Київський столичний університет імені Бориса Грінченка
Факультет інформаційних технологій та математики
Кафедра інформаційної та кібернетичної безпеки
імені професора Володимира Бурячка

«Допущено до захисту»
Завідувач кафедри інформаційної та
кібернетичної безпеки імені
професора Володимира Бурячка
кандидат технічних наук, доцент
Складанний П.М.

_____
(підпис)
« ___ » _____ 20__ р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття другого (магістерського)
рівня вищої освіти

Спеціальність 125 Кібербезпека та захист інформації

Тема роботи:

УДОСКОНАЛЕННЯ СИСТЕМИ ПРОТИДІЇ ВПЛИВУ ЗЛОЯКІСНОГО КОДУ, ШПИГУНСЬКОГО І ЗАВІДОМО ФАЛЬШИВОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Виконав

студент групи БІКСм-1-24-1.4д

Сундучков Андрій Михайлович

(прізвище, ім’я, по батькові)

_____

(підпис)

Науковий керівник

к.в.н., доцент

(науковий ступінь, наукове звання)

Аносов А.О.

(прізвище, ініціали)

_____

(підпис)

Київ – 2025

Київський столичний університет імені Бориса Грінченка
Факультет інформаційних технологій та математики
Кафедра інформаційної та кібернетичної безпеки
імені професора Володимира Бурячка

Освітньо-кваліфікаційних рівень – магістр
Спеціальність 125 Кібербезпека та захист інформації
Освітня програма 125.00.01 Безпека інформаційних і комунікаційних систем

«Затверджую»
Завідувач кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка
кандидат технічних наук, доцент
Складанний П.М.

_____
(підпис)
« ___ » _____ 20__ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
Сундучкову Андрію Михайловичу

1. Тема роботи: Удосконалення системи протидії впливу злоякісного коду, шпигунського і завідомо фальшивого програмного забезпечення; керівник Аносов А.О., к.в.н., доцент, затверджені наказом ректора від «___»____ 20___ року №__.
2. Термін подання студентом роботи «___»_____ 20___ р.
3. Вихідні дані до роботи:
   3.1 науково-технічна та нормативна література з теми дослідження;
   3.2 методи: системний аналіз, порівняльний аналіз;
4. Зміст текстової частини роботи (перелік питань, які потрібно розробити):
   4.1 Проаналізувати сучасні типи злоякісного програмного забезпечення та методи їх функціонування.
   4.2 Дослідити існуючі методи та технології виявлення зловмисного ПЗ, визначити їх переваги та недоліки.
   4.3 Провести порівняльний аналіз ефективності комерційних антивірусних рішень та спеціалізованих засобів захисту.
   4.4 Розробити архітектуру удосконаленої системи протидії злоякісному ПЗ на основі поведінкового аналізу.
   4.5 Реалізувати програмний застосунок для моніторингу системної активності та виявлення підозрілої поведінки процесів.
   4.6 Провести тестування розробленої системи та порівняльний аналіз її ефективності з

існуючими рішеннями.

5. Перелік графічного матеріалу:

5.1 Презентація доповіді, виконана в Microsoft PowerPoint.

6. Дата видачі завдання «___»_____ 20___ р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів підготовки роботи	Термін виконання	Примітка
1.	Уточнення постановки завдання		
2.	Аналіз літератури		
3.	Обґрунтування вибору рішення		
4.	Збір даних		
5.	Виконання та оформлення розділу 1.		
6.	Виконання та оформлення розділу 2.		
7.	Виконання та оформлення розділу 3.		
8.	Вступ, висновки, реферат		
9.	Апробація роботи на науково-методичному семінарі та/або науково-технічній конференції		
10.	Оформлення та друк текстової частини роботи		
11.	Оформлення презентацій		
12.	Отримання рецензій		
13.	Попередній захист роботи		
14.	Захист в ЕК		

Студент _____ Сундучков Андрій Михайлович
(підпис)

Науковий керівник _____ Аносов Андрій Олександрович
(підпис)

РЕФЕРАТ

Кваліфікаційна робота присвячена технологіям забезпечення безпеки систем, зокрема розробці удосконаленої системи моніторингу та виявлення загроз від злоякісного програмного забезпечення для операційних систем Windows.

Робота складається з вступу, трьох розділів, що містять 3 рисунки та 3 таблиці, а також розділу з охорони праці. Загальний обсяг роботи становить 121 сторінок. Перелік використаних джерел налічує 65 найменування.

Об'єктом дослідження є процес виявлення та протидії злоякісному програмному забезпеченню в операційних системах Windows.

Предметом дослідження є методи та технології забезпечення безпеки систем від злоякісного програмного забезпечення на основі поведінкового аналізу та моніторингу системної активності.

Метою роботи є підвищення ефективності виявлення сучасних загроз інформаційної безпеки шляхом розробки системи моніторингу, яка використовує комбінований підхід на основі сигнатурного, поведінкового та евристичного аналізу.

Для досягнення поставленої мети в роботі:

- проведено аналіз наукових підходів до виявлення різних типів злоякісного програмного забезпечення та проаналізовано методи запропоновані іншими дослідниками

- досліджено особливості існуючих антивірусних рішень та виявлено їх недоліки

- розроблено концепцію удосконаленої системи захисту на основі багатомодульної архітектури

- здійснено програмну реалізацію системи моніторингу з використанням Windows API

- обґрунтовано використання комбінованого підходу для виявлення загроз різних типів.

Наукова новизна одержаних результатів полягає в тому, що в роботі запропоновано удосконалений математичний модель виявлення аномальної поведінки в системі на основі багатокритеріального аналізу системних подій з використанням інтегральної оцінки ризику та адаптивних порогів детекції.

Відповідно до отриманих результатів проведеного дослідження підтверджено, що запропонована система моніторингу успішно виявляє програми-шифрувальники з точністю дев'яносто сім відсотків, техніки впровадження коду з точністю вісімдесят чотири – дев'яносто два відсотки та викрадення облікових даних з точністю дев'яносто шість відсотків. Система демонструє середній час виявлення загрози чотири секунди при споживанні ресурсів три-чотири відсотки процесора та двісті-триста мегабайт оперативної пам'яті. При комбінованому використанні з традиційним антивірусом досягається синергетичний ефект з частотою виявлення дев'яносто вісім відсотків.

Ключові слова: БЕЗПЕКА, ЗАГРОЗА, ІНФОРМАЦІЯ, ІНФОРМАЦІЙНО-АНАЛІТИЧНА СИСТЕМА, ОБ'ЄКТ БЕЗПЕКИ, ПОРУШНИК, СИСТЕМА ЗАХИСТУ.

ЗМІСТ

ВСТУП 9
РОЗДІЛ 1 АНАЛІЗ СУЧАСНИХ ЗАГРОЗ ТА МЕТОДІВ ЗАХИСТУ ВІД ЗЛОЯКІСНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 12
1.1 Класифікація та характеристика злоякісного програмного забезпечення 12
1.1.1 Віруси та черв'яки 12
1.1.2 Троянські програми та шпигунське ПЗ 14
1.1.3 Програми-вимагачі (ransomware) 15
1.1.4 Руткіти та буткіти 17
1.2 Сучасні методи виявлення та протидії зловмисному ПЗ 19
1.2.1 Сигнатурний аналіз 19
1.2.2 Евристичний аналіз 21
1.2.3 Поведінковий аналіз 22
1.2.4 Технології «пісочниці» (sandboxing) 24
1.3 Огляд існуючих антивірусних рішень та їх можливостей 26
1.3.1 Комерційні антивірусні продукти 26
1.3.2 Спеціалізовані сканери та утиліти 28
1.3.3 Порівняльний аналіз ефективності 30
1.4 Аналіз недоліків існуючих систем захисту 32
Висновки до розділу 1 34
РОЗДІЛ 2 РОЗРОБКА КОНЦЕПЦІЇ УДОСКОНАЛЕНОЇ СИСТЕМИ ПРОТИДІЇ ЗЛОЯКІСНОМУ ПЗ 37
2.1 Вимоги до системи захисту від сучасних загроз 37
2.2 Архітектура удосконаленої системи захисту 39
2.2.1 Модуль моніторингу файлової системи 40
2.2.2 Модуль контролю цілісності процесів 41
2.2.3 Модуль перевірки цифрових підписів 43
2.2.4 Модуль виявлення аномальної поведінки 44
2.3 Методи виявлення підозрілої активності 46

2.3.1 Моніторинг змін у файловій системі 46
2.3.2 Виявлення міжпроцесної модифікації даних 47
2.3.3 Контроль доступу до пам'яті інших процесів 48
2.3.4 Аналіз частоти створення процесів 50
2.4 Інтеграція з існуючими засобами захисту 51
Висновки до розділу 2 54
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ МОНІТОРИНГУ ТА ВИЯВЛЕННЯ ЗАГРОЗ 56
3.1 Вибір технологій та інструментів розробки 56
3.1.1 Мова програмування та фреймворки 56
3.1.2 Системні API для моніторингу Windows 59
3.2 Реалізація основних модулів системи 61
3.2.1 Модуль моніторингу файлової системи 61
3.2.2 Модуль контролю процесів 63
3.2.3 Модуль перевірки цифрових підписів 66
3.2.4 Модуль виявлення аномалій 68
3.3 Розробка користувацького інтерфейсу 71
3.4 Тестування розробленої системи 73
3.4.1 Методика тестування 73
3.4.2 Тестові сценарії та результати 74
3.5 Порівняльний аналіз ефективності 79
3.5.1 Порівняння з комерційними антивірусами 79
3.5.2 Оцінка швидкодії та ресурсоємності 81
3.5.3 Аналіз виявлення різних типів загроз 83
Висновки до розділу 3 85
РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ 89
4.1 Аналіз умов праці при роботі з комп'ютерною технікою 89
4.2 Вимоги до організації робочого місця 92
4.3 Електробезпека та пожежна безпека 96
4.4 Безпека в надзвичайних ситуаціях 101

Висновки до розділу 4 106
ЗАГАЛЬНІ ВИСНОВКИ 109
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 115

ВСТУП

Актуальність теми дослідження. У сучасному цифровому суспільстві інформаційна безпека стала одним із найважливіших аспектів функціонування як корпоративних структур, так і приватних користувачів. Стрімкий розвиток інформаційних технологій супроводжується паралельним зростанням кількості та складності кіберзагроз. За даними аналітичних компаній, щороку фіксується зростання кількості атак злоякісного програмного забезпечення на 25-30%, а збитки від кіберзлочинності у світі вже перевищують 6 трильйонів доларів США щорічно.

Злоякісне програмне забезпечення (malware) еволюціонує швидкими темпами, набуваючи все більш витончених форм та механізмів уникнення виявлення. Сучасні зловмисники використовують передові технології, включаючи штучний інтелект, поліморфні алгоритми та технології стеганографії для створення складних загроз, які здатні обходити традиційні засоби захисту. Особливу небезпеку становлять програми-вимагачі, шпигунське ПЗ та руткіти, які можуть призвести до втрати критично важливої інформації, фінансових збитків та порушення конфіденційності даних.

Традиційні антивірусні рішення, які базуються переважно на сигнатурному аналізі, часто виявляються неефективними проти нових, раніше невідомих загроз (zero-day attacks). Це створює необхідність розробки удосконалених систем захисту, які поєднують різні методи виявлення загроз та здатні адаптуватися до нових видів атак.

В Україні проблема кібербезпеки набула особливої гостроти у зв'язку з активізацією кібератак на критичну інфраструктуру, державні установи та приватний сектор. Це підкреслює важливість розробки ефективних національних рішень у сфері інформаційної безпеки, які враховують специфіку сучасних загроз та можуть забезпечити надійний захист інформаційних систем.

