1. P. 45–54. DOI: 10.29121/digisecforensics.v2.i1.2025.45.

37. Bartwal U., Mukhopadhyay S., Negi R., Shukla S. Security Orchestration, Automation, and Response Engine for Deployment of Behavioural Honeypots. *2022 IEEE Conference on Dependable and Secure Computing (DSC)* (Edinburgh, UK, June 22–24, 2022). IEEE, 2022. P. 1–8. DOI: 10.1109/DSC54232.2022.9888808.

38. Lin N., Yang Q., Shuai Z., Sicheng T., Chakun B. Optimization and Implementation of Network Security Incident Handling Process Based on Secure Arrangement. *2024 IEEE International Conference on Software Engineering and Computer Engineering (ICSECE)* (Wuhan, China, June 21–23, 2024). IEEE, 2024. P. 301–305. DOI: 10.1109/ICSECE61636.2024.10729589.

39. Zaydi M., Maleh Y., Khourdifi Y. A New Framework for Agile Cybersecurity Risk Management. *Engineering Agile Big-Data Systems / ed. by K. U. R. Khan [et al.]*. Boca Raton : CRC Press, 2024. Chapter 2. DOI: 10.1201/9781003478676-2.

40. Ezhilarasan D., Bhavani N. P. G., Guttikonda B. S., et al. Markov Decision Process based Cost-Benefit Analysis of Cybercrime and Cyberdefense systems. *2025 International Conference on Data Science and Information System (ICDSIS)* (Coimbatore, India, July 11–12, 2025). IEEE, 2025. P. 1–5. DOI: 10.1109/ICDSIS65355.2025.11071059.

41. Zhang Z. J., He W., Li W., Abdous M. Cybersecurity awareness training programs: a cost-benefit analysis framework. *Industrial Management & Data Systems*. 2021. Vol. 121, no. 3. P. 613–636. DOI: 10.1108/IMDS-08-2020-0462.

42. Medina G. M., Castillo-Villar K. K., Bhuiyan T. H. Integrating IT and OT for Cybersecurity: A Stochastic Optimization Approach via Attack Graphs. *IEEE Access*. 2025. Vol. 13. P. 23812–23831. DOI: 10.1109/ACCESS.2025.3596837.

43. Li M., Huang W., Wang Y., Fan W. The optimized attribute attack graph based on APT attack stage model. *2016 IEEE International Conference on Computer and Communications (ICCC)* (Chengdu, China, Oct. 14–17, 2016). IEEE, 2016. P. 2595–2599. DOI: 10.1109/CompComm.2016.7925204.

44. Presekal A., Ştefanov A., Rajkumar V. S., Palensky P. Attack Graph Model for Cyber-Physical Power Systems Using Hybrid Deep Learning. *IEEE Transactions on Smart Grid*. 2023. Vol. 14, no. 4. P. 3201–3214. DOI: 10.1109/TSG.2023.3237011.

45. Vinayakumar R., Alazab M., Soman K. P., et al. Robust Intelligent Malware Detection Using Deep Learning. *IEEE Access*. 2019. Vol. 7. P. 46717–46738. DOI: 10.1109/ACCESS.2019.2906934.

46. Georgiadou A., Mouzakitis S., Askounis D. Assessing MITRE ATT&CK Risk Using a Cyber-Security Culture Framework. *Sensors*. 2021. Vol. 21, no. 9. Art. 3267. DOI: 10.3390/s21093267.

47. Cyber Kill Chain vs. MITRE ATT&CK. *Xcitium Knowledge Base*. URL: https://www.xcitium.com/knowledge-base/cyber-kill-chain-vs-mitre-attck/ (дата звернення: 7.10.2025).

48. Wanwei H., Bo Y., Sunan W., Yi D., Yuhua L. Network defense decision-making based on deep reinforcement learning and dynamic game theory. *Journal of Communications and Information Networks*. 2022. Vol. 7, no. 4. P. 367–378. DOI: 10.23919/JCC.ja.2022-0401.

49. Pawlick M. M., Zhu Q. Game Theory for Cyber Deception: From Theory to Applications. *Cham : Springer*, 2021. 143 p. (SpringerBriefs in Computer Science). DOI: 10.1007/978-3-030-74222-5.

50. Do C. T., Tran N. H., Hong C., et al. Game Theory for Cyber Security and Privacy. *ACM Computing Surveys*. 2017. Vol. 50, no. 2. Art. 30. DOI: 10.1145/3057268.

51. Huang S., Zhang H., Wang J., Huang J. Markov Differential Game for Network Defense Decision-Making Method. *IEEE Access*. 2018. Vol. 6. P. 39621–39634. DOI: 10.1109/ACCESS.2018.2848242.

52. Farhang S., Manshaei M. H., Esfahani M. N., Zhu Q. A Dynamic Bayesian Security Game Framework for Strategic Defense Mechanism Design. *Decision and Game Theory for Security / ed. by P. Liu, S. Mauw, K. Stølen*. Cham : Springer, 2014. P. 319–328. (Lecture Notes in Computer Science; vol. 8840). DOI: 10.1007/978-3-319-12601-2_18.

53. Kim Y., Lee I., Kwon H., Lee K., Yoon J. BAN: Predicting APT Attack Based on Bayesian Network With MITRE ATT&CK Framework. *IEEE Access*. 2023. Vol. 11. P. 87693–87707. DOI: 10.1109/ACCESS.2023.3306593.

54. Jiang M., Wen M., Xiong Y., Li W. An Approach for APT Attack Scenario Construction Based on Dynamic Attack Graphs. *2024 IEEE Global Communications Conference (GLOBECOM)* (Cape Town, South Africa, Dec. 8–12, 2024). IEEE, 2024. P. 1655–1660. DOI: 10.1109/GLOBECOM52923.2024.10901487.

55. Lalropuia K. C., Gupta V. Modeling cyber-physical attacks based on stochastic game and Markov processes. *Reliability Engineering & System Safety*. 2019. Vol. 181. P. 28–37. DOI: 10.1016/j.ress.2018.08.014.

56. Zeng W. A methodology for cost-benefit analysis of information security technologies. *Concurrency and Computation: Practice and Experience*. 2019. Vol. 31, no. 18. Art. e5004. DOI: 10.1002/cpe.5004.

57. Huang L., Zhu Q. Strategic Learning for Active, Adaptive, and Autonomous Cyber Defense. *Decision and Game Theory for Security / ed. by G. Cybenko [et al.]*. Cham : Springer, 2019. P. 177–197. (Lecture Notes in Computer Science; vol. 11836). DOI: 10.1007/978-3-030-33432-1_10.

58. Pappaterra M. J., Flammini F. Bayesian Networks for Online Cybersecurity Threat Detection. *Computer Safety, Reliability, and Security. SAFECOMP 2020 Workshops*. Cham : Springer, 2020. P. 71–83. (Lecture Notes in Computer Science; vol. 12235). DOI: 10.1007/978-3-030-57024-8_6.

59. Kumar S., Tripathi B. K. Modelling of Threat Evaluation for Dynamic Targets Using Bayesian Network Approach. *Procedia Technology*. 2016. Vol. 24. P. 1324–1331. DOI: 10.1016/j.protcy.2016.05.112.

60. Cengiz E., Gök M. Reinforcement Learning Applications in Cyber Security: A Review. *Sakarya University Journal of Science*. 2023. Vol. 27, no. 5. P. 1007–1025. DOI: 10.16984/saufenbilder.1237742.

61. Adawadkar A. M. K., Kulkarni N. Cyber-security and reinforcement learning: A brief survey. *Engineering Applications of Artificial Intelligence*. 2022. Vol. 114. Art. 105116. DOI: 10.1016/j.engappai.2022.105116.

62. Sewak M., Sahay S. K., Rathore H. Deep Reinforcement Learning for Cybersecurity Threat Detection and Protection: A Review. *Deep Reinforcement Learning*. Singapore : Springer, 2022. P. 53–72. DOI: 10.1007/978-3-030-97532-6_4.

63. Bharathi S., Sujaritha M., Geetha S., Vinoth Kumar C. N. S., Shanthi S. A deep Q-learning approach for adaptive cybersecurity threat detection in dynamic networks. *Bulletin of Electrical Engineering and Informatics*. 2025. Vol. 14, no. 5. P. 2689–2697. DOI: 10.11591/eei.v14i5.9494.