Зв'язок роботи з науковими програмами, планами, темами. Дослідження виконано в рамках наукових напрямів кафедри, пов'язаних із забезпеченням інформаційної безпеки та захисту інформації в автоматизованих системах. Тема роботи узгоджується із Стратегією кібербезпеки України та відповідає актуальним напрямам розвитку систем захисту інформації.

Мета і задачі дослідження. Метою роботи є удосконалення системи протидії впливу злоякісного коду, шпигунського і завідомо фальшивого програмного забезпечення шляхом розробки комплексного підходу до виявлення та нейтралізації загроз на основі поведінкового аналізу та моніторингу системних ресурсів.

Для досягнення поставленої мети необхідно вирішити наступні задачі:

1. Проаналізувати сучасні типи злоякісного програмного забезпечення та методи їх функціонування.
2. Дослідити існуючі методи та технології виявлення зловмисного ПЗ, визначити їх переваги та недоліки.
3. Провести порівняльний аналіз ефективності комерційних антивірусних рішень та спеціалізованих засобів захисту.
4. Розробити архітектуру удосконаленої системи протидії злоякісному ПЗ на основі поведінкового аналізу.
5. Реалізувати програмний застосунок для моніторингу системної активності та виявлення підозрілої поведінки процесів.
6. Провести тестування розробленої системи та порівняльний аналіз її ефективності з існуючими рішеннями.

Об'єкт дослідження – процеси виявлення та протидії злоякісному програмному забезпеченню в операційних системах сімейства Windows.

Предмет дослідження – методи та засоби моніторингу системної активності для виявлення аномальної поведінки, характерної для злоякісного програмного забезпечення.

Методи дослідження. У роботі використано комплекс методів дослідження: аналіз і синтез – для визначення особливостей функціонування злоякісного ПЗ; порівняльний аналіз – для оцінки ефективності існуючих антивірусних рішень; системний підхід – для проектування архітектури системи захисту; методи об'єктно-орієнтованого програмування – для реалізації програмного забезпечення; експериментальні методи – для тестування та перевірки ефективності розробленої системи.

Наукова новизна отриманих результатів полягає у розробці комплексного підходу до виявлення злоякісного ПЗ, який поєднує моніторинг файлової системи, контроль міжпроцесної взаємодії, перевірку цифрових підписів та аналіз аномальної поведінки процесів у реальному часі, що дозволяє виявляти раніше невідомі загрози.

Практичне значення отриманих результатів. Розроблена система може бути використана як додатковий рівень захисту інформаційних систем у поєднанні з традиційними антивірусними рішеннями. Програмний застосунок може застосовуватися для моніторингу критично важливих систем, серверів та робочих станцій, де необхідний підвищений рівень контролю за системною активністю.

Апробація результатів роботи. Основні положення та результати роботи доповідалися на науково-практичних конференціях та семінарах кафедри.

Структура та обсяг роботи. Магістерська робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел та додатків.

РОЗДІЛ 1 АНАЛІЗ СУЧАСНИХ ЗАГРОЗ ТА МЕТОДІВ ЗАХИСТУ ВІД ЗЛОЯКІСНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1.1 Класифікація та характеристика злоякісного програмного забезпечення

1.1.1 Віруси та черв'яки

Комп'ютерні віруси є одним з найстаріших типів злоякісного програмного забезпечення, що з'явилися ще в 1970-х роках як експериментальні програми, але згодом стали серйозною загрозою для інформаційних систем. Характерною особливістю вірусів є їх здатність до самовідтворення шляхом впровадження власного коду в інші програми або файли. Комп'ютерний вірус можна визначити як програму, яка може інфікувати інші програми шляхом включення в них своєї, можливо модифікованої, копії, при цьому зараження відбувається таким чином, що інфіковані програми також набувають здатності до самовідтворення. Віруси поширюються через виконання інфікованих файлів, що відрізняє їх від черв'яків, які можуть поширюватися автономно [1, с. 59-69].

Основні типи комп'ютерних вірусів включають файлові віруси, які інфікують виконувані файли додаючи свій код на початок, кінець або середину легітимної програми. При запуску інфікованого файлу вірус отримує керування, виконує власний код, що може включати пошук нових цілей для зараження, та передає керування основній програмі. Завантажувальні віруси інфікують завантажувальний сектор дисків або головну завантажувальну запис, активуючись на ранніх стадіях завантаження операційної системи, що дозволяє їм отримати контроль над системою ще до запуску антивірусного ПЗ. Макровіруси використовують макромови, вбудовані в документи офісних програм, та поширюються через заражені документи, автоматично виконуючись при відкритті файлу.

Поліморфні віруси здатні змінювати власний код при кожному акті зараження, що ускладнює їх виявлення за допомогою сигнатурного аналізу, досягаючи це шляхом шифрування тіла вірусу та використання змінного дешифратора. Метаморфні віруси являють собою більш складний варіант, оскільки вони не лише шифрують свій код, але й фізично змінюють його структуру, використовуючи техніки переписування коду, вставляючи непотрібні інструкції та змінюючи порядок виконання операцій. Ці типи вірусів демонструють високий рівень технічної складності та створюють значні труднощі для традиційних методів виявлення.

Комп'ютерні черв'яки відрізняються від вірусів своєю здатністю до автономного поширення без необхідності інфікування інших програм. Черв'як є самостійною програмою, яка може копіювати себе через мережу або інші канали передачі даних без участі користувача або прив'язки до файлів-носіїв. Мережеві черв'яки використовують вразливості в мережевих протоколах та сервісах для поширення, скануючи мережу в пошуках уразливих систем, експлуатуючи знайдені вразливості для проникнення та копіюючи себе на нові машини. Класичними прикладами є черв'яки Blaster, Sasser та Conficker, які спричинили масштабні епідемії, інфікуючи мільйони комп'ютерів по всьому світу [2, с. 45].

Сучасні черв'яки часто є багатовекторними, використовуючи кілька методів поширення одночасно через мережеві вразливості, USB-носії та електронну пошту, що значно підвищує їх шанси на успішне інфікування нових систем. Особливу небезпеку становлять черв'яки, які експлуатують вразливості нульового дня, оскільки для таких вразливостей ще не існує патчів безпеки, що дозволяє черв'якам швидко поширюватися та інфікувати велику кількість систем до моменту випуску виправлення. Наслідки інфікування вірусами та черв'яками можуть варіюватися від незначних порушень роботи системи до повної втрати даних та виходу з ладу критичної інфраструктури, при цьому вони також можуть використовуватися як інструмент доставки іншого злоякісного ПЗ [3, с. 117].

1.1.2. Троянські програми та шпигунське ПЗ

Троянські програми отримали свою назву за аналогією з легендою про троянського коня та відрізняються від вірусів і черв'яків відсутністю механізмів самовідтворення та самостійного поширення. Натомість вони маскуються під легітимне програмне забезпечення або проникають у систему у складі інших програм, приховуючи свою справжню функціональність від користувача. Троянські програми можуть виконувати широкий спектр шкідливих дій, створюючи потаємні канали доступу до інфікованої системи, викрадаючи фінансову та особисту інформацію, завантажуючи додаткове злоякісне ПЗ або використовуючи ресурси системи для власних цілей зловмисників.

Backdoor-трояни створюють потаємний канал доступу до інфікованої системи, відкриваючи певні порти, встановлюючи з'єднання з командними серверами зловмисників та виконуючи отримані команди, що може використовуватися для викрадення даних, встановлення додаткового злоякісного ПЗ або проведення атак на інші системи. Банківські трояни спеціалізуються на викраденні фінансової інформації та облікових даних для онлайн-банкінгу, перехоплюючи дані введені користувачем на банківських сайтах, роблячи скріншоти екрану та модифікуючи веб-сторінки для відображення фальшивих форм введення даних. Відомі приклади включають Zeus, SpyEye та Dridex, деякі з яких використовують складні техніки типу man-in-the-browser attacks для модифікації трафіку у реальному часі [4, с. 38].

Downloader-трояни та dropper-трояни призначені для завантаження та встановлення додаткового злоякісного ПЗ на інфіковану систему, часто виступаючи першим етапом багатоступеневої атаки. Це дозволяє зловмисникам обходити системи захисту, які можуть блокувати великі або підозрілі файли, шляхом початкового впровадження невеликого трояна, який потім завантажує більш складні та функціональні зразки malware. Сучасні троянські програми використовують різноманітні техніки приховування, включаючи впровадження в легітимні процеси,

використання руткіт-технологій, обфускацію коду та техніку "living off the land", коли замість власних інструментів використовуються легітимні системні утиліти.

Шпигунське програмне забезпечення є особливою категорією злоякісного ПЗ, основною метою якого є таємний збір інформації про користувача та його діяльність. Кейлогери реєструють натискання клавіш користувачем, дозволяючи зловмисникам перехоплювати паролі, номери кредитних карт та іншу конфіденційну інформацію, при цьому сучасні кейлогери також можуть робити скріншоти екрану, записувати буфер обміну та фіксувати активність миші. Інформаційні крадії призначені для викрадення різноманітної інформації зі системи, включаючи збережені паролі з браузерів, cookies, історію відвідувань, облікові дані для різних сервісів та криптовалютні гаманці, часто маючи модульну архітектуру для гнучкого налаштування збору даних.

Особливістю троянських програм та шпигунського ПЗ є їх прагнення залишатися непоміченими якомога довше, використовуючи для цього різноманітні техніки приховування та обфускації. Вони часто поширюються через фішингові електронні листи з шкідливими вкладеннями, заражені веб-сайти з використанням вразливостей браузерів, піратське програмне забезпечення, соціальну інженерію та експлуатацію вразливостей у легітимному ПЗ [5, с. 142-153]. Виявлення троянських програм та шпигунського ПЗ є складним завданням через їх здатність маскуватися та використовувати легітимні системні функції, що вимагає використання багатошарового підходу до захисту з моніторингом поведінки системи та регулярним сканування спеціалізованими інструментами.

1.1.3. Програми-вимагачі (ransomware)

Програми-вимагачі або ransomware є одним з найнебезпечніших та найруйнівніших типів злоякісного програмного забезпечення сучасності, основною метою якого є отримання фінансової вигоди шляхом шифрування або блокування

доступу до даних користувача з подальшою вимогою викупу за їх розблокування. За даними аналітичних компаній, збитки від атак ransomware у світі досягають десятків мільярдів доларів щорічно, і ця цифра продовжує зростати, оскільки програми-вимагачі атакують як приватних користувачів, так і корпоративні структури, державні установи, медичні заклади та критичну інфраструктуру.

Шифрувальні програми-вимагачі є найпоширенішим та найнебезпечнішим типом, які після проникнення в систему шифрують файли користувача, використовуючи сильні криптографічні алгоритми, зазвичай комбінацію симетричного та асиметричного шифрування. Процес роботи типової шифрувальної програми-вимагача включає проникнення в систему через вразливості або фішинг, закріплення та отримання необхідних привілеїв, генерацію пари криптографічних ключів, сканування файлової системи, шифрування файлів, видалення резервних копій та відображення повідомлення з вимогою викупу. Відомі приклади включають WannaCry, Petya/NotPetya, Locky, Cerber та Ryuk, деякі з яких спричинили масштабні епідемії з мільярдними збитками [6].

Блокувальні програми-вимагачі не шифрують файли, а блокують доступ до системи, відображаючи екран з вимогою викупу та блокуючи можливість взаємодії з операційною системою, часто маскуючись під повідомлення від правоохоронних органів про виявлення нелегального контенту. Doxware або leakware представляють новий тип ransomware, який не лише шифрує дані, але й загрожує їх оприлюдненням у випадку несплати викупу, створюючи подвійний тиск на жертву, особливо якщо викрадена інформація є критичною для бізнесу або містить персональні дані клієнтів. Бізнес-модель Ransomware-as-a-Service дозволяє розробникам надавати своє злоякісне ПЗ іншим кіберзлочинцям на комерційній основі, значно знизивши поріг входження в кіберзлочинність.

Сучасні тенденції розвитку ransomware включають цільові атаки на великі організації замість масових атак на приватних користувачів, де зловмисники проводять детальну розвідку перед атакою, вивчаючи інфраструктуру жертви та

визначаючи критичні системи. Практика подвійного та потрійного вимагання передбачає не лише шифрування даних, але й загрозу їх оприлюднення, DDoS-атак на веб-сайт організації або повідомлення клієнтів про витік персональних даних. Атаки на ланцюги постачання дозволяють зловмисникам через компрометацію одного постачальника отримати доступ до сотень або тисяч його клієнтів.

Захист від ransomware вимагає комплексного підходу, що включає регулярне резервне копіювання даних з відокремленням резервних копій від основної мережі, своєчасне оновлення програмного забезпечення та операційних систем, обмеження прав користувачів за принципом найменших привілеїв та сегментацію мережі для обмеження поширення malware. Навчання користувачів розпізнавати фішингові атаки, використання багатошарового захисту включаючи EDR-рішення та моніторинг аномальної активності в файловій системі є критично важливими компонентами ефективної стратегії захисту [7, с. 150-164]. Сучасні атаки ransomware часто використовують легітимні адміністративні інструменти для поширення в мережі та шифрування файлів, що значно ускладнює виявлення атаки традиційними методами.

1.1.4. Руткіти та буткіти

Руткіти представляють собою один з найскладніших та найнебезпечніших типів злоякісного програмного забезпечення через їх здатність приховувати свою присутність у системі та надавати зловмисникам глибокий рівень контролю над інфікованим комп'ютером. Термін "rootkit" походить з світу Unix/Linux, де "root" означає привілейований обліковий запис з повним доступом до системи, а "kit" – набір інструментів. Основна мета руткіта полягає в прихованні присутності іншого злоякісного ПЗ або зловмисної діяльності в системі шляхом приховування файлів, директорій, процесів, ключів реєстру, мережевих з'єднань, драйверів та облікових записів користувачів.

Руткіти користувацького режиму працюють на рівні прикладних програм та не обов'язково потребують прав адміністратора для функціонування, хоча такі права значно розширюють їх можливості. Вони модифікують виконання системних бібліотек та API-функцій, перехоплюючи виклики до операційної системи та фільтруючи результати, наприклад, видаляючи зі списку файлів ті, що належать злоякісному ПЗ. Техніки, що використовуються user-mode руткітами, включають IAT/EAT hooking для модифікації таблиць імпорту та експорту функцій, inline hooking для безпосередньої модифікації коду функцій у пам'яті, DLL injection для впровадження злоякісних бібліотек у процеси та API hooking для перехоплення викликів Windows API.

Руткіти режиму ядра функціонують на найвищому рівні привілеїв операційної системи, маючи повний контроль над системою та можливість модифікувати ядро, перехоплювати системні виклики та приховувати свою присутність від антивірусного ПЗ. Kernel-mode руткіти використовують техніки SSDT hooking для модифікації таблиці системних сервісів, IDT hooking для перехоплення переривань, IRP hooking для перехоплення запитів введення-виведення, DKOM для прямої маніпуляції об'єктами ядра та фільтри файлової системи для перехоплення операцій з файлами. Виявлення kernel-mode руткітів є надзвичайно складним завданням, оскільки вони мають той самий рівень привілеїв, що й антивірусне ПЗ, і можуть ефективно приховувати свою діяльність від засобів моніторингу.

Буткіти являють собою найскладніший та найнебезпечніший тип руткітів, оскільки вони інфікують завантажувальні компоненти системи, включаючи головний завантажувальний запис, завантажувальний сектор, завантажувач операційної системи або компоненти UEFI firmware. Буткіти активуються на найранніших стадіях завантаження системи, ще до запуску операційної системи, що дає їм максимальний контроль та робить їх виявлення надзвичайно складним [8, с. 139]. Переваги буткітів для зловмисників включають активацію до запуску засобів захисту,

здатність модифікувати процес завантаження ОС, стійкість до переінсталяції операційної системи та можливість обходу деяких механізмів захисту на рівні ОС.

Методи виявлення руткітів включають сигнатурне сканування проти відомих зразків, аналіз поведінки системи для виявлення аномалій, порівняння результатів різних методів сканування для виявлення прихованих об'єктів, аналіз пам'яті для пошуку модифікованих системних структур та offline-сканування із завантаження з чистого носія. Захист від руткітів та буткітів забезпечується технологіями Secure Boot для перевірки цифрових підписів завантажувальних компонентів, Measured Boot для створення криптографічного ланцюжка довіри, Kernel Patch Protection у 64-бітних версіях Windows, Driver Signature Enforcement та технологіями віртуалізації безпеки. Незважаючи на ці захисні механізми, руткіти залишаються серйозною загрозою, особливо для застарілих систем, а їх виявлення та видалення вимагає спеціалізованих інструментів та глибоких знань архітектури операційної системи.

1.2. Сучасні методи виявлення та протидії зловмисному ПЗ

1.2.1. Сигнатурний аналіз

Сигнатурний аналіз є найстарішим та найпоширенішим методом виявлення злоякісного програмного забезпечення, який базується на ідентифікації відомих зразків malware шляхом порівняння характеристик файлів з базою даних сигнатур відомих загроз. Сигнатура являє собою унікальний шаблон або послідовність байтів, що характеризує конкретний зразок злоякісного ПЗ, та може включати криптографічні геш-суми файлів, специфічні послідовності байтів, текстові рядки або структурні особливості виконуваних файлів. Принцип роботи достатньо простий: антивірусний сканер обчислює контрольні суми файлів, аналізує їх вміст та порівнює отримані дані з базою відомих сигнатур, класифікуючи файл як злоякісний при виявленні збігу.

Типи сигнатур варіюються за складністю та специфічністю виявлення. Прості геш-основані сигнатури використовують криптографічні функції для обчислення контрольної суми файлу та порівняння її з базою відомих геш-сум malware, що є надзвичайно швидким та точним методом для виявлення точних копій, але абсолютно неефективним проти модифікованих версій. Строкові сигнатури шукають специфічні послідовності байтів або текстових рядків у файлі, такі як характерні URL командних серверів або унікальні послідовності інструкцій, будучи більш гнучкими та здатними виявити злоякісний код навіть після деяких модифікацій. Структурні сигнатури аналізують особливості структури виконуваних файлів, включаючи заголовки, секції, точки входу та імпортовані функції, які часто мають характерні аномалії у злоякісному ПЗ [9, с. 393-398].

Переваги сигнатурного аналізу включають високу швидкість роботи, що дозволяє сканувати великі обсяги даних за короткий час, низьку кількість хибних спрацьовувань при правильно розроблених сигнатурах, точність ідентифікації конкретних зразків malware та простоту реалізації алгоритмів. Проте метод має значні недоліки, найголовнішим з яких є неефективність проти нових загроз zero-day, оскільки можуть бути виявлені лише відомі загрози з існуючими сигнатурами. Проблема часової затримки між виявленням нового malware та створенням і розповсюдженням сигнатури залишає системи вразливими протягом певного періоду, а вразливість до модифікацій дозволяє зловмисникам легко змінити код для обходу існуючих сигнатур.

Зловмисники активно розробляють методи обходу сигнатурного аналізу, використовуючи обфускацію коду для зміни структури програми без зміни функціональності, шифрування та пакування для зміни зовнішнього вигляду malware, поліморфізм для автоматичної генерації різних версій з різними сигнатурами та метаморфізм для складнішого переписування машинного коду. Для підвищення ефективності сигнатурного методу використовуються вдосконалення, такі як емуляція та розпакування перед сканування, нечіткі геші для виявлення

подібності між файлами, машинне навчання для автоматичної генерації сигнатур та кластеризація для створення узагальнених сигнатур, що покривають цілі сімейства загроз [10, с. 56]. Незважаючи на обмеження, сигнатурний аналіз залишається важливою складовою сучасних систем захисту, забезпечуючи швидке та точне виявлення відомих загроз, що становлять переважну більшість реальних атак.

1.2.2. Евристичний аналіз

Евристичний аналіз є наступним кроком в еволюції технологій виявлення злоякісного ПЗ, спрямованим на подолання основного обмеження сигнатурного методу – нездатності виявляти нові, раніше невідомі загрози. Термін "евристика" походить з грецької мови та означає "знаходжу, відкриваю", а в контексті антивірусних технологій базується на виявленні підозрілих характеристик та поведінкових патернів, характерних для злоякісного ПЗ. Замість пошуку точних збігів з відомими сигнатурами, евристичний аналіз оцінює файл на основі набору правил та критеріїв, розроблених експертами з безпеки, які описують характеристики, що часто зустрічаються у злоякісному ПЗ, але рідко у легітимних програмах.

Статичний евристичний аналіз досліджує файл без його виконання, аналізуючи структурні аномалії, такі як незвичне розташування точки входу або підозрілі секції з одночасними правами на запис та виконання. Аналіз використовуваних API-функцій дозволяє виявити підозрілі комбінації функцій, наприклад, одночасне звернення до мережевих функцій, реєстру та файлової системи може вказувати на троянську програму. Пошук підозрілих рядків та метаданих включає виявлення URL відомих злоякісних доменів, команд для шифрування або типових повідомлень програм-вимагачів. Ентропійний аналіз вимірює ступінь випадковості різних секцій файлу, де висока ентропія може вказувати на шифрування або стиснення, що часто використовується malware для приховування [11-12].

Динамічний евристичний аналіз виконує файл у контрольованому віртуальному середовищі та спостерігає за його поведінкою, відстежуючи файлові операції, включаючи створення, модифікацію або видалення файлів, особливо системних. Моніторинг операцій з реєстром виявляє створення ключів автозапуску, модифікацію параметрів безпеки або зміну конфігурації системи. Процесна активність включає створення нових процесів з підозрілими параметрами, впровадження коду в інші процеси або приховування власних процесів. Мережева активність контролюється для виявлення з'єднань з невідомими або підозрілими серверами, відкриття портів для прослуховування або передачі даних на зовнішні адреси.

Правила та оцінювальні системи присвоюють кожній підозрілій характеристиці певну вагу або оцінку, де сумарна оцінка визначає ймовірність того, що файл є злоякісним. Наприклад, звернення до функцій роботи з реєстром може додавати п'ять балів, висока ентропія – десять балів, спроба модифікувати системні файли – п'ятнадцять балів, а встановлення мережевих з'єднань – ще десять балів, і при досягненні певного порогу файл класифікується як злоякісний. Переваги евристичного аналізу включають здатність виявляти нові загрози та варіанти відомого malware без конкретних сигнатур, забезпечуючи проактивний захист, але недоліки включають можливість хибних спрацьовувань, складність налаштування правил, ресурсоємність особливо динамічного аналізу та можливість обходу зловмисниками, які розробляють malware з урахуванням відомих евристичних правил [13]. Сучасні підходи використовують машинне навчання та нейронні мережі для автоматичного виявлення патернів злоякісної поведінки та адаптації евристичних правил на основі аналізу нових загроз.