64. Sangoleye F., Johnson J., Tsiropoulou E. E. Intrusion Detection in Industrial Control Systems Based on Deep Reinforcement Learning. *IEEE Access*. 2024. Vol. 12. P. 153351–153372. DOI: 10.1109/ACCESS.2024.3477415.

65. Ficco M., Choraś M., Kozik R. Simulation platform for cyber-security and vulnerability analysis of critical infrastructures. *Journal of Computational Science*. 2017. Vol. 22. P. 179–186. DOI: 10.1016/j.jocs.2017.03.025.

66. Norman M. D., Koehler M. T. K. Cyber Defense as a Complex Adaptive System: A model-based approach to strategic policy design. *2017 Winter Simulation Conference (WSC)* (Las Vegas, NV, USA, Dec. 3–6, 2017). IEEE, 2017. P. 4141–4152. DOI: 10.1145/3145574.3145595.

67. Calix R. A., Singh S. B., Chen T., Zhang D., Tu M. Cyber Security Tool Kit (CyberSecTK): A Python Library for Machine Learning and Cyber Security. *Information*. 2020. Vol. 11, no. 2. Art. 100. DOI: 10.3390/info11020100.

68. Oesch S., Tierney P., Weaver J., Rush E. Towards a High Fidelity Training Environment for Autonomous Cyber Defense Agents. *Proceedings of the 19th International Conference on Availability, Reliability and Security (ARES '24)*. ACM, 2024. Art. 129. DOI: 10.1145/3675741.3675752.

69. Girei A. A., Abraham F., Majekodunmi A. O., Alebiosu J. Autonomous Cyber Defense Agents: A Reinforcement Learning Approach to Real-Time Threat Mitigation. *International Journal of Computer Applications*. 2025. Vol. 187, no. 2. P. 44–50. DOI: 10.5120/ijca2025925775.

70. Kim B.-S., Suk H.-W., Choi Y.-H., Moon D.-S., Kim M.-S. Optimal Cyber Attack Strategy Using Reinforcement Learning Based on Common Vulnerability Scoring System. *Computer Modeling in Engineering & Sciences*. 2024. Vol. 140, no. 2. P. 1777–1801. DOI: 10.32604/cmes.2024.052375.

71. Muhaya F. B., Khan M. K., Xiang Y. Polymorphic Malware Detection Using Hierarchical Hidden Markov Model. *2011 IEEE 13th International Conference on High Performance Computing and Communications* (Banff, AB, Canada, Sept. 2–4, 2011). IEEE, 2011. P. 292–299. DOI: 10.1109/DASC.2011.47.

72. Thakkar A., Badsha S., Sengupta S. Game theoretic approach applied in cybersecurity information exchange framework. *2020 IEEE Consumer Communications & Networking Conference (CCNC)* (Las Vegas, NV, USA, Jan. 10–13, 2020). IEEE, 2020. P. 1–6. DOI: 10.1109/CCNC46108.2020.9045430.

73. Verma R., Koul S., Ajaygopal K. V., Singh S. Exploring Game Theoretic Applications in Cyber Security. *2024 International Symposium on Cyber Security (ISCS)* (Dubai, UAE, May 14–15, 2024). IEEE, 2024. P. 1–7. DOI: 10.1109/ISCS61804.2024.10581244.

74. Pratomo B. A., Haykal M. F., Studiawan H., Purwitasari D. Graph-Structured Network Traffic Modelling for Anomaly-Based Intrusion Detection. *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*. 2025. Vol. 14, no. 2. P. 15–26. DOI: 10.23887/janapati.v14i2.94959.

75. Mora-Gimeno F. J., Mora-Mora H., Volckaert B., Atrey A. Intrusion Detection System Based on Integrated System Calls Graph and Neural Networks. *IEEE Access*. 2021. Vol. 9. P. 21672–21689. DOI: 10.1109/ACCESS.2021.3049249.

76. Abu Lailaa D., Basheer S., Alojail M., Bamasag O. Deep learning-driven multi-layer intrusion detection and prevention framework for resilient defense against adaptive evasion techniques in modern networks. *International Journal of Data and Network Science*. 2025. Vol. 9, no. 1. P. 1–16. DOI: 10.5267/j.ijdns.2025.10.014.

77. Singh M. T., Borkotokey S., Lahcen R. A. M., Mohapatra R. N. A generic scheme for cyber security in resource constraint network using incomplete information game. *Evolutionary Intelligence*. 2022. Vol. 15. P. 2195–2213. DOI: 10.1007/s12065-021-00684-w.

78. Gao L., Sun J., Li J. Security of Networked Control Systems with Incomplete Information Based on Game Theory. *2020 39th Chinese Control Conference (CCC)* (Shenyang, China, July 27–29, 2020). IEEE, 2020. P. 4668–4673. DOI: 10.23919/CCC50068.2020.9189235.

79. Franke U., Andreasson A., Artman H., et al. Cyber situational awareness issues and challenges. *Cognitive Informatics and Situational Awareness in Smart and Complex Systems*. Elsevier, 2022. P. 343–364. DOI: 10.1016/B978-0-323-90570-1.00015-2.

80. Chunlei W., Qing M., Yiqi D. Network Survivability Analysis Based on Stochastic Game Model. *2012 Fourth International Conference on Multimedia Information Networking and Security* (Nanjing, China, Nov. 2–4, 2012). IEEE, 2012. P. 660–663. DOI: 10.1109/MINES.2012.147.

81. Goyal M., Kumar R. The Pipeline Process of Signature-based and Behavior-based Malware Detection. *2020 5th International Conference on Computing, Communication and Automation (ICCCA)* (Greater Noida, India, Oct. 30–31, 2020). IEEE, 2020. P. 680–684. DOI: 10.1109/ICCCA49541.2020.9250879.

ДОДАТОК А

Список праць

Міністерство освіти і науки України
Хмельницький національний університет

ЗБІРНИК НАУКОВИХ ПРАЦЬ
за матеріалами XVII Всеукраїнської науково-практичної конференції
«Актуальні проблеми комп'ютерних наук АПКН-2025»

14-15 листопада 2025

АКТУАЛЬНІ ПРОБЛЕМИ КОМП'ЮТЕРНИХ НАУК - 2025

XVII Всеукраїнська науково-практична конференція

Метою конференції є висвітлення актуальних проблем комп'ютерних наук, інформатики та інформаційних технологій.

Робочі мови конференції:

українська, англійська

СЕКЦІЇ КОНФЕРЕНЦІЇ:

1. Комп'ютерні науки, штучний інтелект та прикладні інформаційні технології.
2. Комп'ютерна інженерія та системи захисту інформації.
3. Математичне моделювання та інженерія програмного забезпечення
4. Телерадіокомунікації, медійні та комунікаційні системи.
5. Проблеми впровадження інформаційних технологій у виробництво та управління.

СПИСОК ОРГАНІЗАЦІЙ, ПРЕДСТАВНИКИ ЯКИХ БРАЛИ УЧАСТЬ У РОБОТІ КОНФЕРЕНЦІЇ:

Донбаська державна машинобудівна академія
Інститут кібернетики імені В. М. Глушкова НАН України
Кам'янський енергетичний фаховий коледж
Київський національний університет імені Т. Г. Шевченка
Національного аерокосмічного університету імені М. Є. Жуковського «Харківський авіаційний інститут»
Національний технічний університет «Харківський політехнічний інститут
Сумський державний університет
Харківський національний університет радіоелектроніки
Хмельницький національний університет
Хмельницький фаховий економіко-технологічний коледж УЕП

ОРГКОМІТЕТ КОНФЕРЕНЦІЇ:

СИНЮК О. М. – голова оргкомітету, проректор Хмельницького національного університету з наукової роботи, доктор технічних наук, професор.

ГОВОРУЩЕНКО Т. О. – заступник голови оргкомітету, декан факультету інформаційних технологій Хмельницького національного університету, доктор технічних наук, професор.

БАРМАК О. В. – заступник голови оргкомітету, завідувач кафедри комп'ютерних наук Хмельницького національного університету, доктор технічних наук, професор.

САВЕНКО О. С. – професор кафедри комп'ютерної інженерії та інформаційних систем Хмельницького національного університету, доктор технічних наук, професор.

ВИСОЦЬКА О. В. – завідувач кафедри радіоелектронних та біомедичних комп'ютеризованих засобів і технологій Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут», доктор технічних наук, професор.