1.2.3. Поведінковий аналіз

Поведінковий аналіз представляє наступний рівень еволюції технологій виявлення злоякісного ПЗ, відрізняючись від сигнатурного та статичного

евристичного аналізу тим, що моніторить реальну діяльність програм у системі в режимі реального часу або в ізольованому середовищі. Основна ідея полягає в тому, що незалежно від обфускації або модифікації коду, злоякісне ПЗ повинно виконувати певні дії для досягнення своїх цілей, такі як шифрування файлів, викрадення даних, поширення в мережі або встановлення backdoor, і ці дії можна виявити та ідентифікувати як злоякісні. Методи поведінкового аналізу включають моніторинг API-викликів, файлової системи, реєстру Windows, мережевої активності, процесів та доступу до пам'яті [14].

Моніторинг API-викликів відстежує звернення програм до системних функцій операційної системи, де підозріла послідовність викликів може вказувати на злоякісну діяльність. Наприклад, програма, яка відкриває багато файлів, читає їх вміст, шифрує і записує назад, демонструє ознаки ransomware, а впровадження в інший процес через CreateRemoteThread та WriteProcessMemory вказує на техніку process injection. Моніторинг файлової системи відстежує операції створення, читання, запису та видалення файлів, виявляючи аномальну активність, таку як масові модифікації файлів за короткий період, зміна розширень на незвичні, створення файлів у системних директоріях або видалення тіньових копій файлів.

Моніторинг реєстру Windows аналізує зміни в системному реєстрі, включаючи створення ключів автозапуску, модифікацію параметрів безпеки, реєстрацію нових сервісів або зміну конфігурації фаєрвола. Мережевий моніторинг відстежує спроби з'єднання з відомими злоякісними доменами, використання нетипових протоколів або портів, великі обсяги вихідного трафіку для можливої ексфільтрації даних, періодичні з'єднання з командними серверами або сканування мережі в пошуках уразливих систем. Моніторинг процесів аналізує створення дочірніх процесів з незвичними параметрами, використання легітимних системних утиліт для злоякісних цілей, приховування процесів, підвищення привілеїв або впровадження коду в інші процеси [15].

Технології поведінкового аналізу включають host-based аналіз, який працює безпосередньо на захищуваній системі з перевагами детального моніторингу та швидкої реакції, але недоліками споживання ресурсів, та sandbox-based аналіз, який виконує підозрілі файли в ізольованому віртуальному середовищі з перевагами безпеки та детального аналізу, але витратами часу. Виявлення аномалій використовує статистичні моделі та машинне навчання для побудови моделей нормальної поведінки та виявлення відхилень, що можуть вказувати на присутність malware. Переваги поведінкового аналізу включають виявлення zero-day загроз, поліморфного та метаморфного malware, fileless malware та надання контекстної інформації для розслідування інцидентів, але недоліки включають проблему хибних спрацьовувань, реактивний характер, ресурсоємність та можливість обходу складним malware [16, с. 47-52]. Сучасні EDR рішення інтегрують поведінковий аналіз з можливостями розслідування та реагування на інциденти, використовуючи AI та машинне навчання для побудови складних моделей та інтеграцію з threat intelligence для збагачення аналізу.

1.2.4. Технології «пісочниці» (sandboxing)

Технологія "пісочниці" є спеціалізованим різновидом поведінкового аналізу, що передбачає виконання підозрілого програмного забезпечення в ізольованому контрольованому середовищі для детального спостереження за його поведінкою без ризику для основної системи. Термін "sandbox" походить від дитячого ігрового майданчика, символізуючи обмежений простір, де можна безпечно експериментувати без ризику для навколишнього середовища. Пісочниця створює віртуальне середовище, яке максимально імітує реальну операційну систему, але ізольоване від неї, де підозрілий файл запускається з моніторингом всіх його дій, включаючи файлові операції, мережеву активність, зміни реєстру та створення процесів, після

чого sandbox повертається до початкового стану з передачею результатів аналізу для прийняття рішення.

Типи пісочниць включають повну віртуалізацію, яка використовує гіпервізори для створення повноцінних віртуальних машин з максимальною ізоляцією та безпекою, але високими вимогами до ресурсів та можливістю виявлення malware. Контейнерна віртуалізація створює ізольовані середовища на рівні операційної системи, будучи легшою та швидшою за повну віртуалізацію, але надаючи меншу ізоляцію. Емуляція відтворює поведінку процесора та операційної системи програмно, дозволяючи аналізувати malware для різних архітектур, але будучи повільнішою за віртуалізацію. Sandbox на основі API-перехоплення не створює повноцінну віртуальну машину, а перехоплює та контролює системні виклики програми, будучи легким та швидким підходом, але надаючи менше інформації про поведінку malware.

Аналіз в пісочниці включає статичну підготовку для збору базової інформації перед виконанням та динамічне виконання з моніторингом файлової активності, реєстру, процесів, мережевої активності та системних ресурсів. Системи sandbox створюють "поведінкові сигнатури" на основі спостережуваних дій, які можуть бути використані для класифікації malware. Переваги sandboxing включають безпеку аналізу, детальну інформацію про дії malware, виявлення zero-day загроз та масштабованість для автоматизованого аналізу великої кількості зразків, але недоліки включають можливість sandbox evasion через виявлення віртуального середовища, ресурсоємність аналізу, обмежений час виконання та неповну емуляцію реального середовища [17, с. 357].

Зловмисники активно розробляють техніки детектування sandbox, включаючи перевірку віртуального середовища через пошук артефактів віртуалізації, виявлення специфічних драйверів VM, перевірку MAC-адрес та аналіз характеристик процесора. Виявлення обмеженого середовища базується на недостатній кількості файлів у системі, відсутності історії використання, малій кількості запущених

процесів або невідповідності системного часу. Відкладене виконання дозволяє malware не виконувати злоякісні дії одразу, очікуючи на певні умови або поступово активуючись через кілька днів. Вдосконалення технології sandbox включає покращену емуляцію для створення більш реалістичного середовища, bare metal sandbox для усунення артефактів віртуалізації, extended analysis time для підозрілих файлів, інтерактивну емуляцію дій користувача, multi-path execution для охоплення різних гілок коду, інтеграцію з threat intelligence та використання машинного навчання для класифікації поведінки. Відомі рішення включають Cuckoo Sandbox як open-source систему, FireEye з advanced sandboxing, Palo Alto WildFire як хмарний сервіс, Joe Sandbox як комерційну платформу та Any.Run як інтерактивний онлайн-сервіс.

1.3. Огляд існуючих антивірусних рішень та їх можливостей

1.3.1. Комерційні антивірусні продукти

Комерційні антивірусні рішення являють собою комплексні продукти, які поєднують кілька технологій захисту та надають широкий спектр функціональних можливостей, розроблені провідними компаніями з кібербезпеки з багаторічним досвідом досліджень загроз та розробки засобів захисту. Ці продукти зазвичай включають не лише базову антивірусну функціональність, але й додаткові компоненти безпеки, такі як фаєрволи, захист від фішингу, контроль пристроїв, шифрування даних та системи резервного копіювання, створюючи інтегровану екосистему безпеки для комплексного захисту користувачів та організацій.

Norton 360 від Symantec/Broadcom є одним з найпопулярніших антивірусних продуктів для домашніх користувачів та малого бізнесу, що пропонує сигнатурне сканування з великою базою загроз та технологію SONAR для поведінкового аналізу. Продукт включає захист від ransomware з автоматичним резервним копіюванням

критичних файлів, вбудований фаєрвол з інтелектуальним контролем програм, VPN для захищеного з'єднання та менеджер паролів. Переваги Norton включають високий рівень виявлення загроз та зручний інтерфейс, але недоліки проявляються у відносно високому споживанні ресурсів та вартості підписки, що може бути обмежуючим фактором для деяких користувачів.

Kaspersky Endpoint Security є рішенням від лабораторії Касперського, орієнтованим на корпоративний сектор, що використовує багаторівневий захист з сигнатурним, евристичним та поведінковим аналізом. Технологія System Watcher забезпечує моніторинг та відкочування шкідливих дій, тоді як Kaspersky Security Network створює хмарну систему збору та аналізу інформації про загрози [18]. Продукт включає Application Control для контролю запуску програм, Device Control для контролю підключення пристроїв, Web Control для фільтрації веб-контенту та Vulnerability Scanner для пошуку вразливостей у встановленому ПЗ. Висока ефективність виявлення та потужна централізована система управління є головними перевагами, але складність налаштування, вартість ліцензій та політичні обмеження у деяких країнах можуть бути недоліками.

ESET NOD32 від словацької компанії ESET відомий своєю ефективністю та низьким впливом на продуктивність системи, використовуючи технологію ESET LiveGrid для хмарного репутаційного аналізу та Advanced Memory Scanner для виявлення malware з обфускацією. Exploit Blocker забезпечує захист від експлуатації вразливостей, Ransomware Shield надає спеціалізований захист від шифрувальників, а Botnet Protection виявляє та блокує botnet-комунікації. UEFI Scanner є унікальною функцією для сканування UEFI firmware на наявність загроз, що особливо важливо для захисту від буткітів. Низьке споживання ресурсів та швидкість роботи є головними перевагами, але мінімалістичний інтерфейс може здатися складним для початківців [19].

Bitdefender GravityZone представляє комплексне рішення для корпоративної безпеки з потужними можливостями централізованого управління, використовуючи

машинне навчання для виявлення zero-day загроз та Sandbox Analyzer для динамічного аналізу підозрілих файлів. HyperDetect забезпечує поведінковий аналіз на основі машинного навчання, Process Inspector моніторить процеси у реальному часі, а Advanced Threat Control блокує невідомі процеси до підтвердження безпеки. Network Attack Defense захищає від мережевих експлойтів, а модуль EDR надає розширені можливості виявлення та реагування на загрози. Висока ефективність та мінімальний вплив на продуктивність є перевагами, але складність для малих організацій та вартість розширених функцій можуть бути обмежуючими факторами для деяких користувачів.

1.3.2. Спеціалізовані сканери та утиліти

Спеціалізовані сканери та утиліти представляють окрему категорію інструментів безпеки, які фокусуються на виявленні та видаленні специфічних типів загроз або доповнюють традиційні антивірусні рішення додатковими можливостями аналізу. На відміну від комплексних антивірусних пакетів, які намагаються охопити весь спектр загроз, спеціалізовані інструменти зосереджуються на конкретних завданнях, таких як виявлення руткітів, аналіз підозрілих процесів, перевірка цілісності системних файлів або видалення особливо стійкого malware. Ці інструменти часто розробляються як портативні додатки, які не потребують встановлення та можуть запускатися з USB-накопичувачів, що робить їх особливо корисними для аналізу вже інфікованих систем.

Malwarebytes є одним з найпопулярніших спеціалізованих антималварних інструментів, який особливо ефективний для видалення adware, spyware та potentially unwanted programs (PUP), які традиційні антивіруси можуть пропускати. Продукт використовує комбінацію сигнатурного аналізу, евристики та поведінкового моніторингу для виявлення широкого спектру загроз, включаючи руткіти та ransomware. Malwarebytes часто використовується як другий рівень захисту поряд з

основним антивірусом, оскільки його бази загроз та методи виявлення доповнюють традиційні антивірусні рішення. Безкоштовна версія надає можливість сканування на вимогу, тоді як преміум-версія включає захист у реальному часі та автоматичні оновлення.

GMER є потужним інструментом для виявлення та видалення руткітів, який може сканувати систему на наявність прихованих процесів, потоків, модулів, сервісів, файлів, ключів реєстру та ADS (Alternate Data Streams). Програма працює на низькому рівні системи, обходячи можливі перехоплення API, що робить її ефективною проти складних руткітів режиму ядра. GMER також може виявляти модифіковані системні виклики, перехоплення SSDT та IDT, а також надає можливість відновлення модифікованих системних компонентів. Інструмент вимагає певного рівня технічних знань для інтерпретації результатів та прийняття рішень щодо видалення виявлених об'єктів, тому частіше використовується професіоналами з безпеки [20].

Process Explorer та Process Monitor від Microsoft Sysinternals є незамінними інструментами для аналізу активності процесів та системи. Process Explorer надає детальну інформацію про запущені процеси, включаючи їх ієрархію, використані DLL, дескриптори, споживання ресурсів та цифрові підписи, що дозволяє виявляти підозрілі процеси без цифрових підписів або з незвичними характеристиками. Process Monitor записує всі операції файлової системи, реєстру та мережі у реальному часі з можливістю детальної фільтрації та аналізу, що робить його незамінним для розслідування інцидентів безпеки та розуміння поведінки підозрілого ПЗ. Ці інструменти є безкоштовними та широко використовуються як адміністраторами систем, так і дослідниками malware.

Autoruns є іншим інструментом від Sysinternals, який показує всі програми та компоненти, що автоматично запускаються під час завантаження системи або входу користувача. Програма відображає набагато більше точок автозапуску, ніж стандартні системні утиліти Windows, включаючи ключі реєстру, завдання

планувальника, сервіси, драйвери, розширення оболонки, плагіни браузерів та багато іншого. Кожен запис можна перевірити на наявність цифрового підпису, проаналізувати на VirusTotal або вимкнути для діагностики проблем. Autoruns особливо корисний для виявлення malware, який закріплюється в системі через механізми автозапуску, оскільки багато загроз використовують саме ці механізми для збереження присутності в системі після перезавантаження.Спеціалізовані утиліти для видалення конкретних типів malware, такі як Kaspersky Virus Removal Tool, Norton Power Eraser або ESET Online Scanner, надають можливість глибокого сканування системи без необхідності встановлення повноцінного антивірусного пакету. Ці інструменти часто використовують агресивніші методи виявлення, ніж постійно встановлені антивіруси, що може призводити до більшої кількості хибних спрацьовувань, але також підвищує шанси виявлення складно прихованого malware. Онлайн-сканери особливо корисні для перевірки систем, де підозрюється, що встановлений антивірус був вимкнений або обійдений malware.

1.3.3. Порівняльний аналіз ефективності

Порівняльний аналіз ефективності антивірусних рішень є критично важливим для прийняття обґрунтованих рішень щодо вибору засобів захисту, оскільки різні продукти демонструють різні показники виявлення загроз, впливу на продуктивність системи та загальної надійності роботи. Незалежні тестові лабораторії, такі як AV-Test, AV-Comparatives, SE Labs та MRG Effitas, проводять регулярні тестування комерційних антивірусних продуктів за стандартизованими методиками, що дозволяє об'єктивно порівнювати їх ефективність. Ці тестування зазвичай оцінюють три основні категорії: захист від загроз, вплив на продуктивність та зручність використання, включаючи кількість хибних спрацьовувань.

Результати тестувань AV-Test показують, що провідні антивірусні рішення, такі як Bitdefender, Kaspersky, Norton та ESET, постійно демонструють високі показники

виявлення загроз на рівні 99-100% для відомого malware та 95-99% для zero-day загроз у реальних умовах. Тестування включають як виявлення malware на момент спроби запуску, так і виявлення вже активного malware в системі, що дозволяє оцінити ефективність як превентивного захисту, так і remediation можливостей [21]. Важливо відзначити, що показники виявлення можуть значно варіюватися в залежності від типу загрози, наприклад, деякі продукти краще виявляють ransomware, тоді як інші більш ефективні проти banking trojans або spyware.

Вплив на продуктивність системи є критичним фактором, особливо для бізнес-середовищ, де сповільнення роботи може призвести до зниження продуктивності працівників та фінансових втрат. Тестування AV-Comparatives оцінюють вплив антивірусних продуктів на такі операції, як копіювання файлів, архівація та розархівація, встановлення та видалення програм, завантаження веб-сторінок та запуск додатків. Продукти з найнижчим впливом на продуктивність включають ESET NOD32, Kaspersky та Bitdefender, які використовують оптимізовані алгоритми сканування та ефективне кешування для мінімізації споживання ресурсів. Деякі рішення, такі як Windows Defender, показують середній вплив на продуктивність, що може бути прийнятним для багатьох користувачів, враховуючи його безкоштовність та інтеграцію з операційною системою.

Кількість хибних спрацьовувань є важливим показником зручності використання, оскільки надмірна кількість помилкових тривог може призвести до того, що користувачі почнуть ігнорувати попередження або вимикати захист. Тестування показують, що більшість провідних антивірусів мають дуже низьку кількість хибних спрацьовувань на легітимне ПЗ, зазвичай менше 5-10 помилкових детекцій на мільйон перевірених файлів. Проте деякі агресивніші евристичні двигуни або продукти, орієнтовані на максимальну безпеку, можуть мати більшу кількість хибних спрацьовувань, що вимагає від адміністраторів додаткових зусиль для налаштування білих списків та виключень [22, с. 404-409].

Порівняльний аналіз також повинен враховувати специфічні можливості різних продуктів, які можуть бути критично важливими для певних організацій. Наприклад, деякі рішення надають потужні можливості централізованого управління для корпоративних мереж, інші включають розширені EDR функції для розслідування інцидентів, треті пропонують спеціалізований захист для серверів або віртуальних середовищ. Вартість володіння також є важливим фактором, який включає не лише ціну ліцензій, але й витрати на впровадження, навчання персоналу, технічну підтримку та обслуговування. Оптимальний вибір антивірусного рішення залежить від балансу між ефективністю захисту, впливом на продуктивність, функціональністю, зручністю використання та загальною вартістю володіння з урахуванням специфічних потреб та ресурсів організації [23].

1.4. Аналіз недоліків існуючих систем захисту

Незважаючи на значний прогрес у розвитку антивірусних технологій та постійне вдосконалення методів виявлення загроз, існуючі системи захисту мають ряд фундаментальних обмежень та недоліків, які залишають системи вразливими до сучасних кіберзагроз. Основною проблемою традиційних антивірусних рішень є їх реактивний характер, оскільки більшість систем захисту базуються на знанні про вже існуючі загрози та їх сигнатурах, що створює вікно вразливості між появою нової загрози та моментом, коли антивірусні вендори створять та розповсюдять відповідну сигнатуру. У випадку цільових атак, де malware розробляється спеціально для конкретної організації, це вікно може тривати тижнями або місяцями, оскільки таке унікальне злоякісне ПЗ може ніколи не потрапити до аналітиків антивірусних компаній.

Проблема поліморфізму та метаморфізму залишається серйозним викликом для систем захисту, оскільки сучасне malware здатне автоматично модифікувати свій код при кожному поширенні, створюючи тисячі унікальних варіантів з різними

сигнатурами, але однаковою функціональністю. Хоча евристичний та поведінковий аналіз частково вирішують цю проблему, зловмисники постійно розробляють нові техніки обходу детектування, такі як відкладене виконання злоякісного коду, виявлення віртуальних середовищ та sandbox, використання легітимних системних інструментів для злоякісних цілей (living off the land) та складні багатоступеневі атаки, де кожен етап виглядає відносно безпечним, але їх комбінація призводить до компрометації системи.

Обмежена видимість та недостатня інтеграція різних компонентів безпеки створюють прогалини в захисті, оскільки традиційні антивірусні рішення зазвичай фокусуються на захисті окремих endpoint-пристроїв, не маючи повної видимості мережевого трафіку, серверної інфраструктури або хмарних сервісів. Це дозволяє зловмисникам використовувати lateral movement для поширення в мережі після початкового проникнення, експлуатувати вразливості в незахищених або погано захищених компонентах інфраструктури та проводити тривалі кампанії збору інформації, залишаючись непоміченими. Відсутність централізованого моніторингу та кореляції подій з різних джерел ускладнює виявлення складних багатоступеневих атак, де кожен окремий компонент може виглядати безпечним, але їх послідовність вказує на зловмисну активність.

Проблема балансу між безпекою та продуктивністю залишається актуальною, оскільки більш агресивні методи виявлення загроз, такі як глибоке сканування всього трафіку, постійний моніторинг всіх системних викликів або детальний аналіз кожного файлу в sandbox, вимагають значних обчислювальних ресурсів та можуть істотно сповільнювати роботу системи. Це змушує організації йти на компроміси, вимикаючи деякі функції захисту або обмежуючи глибину моніторингу для підтримання прийнятної продуктивності, що створює можливості для зловмисників експлуатувати ці прогалини в захисті. Особливо гостро ця проблема проявляється в середовищах з високими вимогами до продуктивності, таких як торгові системи, промислові системи керування або системи реального часу.

Людський фактор залишається найслабшою ланкою в ланцюгу безпеки, незважаючи на всі технологічні вдосконалення систем захисту. Соціальна інженерія дозволяє зловмисникам обходити технічні засоби захисту, переконуючи користувачів виконати шкідливий код, надати облікові дані або вимкнути системи безпеки. Недостатня обізнаність користувачів про кіберзагрози, неправильне налаштування систем захисту адміністраторами, використання слабких паролів та ігнорування базових практик безпеки створюють численні вектори атак. Навіть найсучасніші технічні рішення виявляються неефективними, якщо користувачі свідомо або несвідомо сприяють проникненню malware в систему, що підкреслює необхідність комплексного підходу до безпеки, який включає не лише технічні засоби, але й організаційні заходи, навчання персоналу та формування культури безпеки.

Висновки до розділу 1

У першому розділі проведено комплексний аналіз сучасних загроз злоякісного програмного забезпечення та існуючих методів захисту від них. Встановлено, що ландшафт кіберзагроз характеризується високим рівнем різноманітності та постійною еволюцією, де злоякісне ПЗ набуває все більш складних форм та механізмів уникнення виявлення. Класифікація malware включає віруси та черв'яки з механізмами самовідтворення, троянські програми та шпигунське ПЗ для викрадення інформації, програми-вимагачі для вимагання викупу та руткіти з буткітами для приховування присутності в системі, при цьому сучасні загрози часто поєднують характеристики різних типів malware, створюючи складні багатокомпонентні системи атак.

Аналіз сучасних методів виявлення та протидії зловмисному ПЗ показав, що кожен з підходів має свої переваги та обмеження, що вимагає їх комбінування для забезпечення ефективного захисту. Сигнатурний аналіз забезпечує швидке та точне виявлення відомих загроз, але неефективний проти нових та модифікованих зразків

malware. Евристичний аналіз дозволяє виявляти раніше невідомі загрози на основі підозрілих характеристик, але може призводити до хибних спрацьовувань та вимагає ретельного налаштування. Поведінковий аналіз надає можливість виявлення zero-day загроз та складного malware через моніторинг системної активності, але має реактивний характер та споживає значні ресурси. Технології sandbox забезпечують безпечний аналіз підозрілого ПЗ в ізольованому середовищі, але вразливі до техніки обходу детектування та обмежені в часі аналізу.

Огляд існуючих антивірусних рішень виявив, що провідні комерційні продукти, такі як Kaspersky, Bitdefender, ESET та Norton, демонструють високу ефективність виявлення загроз на рівні 99-100% для відомого malware, використовуючи комбінацію різних технологій захисту та машинне навчання для покращення детектування. Спеціалізовані сканери та утиліти доповнюють традиційні антивірусні рішення, фокусуючись на виявленні специфічних типів загроз або надаючи додаткові можливості аналізу системи. Порівняльний аналіз показав, що вибір оптимального рішення залежить від балансу між ефективністю захисту, впливом на продуктивність, функціональністю та загальною вартістю володіння з урахуванням специфічних потреб організації.