ЛАВРОВ Є. А. – доктор технічних наук, професор (Сумський державний університет).

ТІМОФЄЄВА Л. В. – відповідальна за студентську науково-дослідну роботу ХНУ.

МАЗУРЕЦЬ О. В. – секретар конференції, доцент кафедри комп'ютерних наук Хмельницького національного університету, кандидат технічних наук, доцент.

МОЛЧАНОВА М. О. – секретар конференції, старший викладач кафедри комп'ютерних наук Хмельницького національного університету, доктор філософії з комп'ютерних наук.

КОНТАКТНА ІНФОРМАЦІЯ:

e-mail для листування: *apkt.khnu@gmail.com*

УДК 004.6

Ратушняк М.В., Рябчук І.С, Муляр І.В.

Хмельницький національний університет

АНАЛІЗ УПРАВЛІННЯ РЕСУРСАМИ КІБЕРЗАХИСТУ В УМОВАХ НЕВИЗНАЧЕНОСТІ ТА АСИМЕТРІЇ ІНФОРМАЦІЇ ШЛЯХОМ АДАПТИВНОЇ ПРІОРИТЕЗАЦІЇ ЗАХОДІВ ЗАХИСТУ

Розглянуто розробку, теоретичне обґрунтування та експериментальну перевірку методу, що дозволяє системі захисту приймати оптимальні рішення щодо розподілу ресурсів в умовах стратегічної протидії та інформаційної невизначеності.

Considered the development, theoretical justification, and experimental verification of a method that allows the defense system to make optimal decisions regarding resource allocation under conditions of strategic opposition and informational uncertainty.

Сучасний ландшафт кіберзагроз характеризується не просто зростанням їхньої кількості, а якісною еволюцією від масових, автоматизованих атак до цілеспрямованих, багатоетапних кампаній, відомих як Advanced Persistent Threats (APTs). Ці кампанії проводяться раціональними супротивниками з високою мотивацією, які адаптують свою тактику в реальному часі, щоб обійти існуючі засоби захисту. Професіоналізація кіберзлочинності, що виражається в моделях Ransomware-as-a-Service (RaaS) та атаках на ланцюги постачання, створює фундаментальну асиметрію: зловмиснику достатньо знайти одну вразливість, тоді як захисник повинен захищати весь периметр та всі вектори атак.

Ця ситуація оголює ключову слабкість традиційних систем безпеки, які функціонують на основі статичних правил, сигнатурного аналізу та реактивних механізмів. Вони виявляються неефективними проти атак "нульового дня" та не здатні протистояти супротивнику, який стратегічно мислить, маскує свої дії та вводить в оману системи моніторингу. Як наслідок, організації зазнають колосальних фінансових та репутаційних втрат.

Таким чином, виникає гостра науково-практична проблема в розробці та обґрунтуванні методів управління кібербезпекою, які здатні функціонувати в умовах стратегічної невизначеності та асиметрії інформації. Необхідно здійснити парадигмальний зсув від реактивного реагування на інциденти до проактивного, керованого ризиками та адаптивного захисту, здатного передбачати дії раціонального супротивника.

Аналіз існуючих підходів до вирішення цієї проблеми виявив їхні суттєві обмеження. Статистичні моделі корисні для аналізу масових атак, вони не враховують ключового фактору – інтенціональності (намірів) та креативності зловмисника. Вони моделюють загрози як стохастичні процеси, ігноруючи стратегічне планування та адаптацію з боку супротивника.

Якісні фреймворки (MITRE ATT&CK, Cyber Kill Chain) надають неоціненну таксономію технік та тактик атак, створюючи загальну мову для опису інцидентів. Однак вони є дескриптивними, а не прескриптивними. Вони допомагають зрозуміти, що сталося, але не пропонують формального математичного апарату для вибору оптимальної контрстратегії в умовах обмежених ресурсів.

Системи на основі машинного навчання ефективні для виявлення аномалій та відомих патернів у великих обсягах даних. Проте, вони вразливі до змагальних атак (adversarial attacks), таких як "ухилення" (evasion) та "отруєння даних" (data poisoning), де зловмисник цілеспрямовано маніпулює вхідними даними, щоб обійти модель.

Платформи SOAR автоматизують стандартні процедури реагування, але їхня ефективність обмежується жорсткістю заздалегідь визначених сценаріїв (плейбуків). Вони не здатні приймати оптимальні рішення в умовах нових, непередбачуваних векторів атак.

Теорія ігор у кібербезпеці – хоча існують дослідження, що застосовують теорію ігор, багато з них обмежуються статичними (однокроковими) іграми або припускають наявність повної інформації та загальновідому раціональність гравців, що є нереалістичним для справжніх кіберконфліктів [1, 3].

Цей аналіз виявляє ключову прогалину в дослідженнях: відсутність інтегрованого, обчислювально ефективного методу, який би поєднував динаміку багатоетапного протистояння, невизначеність щодо типу та намірів зловмисника, та здатність знаходити оптимальну стратегію захисту в реальному часі.

Метою дослідження є підвищення кіберстійкості інформаційної системи шляхом розробки та наукового обґрунтування методу динамічного адаптивного управління безпековими ресурсами. Метод має базуватися на синергії апарату стохастичних байєсівських ігор для моделювання невизначеності та навчання з підкріпленням для обчислення оптимальної політики захисту.

Для досягнення поставленої мети розроблено комплексний метод, наукова новизна якого полягає в інтеграції двох потужних парадигм. Математична модель – динамічна байєсівська гра. Кіберпротистояння формалізується як послідовна гра з неповною інформацією. На відміну від класичних моделей, тут захисник не знає точного "типу" атакуючого (вектора його характеристик: рівень кваліфікації, наявність 0-day експлойтів, фінансування, схильність до ризику). Замість цього, захисник підтримує ймовірнісний розподіл ("belief state") щодо можливих типів супротивника. Після кожної дії атакуючого (наприклад, спроби сканування певного порту) захисник, використовуючи правило Байєса, оновлює свої уявлення, уточнюючи оцінку того, з ким він має справу.

Обчислювальний алгоритм – навчання з підкріпленням (Q-learning). Оскільки простір станів та дій у реалістичних сценаріях є надзвичайно великим, аналітичне рішення такої гри є неможливим. Тому запропоновано використовувати навчання з підкріпленням (Reinforcement Learning, RL). Агент (захисник) навчається оптимальній політиці π^*, яка кожному стану системи (включаючи поточні уявлення про тип атакуючого) ставить у відповідність оптимальну дію (наприклад, перерозподіл обчислювальних потужностей на аналіз трафіку, ізоляцію сегмента

мережі, розгортання приманок). RL дозволяє знаходити немиопічну (далекоглядну) стратегію, максимізуючи сукупну довгострокову винагороду (мінімізуючи збитки) без необхідності мати явну модель поведінки супротивника [2].

Синергія підходу полягає в тому, що байєсівська модель надає формальну структуру для роботи з невизначеністю, а RL – масштабований обчислювальний інструмент для пошуку оптимальної поведінки в цій структурі.

Ефективність методу було перевірено на програмному симуляторі, що імітував корпоративну мережу з різнорідними активами різної цінності. Моделювалися атаки від супротивників з різними стратегічними профілями. Порівняльний аналіз показав, що запропонований адаптивний метод, у порівнянні зі статичними політиками та реактивними стратегіями, дозволив знизити сукупні очікувані збитки та підвищити коефіцієнт повернення інвестицій у безпеку (ROSI) за рахунок раціонального, обґрунтованого розподілу ресурсів захисту в кожен момент часу.

У дослідженні вирішено актуальну науково-прикладну задачу розробки методу адаптивного управління кіберзахистом в умовах стратегічної протидії. Запропонований метод, що поєднує строгість теорії байєсівських ігор та обчислювальну потужність навчання з підкріпленням, є кроком до створення нового покоління автономних систем кіберзахисту. Він дозволяє перейти від пасивного, заснованого на сигнатурах захисту до динамічного, інтелектуального та стратегічно обґрунтованого процесу забезпечення безпеки.

Подальший розвиток даного дослідження можливий у наступних напрямках:

– дослідження застосування алгоритмів глибокого навчання з підкріпленням (Deep Q-Networks, Actor-Critic) для подолання "прокляття розмірності" у великомасштабних гетерогенних мережах.

– інтеграція моделей з поведінкової економіки для врахування обмеженої раціональності та когнітивних упереджень зловмисників.

– використання даних з платформ аналізу загроз (TIPs) для формування початкових апріорних ймовірностей щодо типів атакуючих.

– розробка механізмів інтерпретації рішень, що приймаються системою, для підвищення довіри та ефективної взаємодії з аналітиками Security Operation Center (SOC).

Перелік посилань

1. Wang, J., & Cui, L. (2023). Patrolling games with coordination between monitoring devices and patrols. Reliability Engineering & System Safety, 233, 109109. DOI: 10.1016/j.ress.2023.109109.

2. McDonald, G., Li, L., & Al Mallah, R. (2024). Finding the Optimal Security Policies for Autonomous Cyber Operations With Competitive Reinforcement Learning. IEEE Access, PP(99), 1–1. DOI: 10.1109/ACCESS.2024.3446310.

3. Zhu, M., Anwar, A., & Wan, Z. (2021). Game-Theoretic and Machine Learning-based Approaches for Defensive Deception: A Survey. arXiv preprint arXiv:2101.10121. DOI: 10.48550/arXiv.2101.10121

УДК 004.056

ВОЛОДИМИР ДЖУЛІЙ
Хмельницький національний університет
ORCID http://orcid.org/0000-0003-1878-4301
e-mail: dzhuliivm@khmnu.edu.ua

ІГОР МУЛЯР
Хмельницький національний університет
ORCID http://orcid.org/0000-0002-6659-605X
e-mail: muliariv@khmnu.edu.ua

МАКСИМ РАТУШНЯК
Хмельницький національний університет
https://orcid.org/0009-0005-8083-122X
e-mail: ratsuhnyak@gmail.com

ВІКТОР ЧЕШУН
Хмельницький національний університет
ORCID https://orcid.org/0000-0002-3935-2068
e-mail: cheshunvn@khmnu.edu.ua

АДАПТИВНЕ УПРАВЛІННЯ РЕСУРСАМИ КОМПЛЕКСНОЇ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЇ НА ОСНОВІ СИНТЕЗУ ТЕОРІЇ ІГОР ТА ПОСИЛЕНОГО НАВЧАННЯ

У статті розроблено та теоретично обґрунтовано метод адаптивного управління ресурсами кіберзахисту, що базується на поєднанні підходів динамічних Баєсівських ігор та посиленого навчання. Цей метод моделює протистояння між раціональним захисником та нападником в умовах, коли захисник має неповну інформацію про зловмисника. Невизначеність щодо рівня кваліфікації чи мотивації нападника формалізується через апріорні ймовірнісні припущення про його прихований тип.

Запропонований метод функціонує як безперервний цикл, що складається з моніторингу, адаптації та прийняття рішень. Ключовим елементом є механізм адаптації, який використовує принцип Баєса для коригування ймовірнісних припущень про тип зловмисника щоразу, коли спостерігається його дія. Для розрахунку найкращої довгострокової стратегії захисника (що мінімізує сукупні витрати) застосовується алгоритм посиленого навчання (Q-learning), який обчислює Байєс-Нешівську рівновагу.

Доведено, що цей динамічний та проактивний підхід значно ефективніший за статичні чи реактивні методи, забезпечуючи глобальну мінімізацію очікуваних витрат. Метод має практичне значення для розробки інтелектуальних систем підтримки прийняття рішень (СППР) та легко інтегрується в наявні системи безпеки, такі як Комплексні системи захисту інформації (КСЗІ) та Системи контролювання доступу (СКД).

Ключові слова: динамічні баєсівські ігри, посилене навчання (Q-learning), адаптивне управління кіберзахистом, теорія ігор, асиметрія інформації, комплексна система захисту інформації, система контролювання доступу, оптимізація ресурсів

VOLODYMYR DZHULIY, IHOR MULIAR , MAKSYM RATUSHNYAK,VIKTOR CHESHUN
Khmelnitsky national university

ADAPTIVE MANAGEMENT OF RESOURCES OF A COMPLEX INFORMATION PROTECTION SYSTEM BASED ON THE SYNTHESIS OF GAMES THEORY AND REINFORCED LEARNING

The article develops and theoretically substantiates a method of adaptive cyber defense resource management based on a combination of dynamic Bayesian games and reinforcement learning approaches. This method models the confrontation between a rational defender and an attacker under conditions where the defender has incomplete information about the adversary. Uncertainty regarding the attacker's level of skill or motivation is formalized through prior probabilistic assumptions about the attacker's hidden type.

The proposed method operates not as a one-time calculation but as a continuous, iterative cycle consisting of

monitoring, adaptation, and decision-making phases. A key element of this research is the dynamic adaptation mechanism, which employs the Bayesian principle to update and adjust probabilistic assumptions about the adversary's type each time a specific attack action is observed. This allows the system to refine its understanding of the threat landscape in real-time. However, solving such complex dynamic games analytically is computationally prohibitive. Therefore, to compute the optimal long-term strategy of the defender—specifically, the strategy that minimizes cumulative costs associated with both security implementation and potential damage—a reinforcement learning algorithm, specifically Q-learning, is used to approximate the Bayesian–Nash equilibrium. This allows the defense agent to learn the optimal policy through simulated interactions, balancing immediate defense costs against future risks.

It is theoretically proven that this dynamic and proactive approach is significantly more effective than traditional static or purely reactive methods. By anticipating rational attacker behavior and adapting to the attacker's type, the method ensures a global minimization of expected costs over the entire duration of the conflict. The method has substantial practical significance for the development of next-generation intelligent decision support systems (DSS) for Security Operations Centers (SOCs). Furthermore, the algorithmic nature of the proposed solution allows it to be easily integrated into existing security frameworks, such as Comprehensive Information Protection Systems (CIPS) and Access Control Systems (ACS), providing them with an intelligent core for automated resource allocation and strategic defense.

Keywords: dynamic Bayesian games, reinforcement learning (Q-learning), adaptive cyber defense management, game theory, information asymmetry, comprehensive information protection system (CIPS), access control system (ACS), resource optimization.

Постановка проблеми

Сьогодні кібератаки вже не є випадковими чи простими подіями. Вони стали продуманими, стратегічними операціями, які проводять кваліфіковані люди, що намагаються максимально збільшити свій прибуток чи досягти своїх цілей [1, 2]. Це означає, що захист має справу не просто з технічною помилкою, а з розумним, адаптивним супротивником [3].

Сьогоднішні підходи до побудови КСЗІ мають кілька ключових обмежень, які ми прагнемо подолати. Головна проблема полягає в тому, що більшість наявних інструментів створені для статичного або реактивного захисту проти динамічного та розумного супротивника. Наприклад, системи, що базуються на машинному навчанні, дуже ефективні у виявленні вже відомих патернів, але вони легко обходяться, коли зловмисник цілеспрямовано змінює вхідні дані [4]. Крім того, якісні моделі, такі як матриця ATT&CK, чудово структурують інформацію, але вони не дають жодного кількісного механізму для розрахунку ризику чи визначення оптимального розподілу обмежених ресурсів. Водночас, наявні теоретико-ігрові моделі, які могли б розв'язати проблему стратегічної протидії, часто надто спрощені; вони припускають, що захисник має повну інформацію про можливості та наміри зловмисника, що є нереалістичним в умовах реального кіберпростору [5].

Зараз відбувається реформа у сфері національної кібербезпеки. Однією з найбільш значущих трансформацій є стратегічний відхід від застарілої моделі комплексної системи захисту інформації, яка скомпрометувала себе, до нової філософії захисту, заснованої на міжнародних практиках – авторизації систем з безпеки. Цей підхід є більш гнучким та ефективним, базується на управлінні кіберризиками, розробці та регулярному оцінюванні профілів безпеки.

Системи автоматизації (SOAR) можуть лише виконувати заздалегідь написані сценарії, але не здатні адаптивно приймати стратегічні рішення в ситуації непередбачуваної невизначеності [6].

Таким чином, головна науково-практична проблема, яку ми прагнемо вирішити, полягає в наступному: як створити такий механізм захисту, який міг би динамічно, у режимі реального часу розподіляти свої обмежені ресурси, враховуючи, що дії нашого захисту, своєю чергою, впливають на наступні дії зловмисника [7].

Ми вважаємо, що розв'язання цієї проблеми, яка лежить на перетині кібербезпеки та стратегічного конфлікту, дозволить перейти до по-справжньому проактивного управління безпекою та забезпечити найбільшу віддачу від інвестицій у захист (ROSI) [7].