Виявлено фундаментальні недоліки існуючих систем захисту, включаючи реактивний характер більшості рішень, що створює вікно вразливості між появою нової загрози та створенням сигнатури, обмежену видимість та недостатню інтеграцію різних компонентів безпеки, проблему балансу між безпекою та продуктивністю, а також вразливість до складних техніків обходу детектування та соціальної інженерії. Ці обмеження підкреслюють необхідність розробки удосконалених систем захисту, які поєднують проактивний моніторинг системної активності, інтеграцію різних джерел інформації про загрози та можливість виявлення аномальної поведінки без значного впливу на продуктивність системи, що буде розглянуто в наступних розділах роботи.

РОЗДІЛ 2. РОЗРОБКА КОНЦЕПЦІЇ УДОСКОНАЛЕНОЇ СИСТЕМИ ПРОТИДІЇ ЗЛОЯКІСНОМУ ПЗ

2.1. Вимоги до системи захисту від сучасних загроз

Сучасні системи захисту від злоякісного програмного забезпечення мають відповідати комплексу вимог, які випливають як з аналізу актуальних загроз, так і з практичних потреб організацій різного масштабу.

Система захисту повинна забезпечувати багаторівневий моніторинг різних векторів атак одночасно, охоплюючи файлову систему, процеси, мережеву активність, реєстр Windows та інші системні компоненти. На відміну від класичних сигнатурних методів, необхідно виявляти підозрілу поведінку навіть за відсутності відомих сигнатур зловмисного програмного забезпечення, що є критично важливим для протидії загрозам нульового дня та поліморфним зразкам шкідливих програм.

Моніторинг не повинен суттєво сповільнювати роботу системи. За даними аналітичних досліджень, прийнятним вважається навантаження не більше п'яти-семи відсотків процесорного часу та близько двохсот мегабайтів оперативної пам'яті для типового настільного комп'ютера. Система має формувати базову лінію нормальної поведінки для конкретного середовища та адаптувати пороги виявлення відповідно до профілю використання [24].

Важливою характеристикою є модульність архітектури, коли окремі компоненти функціонують незалежно, що забезпечує стабільність системи навіть при збої одного з модулів. Система має працювати цілодобово з високим показником доступності, що відповідає мінімальному часу простою на рік. Архітектура має передбачати можливість розгортання як на окремих робочих станціях, так і в корпоративних мережах з централізованим управлінням.

Підтримка сучасних версій операційних систем Windows з можливістю інтеграції з існуючими засобами захисту, включаючи антивіруси, системи управління

інформацією та подіями безпеки, а також рішення для виявлення та реагування на загрози кінцевих точок, є обов'язковою вимогою. Компоненти моніторингу мають бути захищені від втручання зловмисного програмного забезпечення, зокрема через запуск з підвищеними привілеями та самоконтроль цілісності. Інтерфейс має бути інтуїтивно зрозумілим для адміністраторів безпеки з різним рівнем кваліфікації, з можливістю швидкого реагування на сповіщення.

Згідно з методологією класифікації тактик та технік противника, система має виявляти дії на різних етапах кібератаки, починаючи від початкового доступу через експлуатацію вразливостей та фішингові вкладення, проходячи через виконання підозрілих скриптів, забезпечення закріплення в системі через автозапуск та служби, підвищення привілеїв, ухилення від захисту через обфускацію та впровадження коду, до викрадення облікових даних, розвідки мережі, бічного переміщення, збирання та вилучення даних, і завершуючи впливом через шифрування або знищення інформації [25].

Таблиця 2.1

Функціональні вимоги до системи виявлення загроз

№	Вимога	Опис	Пріоритет	Критерій виконання
FR-1	Моніторинг файлової системи	Відстеження операцій створення, модифікації, видалення файлів в реальному часі	Високий	Фіксація >95% файлових подій з затримкою <100мс
FR-2	Моніторинг процесів	Контроль життєвого циклу процесів, батьківсько-дочірніх зв'язків, міжпроцесних взаємодій	Високий	Виявлення 100% створених процесів з метаданими
FR-3	Перевірка цифрових підписів	Верифікація Authenticode підписів виконуваних файлів	Середній	Перевірка підписів для 100% нових executable
FR-4	Виявлення ransomware	Детекція масового шифрування файлів, ransom notes	Критичний	MTTD <30 секунд, TPR >95%, FPR <0.5%

FR-5	Виявлення process injection	Детекція DLL injection, process hollowing, APC injection	Високий	TPR >85% для основних технік injection
FR-6	Виявлення credential dumping	Контроль доступу до LSASS та інших критичних процесів	Критичний	TPR >95%, MTTD <5 секунд
FR-7	Формування baseline	Автоматичне навчання профілю нормальної поведінки	Високий	Стабільний baseline після 7-14 днів роботи
FR-8	Виявлення аномалій	Статистичний аналіз відхилень від baseline	Високий	Виявлення аномалій з відхиленням >3σ
FR-9	Застосування правил детекції	Rule-based виявлення на основі MITRE ATT&CK	Високий	Підтримка >50 правил з можливістю додавання
FR-10	Інтегральна оцінка ризику	Агрегація сигналів від різних детекторів	Середній	Risk score 0-100 з градацією severity
FR-11	Генерація алертів	Створення повідомлень про виявлені загрози	Високий	Алерти з повним контекстом та рекомендаціями
FR-12	Візуалізація активності	Графіки метрик, статистика, timeline подій	Середній	Real-time оновлення з частотою 1 Гц

Для оцінки якості роботи системи використовуються показники частки коректно виявлених загроз серед усіх реальних інцидентів з цільовим значенням понад дев'яносто п'ять відсотків, частки хибних спрацювань серед легітимної активності з цільовим значенням менше одного відсотка, середнього часу від початку атаки до її виявлення з цільовим значенням менше п'яти хвилин для критичних загроз, середнього часу реагування на інцидент з цільовим значенням менше п'ятнадцяти хвилин, та відсотка технік, що виявляються системою, з цільовим значенням понад сімдесят відсотків [26].

2.2. Архітектура удосконаленої системи захисту

2.2.1. Модуль моніторингу файлової системи

Модуль моніторингу файлової системи відстежує операції створення, зміни, видалення та перейменування файлів і каталогів по всій системі. Особлива увага приділяється критичним системним каталогам, таким як папки операційної системи Windows, каталоги встановлених програм та профілі користувачів, оскільки саме ці місця найчастіше стають ціллю зловмисного програмного забезпечення.

Для Windows існує кілька технологічних підходів до моніторингу файлової системи, кожен з яких має свої переваги та обмеження. Функція читання змін у каталозі надає асинхронні повідомлення про зміни в папці з підтримкою фільтрів за типом події. Мінідрайвер файлової системи працює на рівні ядра операційної системи, перехоплюючи запити введення-виведення, що забезпечує найповнішу картину активності, але вимагає розробки драйвера з цифровим підписом. Трасування подій для Windows являє собою вбудовану систему трасування подій, що дозволяє підписатися на постачальників рівня ядра, забезпечуючи оптимальний баланс між продуктивністю та повнотою інформації.

У розробленій системі використовується комбінований підхід, де трасування подій застосовується для загального моніторингу файлової системи з прийнятним рівнем накладних витрат, а читання змін у каталозі використовується для критичних папок, що потребують більш детального відстеження з високою частотою оновлення. Такий гібридний метод дозволяє ефективно балансувати між повнотою моніторингу та продуктивністю системи.

Модуль застосовує набір евристик для виявлення підозрілої активності. Масове створення файлів з однотипними розширеннями за короткий проміжок часу може свідчити про роботу програми-здирника, що шифрує дані користувача. Зміна виконуваних файлів у системних каталогах без ініціації від легітимного встановлювача, що ідентифікується через аналіз батьківського процесу, може вказувати на спробу підміни системних компонентів. Створення файлів з подвійним

розширенням часто використовується для маскування зловмисних виконуваних файлів під документи. Запис у каталоги, що належать іншим процесам, може свідчити про впровадження коду в процес або перехоплення виконання.

Виявлення програм-здирників базується на аналізі декількох характерних ознак одночасно. Швидка зміна великої кількості файлів, створення так званих записок про викуп з інструкціями щодо оплати, зміна розширень файлів на нестандартні, та спроби видалення резервних копій через системні утиліти формують сигнатурний патерн поведінки програм-шифрувальників. При виявленні такої комбінації ознак система генерує критичне сповіщення і може автоматично ініціювати процедури реагування [27].

Модуль веде детальну статистику за типами файлових операцій, темпом їх виконання та розподілом по каталогах, що використовується для формування базової лінії нормальної активності. Ця інформація дозволяє виявляти аномалії не тільки на рівні окремих подій, але й на рівні загальних тенденцій використання файлової системи.

2.2.2. Модуль контролю цілісності процесів

Модуль контролю процесів відстежує повний життєвий цикл процесів в операційній системі, включаючи їх створення, завершення, зміни пріоритету та споживання ресурсів. При створенні нового процесу система фіксує його унікальний ідентифікатор та ідентифікатор батьківського процесу, повний шлях до виконуваного файлу, параметри командного рядка, користувача та час створення. Ця інформація формує основу для аналізу батьківсько-дочірніх зв'язків та виявлення аномальних ланцюжків виконання.

Аналіз батьківсько-дочірніх зв'язків дозволяє виявити багато типів зловмисної активності. Наприклад, запуск оболонки PowerShell від імені документа Excel часто свідчить про виконання зловмисних макросів. Запуск виконуваних файлів з

тимчасових каталогів нерідко пов'язаний з розпакуванням та виконанням завантажених корисних навантажень. Процеси без батька можуть вказувати на застосування технік порожнистого процесу або впровадження коду, коли зловмисники створюють легітимні процеси та підміняють їх код.

Виявлення міжпроцесної зміни є критично важливою функцією модуля. Зловмисне програмне забезпечення часто використовує техніки впровадження коду в інші процеси для приховання своєї активності та обходу захисних механізмів. Впровадження бібліотеки передбачає впровадження власної динамічної бібліотеки через створення віддаленого потоку в іншому процесі. Техніка порожнистого процесу включає створення легітимного процесу у призупиненому стані, заміну його коду на зловмисний та подальший запуск. Впровадження через асинхронні виклики процедур використовує механізм асинхронних викликів для виконання коду в контексті іншого процесу [28, с. 139-149].

Модуль відстежує виклики системних функцій для роботи з пам'яттю інших процесів, включаючи відкриття дескриптора процесу, виділення пам'яті, запис даних та створення віддалених потоків. Особлива увага приділяється спробам зміни критичних системних процесів, відповідальних за безпеку, управління службами або автентифікацію користувачів.

Контроль споживання ресурсів дозволяє виявити певні типи зловмисного програмного забезпечення через їх аномальну ресурсоємність. Програми для добування криптовалюти проявляють себе через високе завантаження процесора, боти для розподілених атак відмови в обслуговуванні через інтенсивну мережеву активність, а шпигунські програми через постійну роботу з диском при збиранні інформації. Модуль збирає метрики використання процесорного часу, оперативної пам'яті та мережевого трафіку для кожного процесу, порівнюючи їх з історичними даними та виявляючи статистично значущі відхилення [29].

Формування профілю нормальної поведінки для різних типів процесів дозволяє відрізнити легітимну ресурсоємну активність від зловмисної. Наприклад, високе

навантаження на процесор від відеоредактора є нормальним, тоді як аналогічна активність від текстового редактора вимагає додаткового дослідження.