Аналіз останніх джерел

Аналіз останніх наукових джерел та публікацій показує, що наявні методики побудови КСЗІ мають обмежену ефективність у протидії адаптивному та стратегічному супротивнику [4, 5]. Ми бачимо, що системи,

засновані на машинному навчанні та статистичних моделях, є високоточними для виявлення вже відомих загроз і патернів аномалій. Однак, ми констатуємо їхню фундаментальну нездатність передбачати наступний невідомий крок раціонального зловмисника, а також їхню вразливість до змагальних атак, де супротивник цілеспрямовано маніпулює даними для обходу детектора [8]. Жоден з цих методів не пропонує механізму для стратегічного прийняття рішень.

Водночас ми спираємося на роботи, що використовують теорію ігор для моделювання конфліктів у кіберпросторі. Проте більшість із них описують статичні ситуації або припускають, що захисник має повну інформацію про можливості та мотиви зловмисника [5, 9]. У реальному світі це не так. Якісні фреймворки, такі як матриця ATT&CK, є чудовими інструментами для структурування знань про атаки, але вони не надають кількісного апарату для розрахунку ризику та оптимального розподілу обмежених ресурсів [4, 10].

Таким чином, невирішеною частиною проблеми, якій присвячена наша стаття, є синтез цих підходів. Нам бракує цілісного методу, який би поєднував динамічне моделювання, можливість приймати рішення в умовах неповної інформації про супротивника та обчислювально-ефективний алгоритм для знаходження найкращої стратегії захисту в реальному часі [9, 11].

Формулювання цілей

Головна мета нашої статті полягає в тому, щоб підвищити ефективність захисту інформаційних систем. Ми прагнемо розробити та обґрунтувати новий, адаптивний метод, який дозволить системі безпеки динамічно керувати своїми обмеженими ресурсами [8, 12]. Цей метод повинен самостійно пристосовуватися до того, як змінюється стратегія нашого супротивника, і до того, який поточний стан захищеності системи [12, 13]. Для цього ми використовуємо математичний апарат, що поєднує теорію ігор та посилене навчання [9, 14].

Щоб досягти цієї головної мети, ми поставили перед собою кілька конкретних завдань. По-перше, нам необхідно створити точну математичну модель конфлікту між захисником та зловмисником, яка обов'язково повинна враховувати динаміку їхніх послідовних дій та факт, що захисник не має повної інформації про свого супротивника [14,15]. По-друге, ми повинні розробити метод адаптації цієї моделі, який дозволить системі вчитися на спостереженнях [3]. І, нарешті, по-третє, нам потрібно сформулювати обчислювальний алгоритм, який на основі цієї адаптивної моделі зможе швидко розраховувати найкращу стратегію розподілу ресурсів для захисника в будь-який момент часу [6,13].

Виклад основного матеріалу

Математична модель динамічної Баєсівської гри

Щоб формально описати процес протистояння, ми визначаємо його як динамічну гру. У цій грі є два раціональні гравці захисник (D) та нападник (A) [4, 9].

Поточний стан нашої інформаційної системи в будь-який момент часу описується як стан (s) із множини всіх можливих станів (S). Стан може містити, наприклад, наявність певних вразливостей або статус критичних сервісів.

На кожному етапі гри обидва гравці обирають дії зі своїх наборів. Захисник обирає захисну дію (a_D), а нападник обирає атакувальну (a_A). Оскільки гра динамічна, ці спільні дії призводять до зміни стану системи. Ми моделюємо це за допомогою функції переходу, яка визначає ймовірність того, що система перейде у новий стан s', виходячи з поточного стану s та обраних гравцями дій [10].

Функція переходу, що відображає динаміку системи, має вигляд:

$$T(\mathrm{s}'|\mathrm{s}, \mathrm{a_D}, \mathrm{a_A}), \quad (1)$$

Ця формула (1) є ключовою для опису стохастичної природи гри та є математичним відображенням того, як спільні дії гравців змінюють стан інформаційної системи [15].

Ключовою особливістю нашої моделі є асиметрія інформації, що робить її Баєсівською. У реальному світі захисник ніколи не знає напевно, з ким має справу [9]. Ми формалізуємо цю невизначеність шляхом введення поняття "типу" зловмисника (θ) із множини $\boldsymbol{\theta}$. "Тип" це прихована характеристика нападника, його рівень кваліфікації, доступні ресурси або мотивація [16]. На рис. 1 зображено модель гри:

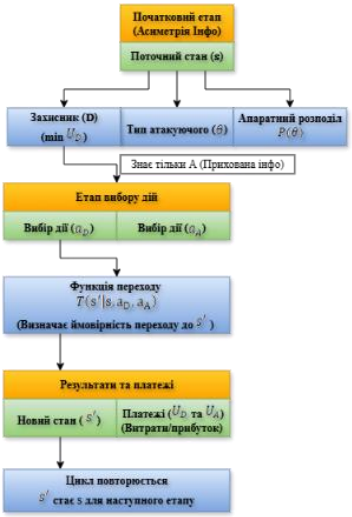

Рис. 1. Модель Динамічної Баєсівської Гри

Нападник знає свій тип, але захисник ні. Захисник має лише початкове припущення про те, з ким він зіткнувся.

Це припущення ми описуємо як апріорний розподіл ймовірностей:

$$P(\theta)$$

Цей розподіл показує, наскільки ймовірним захисник вважає кожен можливий тип зловмисника до початку активної взаємодії [9].

Щоб модель могла розраховувати оптимальну поведінку, ми повинні визначити мотивацію гравців через платіжні функції (U_D та U_A).

Функція захисника (U_D) це, по суті, функція сукупних витрат, яку він прагне мінімізувати. Вона складається з прямих витрат на захисну дію та очікуваної шкоди від успішної атаки [5]. Навпаки, функція нападника (U_A) це функція прибутку, яку він прагне максимізувати. Його "виграш" залежить від цінності активу та ймовірності успіху атаки, мінус його власні витрати на проведення атаки [7, 9]. На рисунку 1 зображено модель гри:

Важливо, що ймовірність успіху не є сталою: вона динамічно залежить від поточного стану системи, обраної нападником атаки та обраної захисником контрдії [17].

Метод адаптивного управління ресурсами

Маючи математичну модель, ми розробили практичний метод адаптивного управління, який працює як безперервний цикл [11, 13]. Цей цикл складається з трьох ключових фаз:

1. Моніторинг де система збирає дані про стан мережі та дії зловмисника;
2. Адаптація де система оновлює свої уявлення про загрозу.
3. Прийняття рішень де система розраховує найкращу контрдію [17].

Ця циклічна структура дозволяє нашому методу постійно пристосовуватися до мінливої ситуації, а не діяти за статичним планом[12].

Найважливішою частиною методу є механізм адаптації. Коли ми спостерігаємо якусь дію зловмисника (E) наприклад, він використовує складну zero-day вразливість, це дає нам нову інформацію. Ми використовуємо теорему Баєса, щоб оновити наші початкові ймовірнісні припущення $P(\theta)$ про його тип [14,16].

Теорема Баєса для оновлення апріорного розподілу виглядає так:

$$F(\theta|E) \infty P(E|\theta)P(\theta), \quad (2)$$

Згідно з формулою (2), якщо просту атаку міг би провести новачок, то складна атака (E) значно підвищує ймовірність $F(\theta|E)$, що ми маємо справу з APT-угрупованням [16]. Таким чином, наша модель навчається на діях зловмисника, стаючи точнішою з кожним кроком [13, 18].

Після того як модель адаптувалася, настає фаза прийняття рішень. Наша мета це знайти найкращу довгострокову стратегію для Захисника, яка в теорії ігор називається Байєс-Нешівською рівновагою [9, 14]. Цикл адаптивного управління представлено на рис. 2:

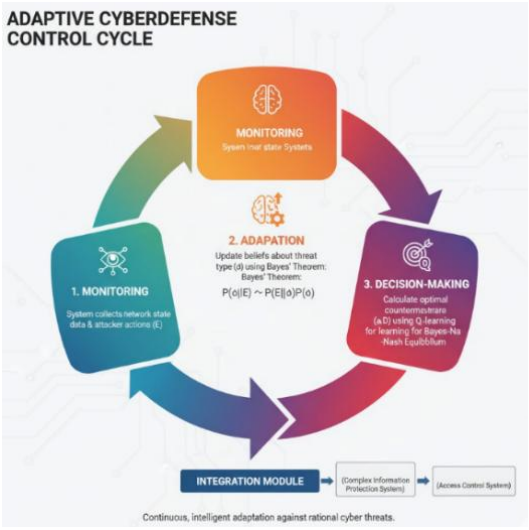

Рис. 2. Цикл адаптивного управління

Оскільки наша гра є складною, динамічною та з неповною інформацією, знайти це рішення аналітично (тобто, "на папері" за допомогою формул) практично неможливо. Тому ми обґрунтовуємо використання алгоритму посиленого навчання Q-learning [11, 19]. Цей чисельний метод дозволяє нашому захиснику "пограти" мільйони разів проти симульованих зловмисників різних типів і поступово, шляхом спроб та помилок, навчитися, яка дія (a_D) приносить найкращий сукупний результат у кожному конкретному стані (s) [12, 18]. Фактично, Q-learning обчислює для нас цю оптимальну стратегію, роблячи теоретичну модель практично придатною для вирішення реальних завдань.

Фактично, Q-learning обчислює для нас цю оптимальну стратегію, роблячи теоретичну модель практично придатною для вирішення реальних завдань.

Цей механізм прийняття рішень може бути інтегрований як керівний модуль в наявній системі безпеки, наприклад, в Комплексну систему захисту інформації (КСЗІ) або Систему контролювання доступу (СКД), для автоматизації та оптимізації захисних реакцій [6].

Обґрунтування наукових результатів

Наукові результати, отримані в ході нашого дослідження, полягають не лише в розробці самої моделі, але й у теоретичному обґрунтуванні її ефективності порівняно з традиційними підходами до захисту [8, 11]. Запропонований ми синтез теорії ігор та посиленого навчання (RL) має фундаментальні переваги над статичними чи простими реактивними стратегіями [13, 19].

По-перше, статичні стратегії (наприклад, одноразовий розподіл ресурсів на основі початкової оцінки ризиків) є за своєю природою крихкими. Вони ефективні лише доти, доки раціональний зловмисник не знайде в них слабке місце. Як тільки він адаптується, така статична оборона стає передбачуваною і легко експлуатується [5, 8]. Наш метод, навпаки, є динамічним і проактивним: він не просто реагує, а передбачає найбільш імовірні раціональні дії зловмисника, постійно змінюючи стратегію захисту, щоб зробити експлуатацію невигідною [11, 14].

По-друге, періодичні або суто реактивні стратегії (наприклад, зміна правил після інциденту) завжди запізнюються. Вони діють після того, як шкода вже завдана, або оновлюються через фіксовані інтервали, що дозволяє зловмиснику діяти у "вікнах можливостей" [2, 5]. Наш метод, завдяки баєсівському оновленню, є адаптивним у реальному часі. Він коригує свою стратегію не лише після інциденту, але й на основі будь-якої спостережуваної дії, що дозволяє виявляти наміри зловмисника на ранніх стадіях і запобігати шкоді [16, 18].

Таким чином, ми показуємо, що розроблений метод забезпечує мінімальні очікувані сукупні витрати для

захисника в довгостроковій перспективі [11, 19]. Це досягається завдяки тому, що алгоритм Q-learning за своєю математичною суттю шукає оптимальну політику, яка максимізує сукупну дисконтовану винагороду (або мінімізує сукупні витрати). На відміну від статичних методів, що оптимізують захист лише для початкового моменту часу, і реактивних, що мінімізують лише поточні втрати, наш метод знаходить оптимальний баланс між поточними витратами на захист та майбутніми очікуваними збитками, що й призводить до глобальної, а не локальної, мінімізації витрат протягом усього життєвого циклу протистояння [12, 19].

Після створення математичної моделі та реалізації алгоритму Q-learning ми перевірили, як наш метод працює на практиці.

Для цього провели серію симуляцій і порівняли три підходи до захисту:

1. Статичну стратегію, де правила не змінюються;
2. Реактивну, яка реагує тільки після атаки;
3. Адаптивну (RL стратегію), що навчається і підлаштовується під дії зловмисника.

На рис. 3 показано, як змінювалися середні втрати системи під час симуляції:

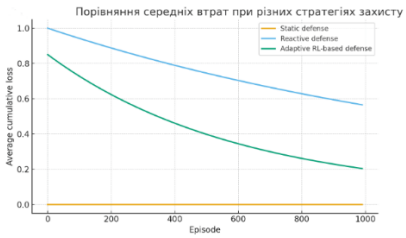

Рис. 3. Порівняння середніх втрат при різних стратегіях захисту

Як видно, статична стратегія постійно має високі втрати, реактивна поступово знижує їх, а адаптивна стратегія RL швидко навчається і стабілізує втрати на мінімальному рівні. Це підтверджує, що модель з посиленим навчанням дійсно «вчиться» на досвіді й з часом приймає кращі рішення, що допомагає зменшити загальні втрати.

Далі ми перевірили, як швидко система пристосовується до нових типів атак.

На рис. 4 видно, скільки ітерацій потрібно кожній стратегії, щоб відновити стабільну роботу після появи нової загрози:

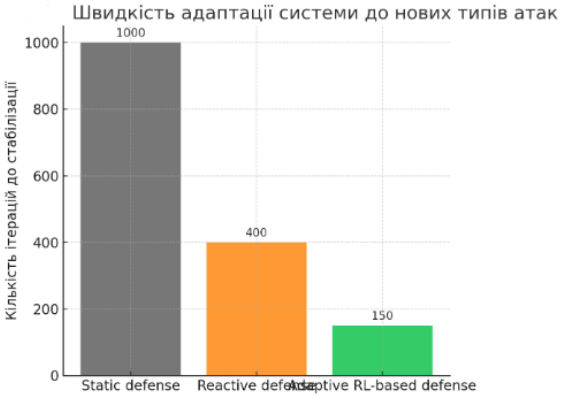

Рис. 4. Швидкість адаптації системи до нових типів атак

Статичний захист взагалі не змінюється, реактивний потребує багато часу, а адаптивна RL модель стабілізується приблизно втричі швидше. Це означає, що вона ефективно оновлює свої уявлення про зловмисника і швидко підлаштовується до нових умов.

На рис. 5 показано баланс між витратами на захист і збитками від атак:

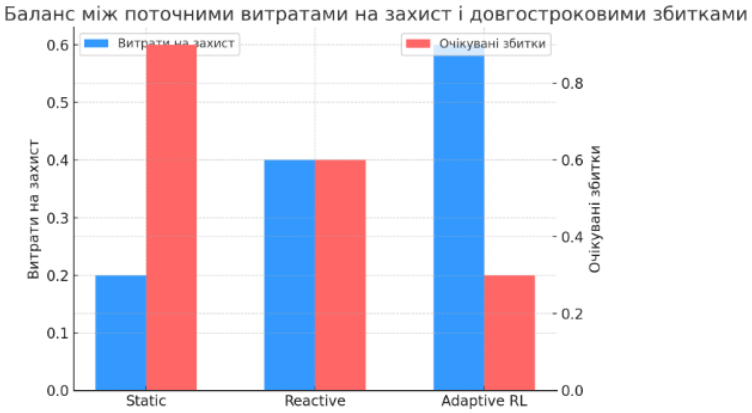

Рис. 5. Баланс між поточними витратами на захист і довгостроковими збитками

Можна побачити, що адаптивна стратегія потребує трохи більше ресурсів на початку, але завдяки цьому знижує загальні збитки в майбутньому.

Тобто, система витрачає трохи більше зараз, щоб потім уникнути великих втрат.

Загалом отримані результати показують, що адаптивний метод на основі Q-learning є найбільш ефективним. Він швидко навчається, стабільно реагує на зміни в поведінці зловмисників і забезпечує оптимальне співвідношення між витратами на захист і рівнем безпеки системи.

Висновки з даного дослідження і перспективи подальшого розвитку у даному напрямку

У цій роботі ми досягли поставленої мети, а саме розробили та теоретично обґрунтували метод адаптивного управління ресурсами кіберзахисту. Запропонований синтез динамічних баєсівських ігор та посиленого навчання забезпечує значно вищу ефективність порівняно з традиційними статичними чи реактивними підходами. Перевага нашого методу полягає в його адаптивності, оскільки він оновлює свої припущення про зловмисника на основі його дій, та в стратегічній обґрунтованості рішень, що мінімізують очікувані сукупні витрати в довгостроковій перспективі.