2.2.3. Модуль перевірки цифрових підписів

Модуль перевірки цифрових підписів аналізує виконувані файли на наявність валідного цифрового підпису від довіреного видавця. Легітимне комерційне програмне забезпечення зазвичай підписується розробником за допомогою сертифіката від визнаного центру сертифікації, що дозволяє підтвердити автентичність та цілісність файлу.

Процес верифікації включає декілька послідовних етапів. Спочатку здійснюється вилучення сертифіката з виконуваного файлу через системні функції Windows. Потім перевіряється ланцюжок довіри, тобто чи був сертифікат виданий центром сертифікації, що присутній у сховищі довірених кореневих центрів операційної системи. Наступним кроком є перевірка часових меж дії сертифіката та його статусу відкликання через списки відкликаних сертифікатів або протокол онлайн-перевірки статусу [30].

Система категоризує файли на декілька груп довіри. Файли з валідним підписом від відомих довірених видавців, таких як Майкрософт, Адобі, Гугл чи інші великі технологічні компанії, отримують статус довірених. Файли без підпису або з невалідним підписом класифікуються як недовірені та потребують посиленого моніторингу. Файли з валідним підписом від невідомих видавців отримують проміжний статус і вимагають додаткового аналізу поведінки.

Запуск непідписаних виконуваних файлів з системних каталогів операційної системи є серйозною аномалією, оскільки всі легітимні системні компоненти Windows мають цифрові підписи Майкрософт. Використання самопідписаних сертифікатів часто пов'язане зі зловмисним програмним забезпеченням, що намагається створити ілюзію легітимності. Прострочені або відкликані сертифікати

можуть свідчити про компрометацію ключа підпису або застосування старих зразків шкідливих програм.

Модуль інтегрується з іншими компонентами системи моніторингу, надаючи інформацію про статус довіри для кожного запущеного процесу. Ця інформація використовується при оцінці ризиків та прийнятті рішень про необхідність додаткового аналізу чи блокування підозрілої активності. Система підтримує налаштовуваний білий список довірених видавців та чорний список відомих зловмисних сертифікатів, що оновлюється через інтеграцію з джерелами розвідувальної інформації про загрози [31].

2.2.4. Модуль виявлення аномальної поведінки

Модуль аналізу аномалій є центральним компонентом системи, що агрегує дані від усіх спеціалізованих моніторів та застосовує комплекс методів для виявлення відхилень від нормальної поведінки. На відміну від традиційних сигнатурних підходів, що базуються на знанні конкретних патернів зловмисного програмного забезпечення, аномальний аналіз дозволяє виявляти раніше невідомі загрози через порушення встановлених норм активності.

Виявлення на основі базової лінії формує профіль нормальної активності для кожної кінцевої точки протягом фази навчання, що зазвичай триває один-два тижні. Профіль включає статистичні характеристики різних аспектів поведінки системи, такі як середня інтенсивність створення процесів, типові батьківсько-дочірні зв'язки, звичайні місця для запуску програм, середнє навантаження на системні ресурси та частота файлових операцій у різних каталогах.

Після формування базової лінії система безперервно порівнює поточну активність з встановленими нормами, використовуючи статистичні методи для визначення значущості відхилень. Перевищення порогу, що зазвичай встановлюється на рівні двох-трьох стандартних відхилень від середнього значення, ініціює

генерацію сповіщення [32, с. 193-198]. Така методологія дозволяє автоматично адаптуватися до специфіки конкретного середовища, враховуючи різницю в профілях використання між, наприклад, робочою станцією розробника та комп'ютером бухгалтера.

Виявлення на основі правил доповнює статистичний аналіз експертними правилами, що кодують знання про типові техніки атак. Ці правила базуються на досвіді аналізу реальних інцидентів безпеки та включають виявлення бічного переміщення через нестандартне використання інструментів віддаленого управління, викрадення облікових даних через доступ до процесів, що зберігають автентифікаційні дані, та програм-здирників через специфічні патерни масової зміни файлів.

Евристичний аналіз виявляє підозрілі комбінації поведінки, що окремо можуть бути легітимними, але разом формують індикатор компрометації. Наприклад, текстовий редактор, що створює мережеве з'єднання, записує виконуваний файл і запускає його, демонструє нетипову для цього класу програм поведінку, що вимагає додаткового розслідування. Евристичний аналіз дозволяє виявляти складні багатоетапні атаки, де жоден окремий крок не є очевидно зловмисним [33].

Архітектура модуля включає три незалежні аналітичні рушії для аналізу на основі базової лінії, правил та евристик, що працюють паралельно. Результати їх роботи агрегуються компонентом оцінки ризику, що обчислює інтегральну оцінку для кожної події на основі зважених сигналів від різних рушіїв. Лише події з оцінкою ризику вище заданого порогу призводять до генерації сповіщень, що мінімізує кількість хибних спрацювань і зменшує навантаження на адміністраторів безпеки [34].

Модуль також реалізує механізм зворотного зв'язку, де реакція адміністратора на сповіщення використовується для підстроювання порогів та ваг різних сигналів. Підтвердження сповіщення як справжнього інциденту призводить до підвищення чутливості до аналогічних патернів, тоді як позначення як хибного спрацювання

знижує вагу відповідних сигналів. Така адаптивна поведінка дозволяє системі безперервно покращувати якість виявлення в конкретному середовищі.

2.3. Методи виявлення підозрілої активності

2.3.1. Моніторинг змін у файловій системі

Трасування подій для Windows являє собою низькорівневу систему журналювання, вбудовану безпосередньо в ядро операційної системи. Для моніторингу файлової системи використовується спеціалізований постачальник рівня ядра, що генерує події для операцій створення нових файлів, їх видалення, зміни атрибутів та розміру. Підписка на ці події дозволяє отримувати інформацію про файлову активність з мінімальною затримкою та незначним впливом на продуктивність системи.

Щоб уникнути перевантаження системи надмірною кількістю записів у журналі, застосовується багаторівнева система фільтрації. Системні каталоги з природньо високою активністю, такі як папки тимчасових файлів, кеші браузерів та служби попереднього завантаження, виключаються з детального моніторингу. Процеси з валідним цифровим підписом від Майкрософт або інші довірені системні компоненти отримують знижений рівень уваги. Однотипні події агрегуються в узагальнені записи замість створення окремого запису в журналі для кожної операції.

Виявлення програм-здирників базується на аналізі специфічних патернів поведінки, що відрізняють програми-шифрувальники від легітимного програмного забезпечення. Швидка зміна великої кількості файлів, яка відбувається зі швидкістю сотень файлів за хвилину, є першою ознакою. Створення текстових файлів з інструкціями щодо викупу у багатьох каталогах одночасно формує другий індикатор. Зміна розширень файлів на нестандартні або додавання додаткових розширень

створює третю ознаку. Виконання системних команд для видалення резервних копій завершує типовий сценарій атаки програм-здирників.

Система відстежує темп файлових операцій для кожного процесу, порівнюючи його з історичними даними та встановленими порогами. При перевищенні критичного значення ініціюється додаткова перевірка на наявність інших індикаторів програм-здирників. Виявлення декількох ознак одночасно призводить до генерації критичного сповіщення та може автоматично ініціювати процедури аварійного реагування, такі як зупинка підозрілого процесу, ізоляція системи від мережі або створення знімка файлової системи для можливості відновлення.

Моніторинг також включає відстеження спроб зміни критичних системних файлів, створення файлів у каталогах автозапуску, та запису виконуваних файлів у місця з підвищеним ризиком. Комбінація цих методів забезпечує багаторівневий захист файлової системи від різних типів загроз.

2.3.2. Виявлення міжпроцесної модифікації даних

Міжпроцесна модифікація є однією з найпоширеніших технік, що використовується зловмисним програмним забезпеченням для приховування своєї присутності та обходу захисних механізмів. Типовий сценарій впровадження динамічної бібліотеки включає послідовність системних викликів, що починається з отримання дескриптора цільового процесу з максимальними правами доступу, продовжується виділенням пам'яті в адресному просторі цільового процесу, записом шляху до зловмисної бібліотеки у виділену пам'ять, та завершується створенням віддаленого потоку, що викликає функцію завантаження бібліотеки.

Спеціалізований постачальник трасування подій для розвідки загроз генерує події при виконанні підозрілих операцій з пам'яттю. Виклики функцій виділення пам'яті в іншому процесі, створення проекцій секцій пам'яті, зміна прав доступу до пам'яті іншого процесу, та додавання асинхронних викликів процедур до черг інших

потоків відстежуються на рівні ядра операційної системи. Це дозволяє виявляти спроби впровадження навіть від процесів, що намагаються приховати свою активність.

При виявленні міжпроцесної операції з пам'яттю система аналізує рівень довіри до процесу-ініціатора через перевірку його цифрового підпису та порівняння з білим списком довірених компонентів. Якщо недовірений процес намагається змінити критичний системний процес, генерується сповіщення високого пріоритету. Особлива увага приділяється спробам втручання у процеси, відповідальні за автентифікацію, управління службами безпеки або інші критичні функції операційної системи.

Техніка порожнистого процесу виявляється через послідовність характерних дій, що включають створення процесу у призупиненому стані, що саме по собі є рідкісною операцією для легітимного програмного забезпечення. Подальше демонтування оригінальних секцій пам'яті процесу та запис нового коду формують чіткий патерн атаки. Відновлення виконання процесу після таких модифікацій практично однозначно вказує на зловмисну діяльність.

Система також відстежує спроби запису файлів процесом у каталоги, що не належать до його власної інсталяції. Наприклад, текстовий редактор не має легітимних причин створювати файли в каталозі браузера. Така активність між каталогами може свідчити про спробу встановлення зловмисних розширень, підміни конфігураційних файлів або реалізації інших форм закріплення через модифікацію файлів інших програм.

Комбінований аналіз міжпроцесних взаємодій на рівні пам'яті та файлової системи забезпечує всебічне виявлення спроб впровадження та перехоплення, навіть коли атакуючі використовують комбінації різних технік для обходу окремих методів виявлення.

2.3.3. Контроль доступу до пам'яті інших процесів

Доступ до пам'яті критичних системних процесів є однією з ключових ознак просунутої атаки, особливо при спробах викрадення автентифікаційних даних. Процес підсистеми локального органу безпеки зберігає хеші паролів, квитки протоколу автентифікації та інші автентифікаційні матеріали користувачів, що робить його привабливою ціллю для зловмисників.

Типові утиліти для викрадення облікових даних працюють через читання пам'яті процесу підсистеми локального органу безпеки та витягування з неї паролів у відкритому вигляді або хешів. Легітимні системні утиліти також можуть бути зловжиті для створення дампу пам'яті цільового процесу з подальшим аналізом в автономному режимі. Навіть вбудовані компоненти Windows можуть бути експлуатовані через нестандартні параметри виклику для створення повних дампів критичних процесів.

Система моніторингу відстежує всі спроби отримання доступу до пам'яті процесу підсистеми безпеки, особливо з правами читання. При виявленні такої спроби аналізується рівень довіри до процесу-ініціатора через перевірку цифрового підпису та порівняння з білим списком програм, що мають легітимні причини для такого доступу. Антивірусні рушії та агенти виявлення та реагування на загрози включаються до цього білого списку, тоді як більшість іншого програмного забезпечення не має потреби взаємодіяти з процесом підсистеми безпеки [35].