Практичне значення нашої роботи полягає в тому, що розроблений метод може слугувати теоретичною основою та ядром для створення інтелектуальних СППР. Такі системи здатні надавати аналітикам у центрах безпеки (SOC) обґрунтовані рекомендації. Цей метод може бути безпосередньо інтегрований як керівний модуль в наявні КСЗІ або СКД, автоматизуючи процес оптимального розподілу ресурсів захисту в умовах активної, раціональної кіберзагрози.

Ми бачимо три основні напрямки для перспектив подальшого розвитку:

1. Масштабування, адаптація моделі для її застосування у великих, гетерогенних корпоративних мережах;
2. Удосконалення алгоритмів, дослідження глибинного посиленого навчання (Deep Q-Networks) для роботи з величезним або неперервним простором станів;
3. Інтеграція даних, розробка механізмів інтеграції моделі з реальними даними про загрози (Threat Intelligence Platforms), що дозволить формувати початкові ймовірнісні оцінки ($P(\theta)$) на основі актуальної оперативної інформації.

Література

1. Edwards, B., Furnas, A., & Forrest, S. (2017). Strategic aspects of cyberattack, attribution, and blame. Proceedings of the National Academy of Sciences, 114(11), 2825–2830. https://doi.org/10.1073/pnas.1700442114

2. Gomez, M. A. (2022). Unpacking strategic behavior in cyberspace: A schema-driven approach. Cybersecurity, 5(1), 1–14. https://doi.org/10.1093/cybsec/tyac005

3. Lawson, S. (2019). Strategic Stability, Cyber Operations and International Security. Waterloo: Centre for International Governance Innovation.

4. Li, Y., Zhu, H., Zhang, H., & Chen, X. (2021). A Comprehensive Review Study of Cyber-Attacks and Cyber-Security Mechanisms. Amsterdam: Elsevier.

5. Xu, H., Zhao, D., Sandberg, H., & Johansson, K. H. (2017). A Game-Theoretic Approach for Intelligent

Allocation of Cyber Alerts. New York: ACM Press.

6. Hoffman, D., & Roman, E. (2020). Security Orchestration, Automation and Response (SOAR): Concepts and Challenges. Boston: Wiley.

7. August, T., & Nix, D. (2024). Cyberattacks, Operational Disruption, and Investment in Cyber Resilience. Chicago: University of Chicago Press.

8. Biggio, B., & Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. Pattern Recognition, 84, 317–331. https://doi.org/10.1016/j.patcog.2018.07.023

9. Zhu, Q., & Rass, S. (2018). *Game Theory Meets Network Security: A Tutorial.* In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS '18) (Article 4). New York, NY: ACM. https://doi.org/10.1145/3243734.3264421

10. Strom, B. E., et al. (2020). MITRE ATT&CK: Design and Philosophy. McLean (VA): The MITRE Corporation. Retrieved from https://attack.mitre.org/ (Accessed 01.11.2025).

11. Guo, Y., et al. (2021). Reinforcement-learning-based dynamic defense strategy against large-scale automated attacks. Applied Soft Computing, 110, 107560. https://doi.org/10.1016/j.asoc.2021.107560

12. Shah, P. D. (2024). Reinforcement Learning for Adaptive Cyber Defense: A Dynamic Approach to Threat Mitigation. International Meridian Journal, 6(6).

13. Hu, Z., Chen, P., Zhu, M., & Liu, P. (2019). Reinforcement Learning for Adaptive Cyber Defense Against Zero-Day Attacks. In Lecture Notes in Computer Science (Vol. 11830, pp. 54–93). Berlin: Springer. https://doi.org/10.1007/978-3-030-30719-6_4

14. Elderman, R., Pater, L. J. J., Thie, A. S., Drugan, M. M., & Wiering, M. (2017). Adversarial Reinforcement Learning in a Cyber Security Simulation. University of Groningen. Retrieved from https://www.ai.rug.nl/~mwiering/GROUP/ARTICLES/CyberSec_ICAART.pdf (Accessed 01.11.2025).

15. He, X., & Dai, H. (2018). Dynamic Games for Network Security. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-75871-8

16. Huang, L., & Zhu, Q. (2018). Dynamic Bayesian Games for Adversarial and Defensive Cyber Deception. arXiv preprint arXiv:1809.02013. Retrieved from https://arxiv.org/abs/1809.02013 (Accessed 02.11.2025).

17. Chung, K., Farraj, S., et al. (2015). Game Theory with Learning for Cyber Security Monitoring. Retrieved from https://assured-cloud-computing.illinois.edu/files/2014/03/Game-Theory-with-Learning-for-Cyber-Security-Monitoring.pdf (Accessed 03.11.2025).

18. Lopes, A. F. N. (2021). Bayesian Reinforcement Learning Methods for Network Security (Doctoral dissertation). DIVA Portal. Retrieved from https://www.diva-portal.org/smash/get/diva2:1631269/FULLTEXT03.pdf (Accessed 03.11.2025).

19. Goel, D., Moore, K., Guo, M., Wang, D., Kim, M., & Camtepe, S. (2024). Optimizing Cyber Defense in Dynamic Active Directories through Reinforcement Learning. arXiv preprint arXiv:2406.19596. Retrieved from https://arxiv.org/abs/2406.19596 (Accessed 07.11.2025).

References

1. Edwards, B., Furnas, A., & Forrest, S. (2017). Strategic aspects of cyberattack, attribution, and blame. *Proceedings of the National Academy of Sciences*, 114(11), 2825–2830. https://doi.org/10.1073/pnas.1700442114

2. Gomez, M. A. (2022). Unpacking strategic behavior in cyberspace: A schema-driven approach. *Cybersecurity*, 5(1), 1–14. https://doi.org/10.1093/cybsec/tyac005

3. Lawson, S. (2019). *Strategic Stability, Cyber Operations and International Security.* Waterloo: Centre for International Governance Innovation.

4. Li, Y., Zhu, H., Zhang, H., & Chen, X. (2021). *A Comprehensive Review Study of Cyber-Attacks and Cyber-Security Mechanisms.* Amsterdam: Elsevier.

5. Xu, H., Zhao, D., Sandberg, H., & Johansson, K. H. (2017). *A Game-Theoretic Approach for Intelligent Allocation of Cyber Alerts.* New York: ACM Press.

6. Hoffman, D., & Roman, E. (2020). *Security Orchestration, Automation and Response (SOAR): Concepts and Challenges.* Boston: Wiley.

7. August, T., & Nix, D. (2024). *Cyberattacks, Operational Disruption, and Investment in Cyber Resilience.* Chicago: University of Chicago Press.

8. Biggio, B., & Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84, 317–331. https://doi.org/10.1016/j.patcog.2018.07.023

9. Zhu, Q., & Rass, S. (2018). Game Theory Meets Network Security: A Tutorial. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS '18) (Article 4). New York, NY: ACM. https://doi.org/10.1145/3243734.3264421

10. Strom, B. E., et al. (2020). *MITRE ATT&CK: Design and Philosophy.* McLean (VA): The MITRE Corporation. Retrieved from https://attack.mitre.org/ (Accessed 01.11.2025)

11. Guo, Y., et al. (2021). Reinforcement-learning-based dynamic defense strategy against large-scale automated attacks. Applied Soft Computing, 110, 107560. https://doi.org/10.1016/j.asoc.2021.107560

12. Shah, P. D. (2024). Reinforcement learning for adaptive cyber defense: A dynamic approach to threat mitigation. *International Meridian Journal*, 6(6).

13. Hu, Z., Chen, P., Zhu, M., & Liu, P. (2019). Reinforcement learning for adaptive cyber defense against zero-day attacks. In *Lecture Notes in Computer Science* (Vol. 11830, pp. 54–93). Berlin: Springer. https://doi.org/10.1007/978-3-030-30719-6_4

14. Elderman, R., Pater, L. J. J., Thie, A. S., Drugan, M. M., & Wiering, M. (2017). Adversarial reinforcement learning in a cyber security simulation. *University of Groningen.* Retrieved from https://www.ai.rug.nl/~mwiering/GROUP/ARTICLES/CyberSec_ICAART.pdf (Accessed 01.11.2025)

15. He, X., & Dai, H. (2018). *Dynamic Games for Network Security.* Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-75871-8

16. Huang, L., & Zhu, Q. (2018). Dynamic Bayesian games for adversarial and defensive cyber deception. *arXiv preprint arXiv:1809.02013.* Retrieved from https://arxiv.org/abs/1809.02013 (Accessed 02.11.2025)

17. Chung, K., Farraj, S., et al. (2015). Game theory with learning for cyber security monitoring. Retrieved from https://assured-cloud-computing.illinois.edu/files/2014/03/Game-Theory-with-Learning-for-Cyber-Security-Monitoring.pdf (Accessed 03.11.2025)

18. Lopes, A. F. N. (2021). *Bayesian Reinforcement Learning Methods for Network Security* (Doctoral dissertation). DIVA Portal. Retrieved from https://www.diva-portal.org/smash/get/diva2:1631269/FULLTEXT03.pdf (Accessed 03.11.2025)

19. Goel, D., Moore, K., Guo, M., Wang, D., Kim, M., & Camtepe, S. (2024). Optimizing cyber defense in dynamic active directories through reinforcement learning. *arXiv preprint arXiv:2406.19596.* Retrieved from https://arxiv.org/abs/2406.19596 (Accessed 07.11.2025)

Завідувачу кафедри кібербезпеки
канд.техн.наук, доц. Кльоцу Ю.П.
здобувача вищої освіти
Ратушняк Максим Віталійович
студента ФІТ, 2 курсу, групи КБЗІм-24-1

ЗАЯВА

З правилами чинного Положення про систему забезпечення академічної доброчесності у Хмельницькому національному університеті, згідно з яким виявлення академічного плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту і застосування заходів академічної відповідальності, ознайомлений. Про використання спеціалізованих програмних засобів (СПЗ) StrikePlagiarism та Anti-Plagiarism для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність академічного плагіату оповіщений. Надаю університету право на передачу моєї роботи для обробки та збереження в базах даних СПЗ і використання роботи для виявлення академічного плагіату в інших роботах, які перевіряються СПЗ.

Також надаю свою згоду на обробку й збереження університетом моєї роботи в Інституційному репозитарії Хмельницького національного університету.

Робота надається для перевірки в електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

01.12.2025
дата

підпис

Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Ратушняк Максим Віталійович

Співавтор:

Назва: Метод адаптивного управління ресурсами захисту інформаційної системи в умовах динамічних загроз

Науковий керівник: Муляр Ігор Володимирович

Підрозділ: Кафедра кібербезпеки

Коефіцієнт подібності 1:1.7%

Коефіцієнт подібності 2:0.3%

Мікропробіли: 0

Заміна букв: 0

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2025-12-12 22:55:57.0

Після аналізу Звіту подібності констатую наступне:

☑ **Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.**

☐ **Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.**

☐ **Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.**

Обгрунтування:

Дата 15.12.2025р. експерт

Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 1.0%

Dictionaries check: en_US, ru_RU, ua_UA. **Errors in the documents: 7%**

ID: 252719 Title: Метод адаптивного управління ресурсами захисту інформаційної системи в умовах динамічних загроз Added in a DB: 2025-12-12 Authors: Ратушняк Максим Віталійович Heads: Муляр І.В. Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	105536	962	2418 (2%)	26 (3%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ КАФЕДРИ КІБЕРБЕЗПЕКИ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи: Метод адаптивного управління ресурсами захисту інформаційної системи в умовах динамічних загроз
Автор: Ратушняк Максим Віталійович
Освітня програма: Кібербезпека та захист інформації
Рівень вищої освіти: другий (магістерський)
Спеціальність: 125 – Кібербезпека та захист інформації
Науковий керівник: Муляр І.В. , канд.техн.наук, доц.
На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмним засобом(ами) комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту	відповідає
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	

Підтвердження:

Оригінальність тексту роботи за результатами перевірки системою StrikePlagiarism складає 99.0%, оригінальність тексту роботи за результатами перевірки системою Anti-Plagiarism складає 99.7%

Згідно з Положенням про систему забезпечення академічної доброчесності у ХНУ (https://khmnu.edu.ua/wp-content/uploads/normatyvni-dokumenty/polozhennya/pro-systemu-zabezpechennya-akademichnoyi-dobrochesnosti.pdf, Додаток В) кваліфікаційна робота, виконана за освітньо-професійною програмою, кількісні показники рівня унікальності тексту у відсотках до загального обсягу матеріалу в якій складає 75-100 %, визнається роботою з високим рівнем унікальності тексту: «Текст вважається унікальним і не потребує додаткових дій щодо запобігання неправомірним запозиченням».

Дата: 12.12.2025

Завідувач кафедри кібербезпеки _____ Юрій КЛЬОЦ

Гарант освітньої програми _____ Віра ТІТОВА

Керівник кваліфікаційної роботи _____ Ігор МУЛЯР

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

освітнього ступеня «магістр»

Студент Ратушняк Максим Віталійович

Тема Метод адаптивного управління ресурсами захисту інформаційної системи в умовах динамічних загроз

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека та захист інформайції

Обсяг кваліфікаційної роботи освітньо-кваліфікаційного рівня «магістр»:

кількість листів креслень - ; кількість сторінок записки 92

1. Короткий зміст кваліфікаційної роботи та прийнятих рішень В рамках роботи вирішено актуальну науково-прикладну задачу підвищення ефективності захисту інформаційних систем від динамічних загроз. Автором розроблено метод адаптивного управління ресурсами кіберзахисту, який, на відміну від традиційних статичних підходів, базується на математичному апараті динамічних баєсівських ігор та навчанні з підкріпленням. Розроблено програмний симулятор що дозволив експериментально підтвердити ефективність запропонованого методу: досягнуто зниження сумарних збитків та підвищення відсотка заблокованих атак завдяки динамічному перерозподілу ресурсів у реальному часі.

2. Висновок про відповідність кваліфікаційної роботи завданню Кваліфікаційна робота у повній мірі відповідає поставленому завданню. Автор продемонстрував глибоке розуміння предметної області, вдало поєднав теоретичні дослідження з практичною реалізацією та провів ґрунтовний аналіз отриманих результатів.

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У вступі чітко обґрунтовано актуальність переходу від статичних моделей захисту до адаптивних в умовах еволюції загроз та обмеженості ресурсів. У першому розділі виконано глибокий аналіз сучасного стану проблеми, зокрема недоліків сигнатурних методів та систем класу SOAR у протидії динамічним загрозам, що підтверджує необхідність використання методів штучного інтелекту. У другому розділі розроблено потужну математичну модель на основі теорії ігор, яка враховує невизначеність типу зловмисника. У третьому розділі описано архітектуру та реалізацію програмного комплексу, проведено серію експериментів за сценаріями MITRE ATT&CK, які довели економічну доцільність впровадження методу.

4. Позитивні сторони роботи Робота відзначається високим рівнем наукової новизни та практичної значущості. Її переваги полягають у створенні гібридної моделі, що геніально поєднує методи стратегічного планування з оперативною адаптацією. Розроблено симулятор, що дозволяє моделювати складні та реалістичні сценарії кібератак.

5. Негативні сторони роботи В якості незначного недоліку можна відзначити наявність ефекту «холодного старту» в алгоритмах навчання з підкріпленням, через що система потребує початкового періоду тренування для досягнення максимальної ефективності. Також доцільно було б розширити набір сценаріїв тестування для більш масштабних мережевих топологій. Зазначені зауваження не впливають на загальну високу оцінку роботи.

6. Оцінка графічного оформлення та пояснювальної записки роботи Пояснювальна записка відповідає нормам для її оформлення.

7. В загальному кваліфікаційна робота заслуговує позитивної оцінки. Матеріал роботи викладено логічно, послідовно та аргументовано. Структура роботи повністю розкриває тему дослідження. Автор продемонстрував здатність до самостійної наукової роботи та вирішення складних інженерних задач. Робота має завершений вигляд і може бути рекомендована до практичного впровадження.

8. Інші зауваження Відсутні.

9. Оцінка кваліфікаційної роботи Розглянувши позитивні сторони, актуальність теми та якість виконання представленої кваліфікаційної роботи, можна зробити висновок, що вона заслуговує на оцінку «відмінно» (94 бали).

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи)
Стецюк Віктор Іванович, доцент кафедри ТМІТ, кандидат технічних наук

« 16 » 12 2025. Віктор СТЕЦЮК (підпис)