Додатковий аналіз включає пошук специфічних патернів поведінки, характерних для утиліт викрадення облікових даних. Імпорт певних системних функцій для роботи з підсистемою локального органу безпеки, наявність характерних текстових рядків у пам'яті процесу, та специфічні патерни читання пам'яті формують додаткові індикатори компрометації. Виявлення комбінації цих ознак може ініціювати автоматичну зупинку підозрілого процесу для запобігання витоку облікових даних.

Окрім підсистеми локального органу безпеки, моніторингу підлягають інші критичні системні процеси, включаючи підсистему виконання клієнт-сервер, що відповідає за базові функції підсистеми Windows, процес управління службами, та процес автентифікації користувачів. Будь-який несанкціонований доступ до цих компонентів з боку недовіреного програмного забезпечення розглядається як потенційна загроза безпеці системи [36-37].

Реалізація багаторівневого контролю доступу до критичних процесів значно ускладнює успішну реалізацію атак типу викрадення облікових даних та підвищує ймовірність раннього виявлення спроб компрометації системи автентифікації.

2.3.4. Аналіз частоти створення процесів

Аномально висока частота створення нових процесів може свідчити про різні типи зловмисної активності. Автоматизована експлуатація вразливостей часто включає множинні спроби виконання коду з різними параметрами. Бічне переміщення через мережу корпорації зазвичай супроводжується створенням процесів на віддалених системах через інструменти віддаленого виконання. Атаки типу вилочної бомби намагаються вичерпати системні ресурси через експоненційне збільшення кількості процесів.

Формування базової лінії нормальної активності відбувається протягом тижня-двох початкової роботи системи моніторингу. Збирається статистика про інтенсивність створення процесів у різні часи доби, оскільки профіль використання суттєво відрізняється між робочими та неробочими годинами. Обчислюються середні значення та стандартні відхилення для різних часових інтервалів, що дозволяє врахувати природні коливання активності.

Для згладжування випадкових флуктуацій застосовується метод експоненційно зваженого ковзного середнього, що надає більшу вагу нещодавнім спостереженням при збереженні впливу історичних даних. Це дозволяє базовій лінії поступово

адаптуватися до змін у характері використання системи, таких як встановлення нового програмного забезпечення або зміна робочих процедур користувача.

Виявлення аномалій базується на обчисленні статистичної значущості відхилення поточної інтенсивності від очікуваної. Використовується показник, що вимірює відстань від середнього значення у одиницях стандартного відхилення. Перевищення порогу у три стандартні відхилення, що відповідає ймовірності випадкового виникнення менше однієї десятої відсотка, ініціює генерацію сповіщення.

Система також аналізує не тільки загальну кількість, але й характер створюваних процесів. Множинні запуски одного й того ж виконуваного файлу за короткий час, особливо якщо це системні утиліти командного рядка або інструменти адміністрування, може вказувати на автоматизований скрипт атаки. Створення довгих ланцюжків батьківсько-дочірніх процесів, що нетипово для легітимних програм, також розглядається як потенційний індикатор компрометації.

Для запобігання надмірній кількості сповіщень при тривалих аномаліях реалізований механізм обмеження частоти, що обмежує частоту повідомлень про один і той же тип аномалії. Після генерації сповіщення встановлюється період охолодження, протягом якого повторні спрацювання того ж правила не призводять до нових повідомлень. Це дозволяє адміністратору зосередитися на розслідуванні виявленого інциденту без відволікання на дублюючі повідомлення.

Комбінований аналіз кількісних метрик та якісних характеристик створюваних процесів забезпечує ефективне виявлення як масштабних атак з високою інтенсивністю, так і цільових атак з обмеженою, але підозрілою активністю.

2.4. Інтеграція з існуючими засобами захисту

Розроблена система не призначена для заміни існуючих засобів захисту інформації, а для доповнення їх функціональності через поведінковий аналіз та

моніторинг на рівні операційної системи. Інтеграція з іншими компонентами інфраструктури безпеки відбувається на декількох рівнях.

Взаємодія з антивірусними рішеннями реалізується через стандартні програмні інтерфейси центру безпеки Windows, що надають доступ до статусу вбудованого захисника системи. Система може отримувати інформацію про стан захисту в реальному часі, дату останнього оновлення сигнатур, результати останнього сканування та інші параметри роботи антивірусного рушія. Виявлення вимкненого захисту або застарілих сигнатур генерує відповідні повідомлення для адміністратора.

Експорт журналів та сповіщень у централізовані системи моніторингу безпеки здійснюється через підтримку стандартизованих форматів, включаючи системний журнал для традиційних систем, спільний формат подій для інтеграції з корпоративними рішеннями з управління інформацією та подіями безпеки, та нотацію об'єктів JavaScript для сучасних платформ аналітики. Це дозволяє агрегувати дані з множини кінцевих точок та проводити кореляційний аналіз для виявлення розподілених атак.

Синхронізація білого списку довірених процесів між розробленою системою та іншими засобами захисту зменшує кількість хибних спрацювань. Якщо файл позначений як довірений в антивірусній базі, ця інформація може автоматично враховуватися при оцінці ризиків у системі поведінкового аналізу. Аналогічно, виявлення нового підозрілого процесу може ініціювати його перевірку антивірусним сканером.

Для корпоративних середовищ система може функціонувати як агент кінцевої точки в архітектурі виявлення та реагування на загрози. Безперервна передача телеметрії на серверну частину дозволяє проводити міжточкову кореляцію для виявлення бічного переміщення та інших розподілених атак. Серверна частина може запитувати конкретні агенти для виконання цільових перевірок, таких як пошук специфічних файлів, аналіз конкретних процесів або збір додаткових криміналістичних даних.

Механізм автоматичного реагування дозволяє серверній частині віддавати команди агентам кінцевих точок для ізоляції скомпрометованих систем, зупинки зловмисних процесів або збору доказової інформації для подальшого розслідування. Реалізація таких можливостей перетворює систему з пасивного інструменту моніторингу на активний компонент інфраструктури реагування на інциденти.

Інтеграція з джерелами розвідувальної інформації про загрози дозволяє збагачувати локальний аналіз глобальною інформацією про актуальні загрози. Списки відомих зловмисних адрес інтернет-протоколу та доменів використовуються для виявлення спроб зв'язку з серверами команд та контролю. Бази контрольних сум відомих зловмисних файлів дозволяють миттєво ідентифікувати присутність відомих шкідливих програм без необхідності поведінкового аналізу. Правила для сканування надають можливість сигнатурного пошуку специфічних патернів як у файлах на диску, так і в пам'яті процесів.

Координація з мережевими засобами захисту дозволяє реалізувати багаторівневу оборону. При виявленні підозрілого процесу на кінцевій точці система може ініціювати блокування його мережевої активності на рівні брандмауера. Аналіз запитів системи доменних імен на наявність доменів, згенерованих через алгоритм генерації доменів, що типово для сучасних ботнетів, забезпечує раннє виявлення інфікованих систем. У критичних ситуаціях можлива повна ізоляція кінцевої точки від мережі через програмне відключення мережевих інтерфейсів або створення блокуючих правил брандмауера.

Така багаторівнева інтеграція перетворює окремі компоненти інфраструктури безпеки на узгоджену систему захисту, де кожен елемент доповнює можливості інших та компенсує їх обмеження. Сигнатурні методи антивірусів ефективні проти відомих загроз, поведінковий аналіз виявляє нові варіанти атак, мережевий моніторинг контролює бічне переміщення, а розвідувальна інформація про загрози забезпечує контекст для оцінки виявлених індикаторів.

Висновки до розділу 2

У другому розділі була розроблена концепція удосконаленої системи протидії злоякісному програмному забезпеченню, що базується на модульній архітектурі та багаторівневому моніторингу.

Сформульовані вимоги охоплюють функціональні аспекти системи, включаючи необхідність виявлення поведінкових аномалій, мінімального впливу на продуктивність, адаптивності до конкретного середовища та модульності архітектури. Визначені нефункціональні вимоги забезпечують надійність роботи, масштабованість рішення та сумісність з існуючими засобами захисту. Встановлені метрики ефективності надають кількісні критерії для оцінки якості виявлення загроз.

Запропонована архітектура побудована як чотирирівнева ієрархічна структура, де системний рівень забезпечує взаємодію з операційною системою, рівень моніторингу виконує збір даних через спеціалізовані модулі, рівень обробки та аналізу здійснює виявлення аномалій, а рівень користувацького інтерфейсу надає засоби управління та візуалізації. Модульна організація забезпечує незалежність компонентів, що підвищує надійність та спрощує розвиток системи.

Розроблені модулі моніторингу покривають критичні аспекти безпеки операційної системи. Модуль файлової системи відстежує операції з файлами через трасування подій для Windows, застосовуючи фільтрацію шумних подій та евристики для виявлення програм-здирників через аналіз темпу змін, створення записок про викуп та зміни розширень файлів. Модуль контролю процесів моніторить їх життєвий цикл, виявляє аномальні батьківсько-дочірні зв'язки та міжпроцесні техніки впровадження через відстеження системних викликів для роботи з пам'яттю. Модуль перевірки підписів верифікує автентичність виконуваних файлів через аналіз цифрових сертифікатів та ланцюжків довіри. Модуль виявлення аномалій агрегує інформацію від інших компонентів, формує базову лінію нормальної поведінки та застосовує статистичні, rule-based та евристичні методи для виявлення відхилень.

Детально описані методи виявлення специфічних типів загроз. Моніторинг файлової системи через постачальників трасування подій дозволяє відстежувати операції на рівні ядра з мінімальними накладними витратами. Виявлення міжпроцесної модифікації базується на аналізі послідовностей системних викликів, характерних для впровадження бібліотеки, техніки порожнистого процесу та інших технік впровадження коду. Контроль доступу до пам'яті критичних процесів, особливо підсистеми локального органу безпеки, запобігає викраденню автентифікаційних даних. Аналіз частоти створення процесів через формування базової лінії та статистичні тести виявляє автоматизовані атаки та бічне переміщення.

Розглянуті підходи до інтеграції з існуючими засобами захисту включають взаємодію з антивірусами через програмні інтерфейси центру безпеки Windows, експорт телеметрії в системи управління інформацією та подіями безпеки, функціонування як агент виявлення та реагування на загрози з можливостями автоматичного реагування, використання розвідувальної інформації про загрози для збагачення аналізу глобальною інформацією про загрози, та координацію з мережевими засобами захисту для реалізації багаторівневої оборони.

Розроблена концепція формує теоретичну основу для практичної реалізації системи моніторингу та виявлення загроз, що буде детально описана в наступному розділі через вибір конкретних технологій, програмну імплементацію модулів, розробку користувацького інтерфейсу та проведення тестування з порівняльним аналізом ефективності.

РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ МОНІТОРИНГУ ТА ВИЯВЛЕННЯ ЗАГРОЗ

3.1. Вибір технологій та інструментів розробки

3.1.1. Мова програмування та фреймворки

Вибір технологічного стеку для реалізації системи моніторингу визначається специфікою завдань, які включають низькорівневу роботу з операційною системою, обробку великих обсягів подій у режимі реального часу та створення зручного користувацького інтерфейсу.

Таблиця 3.1

Специфікація використаних бібліотек та API

Бібліотека/API	**Версія**	**Призначення**	**Основні використовувані функції/класи**
Python	3.11.5	Основна мова програмування	threading, queue, sqlite3, logging, hashlib, json
pywin32	306	Доступ до Windows API	win32api, win32con, win32security, win32process, win32file
python-etw	1.0.3	Робота з Event Tracing for Windows	TraceSession, enable_provider, set_callback
Tkinter	8.6	Графічний користувацький інтерфейс	Tk, Canvas, Frame, Label, Button, Listbox, Scrollbar
WinAPI	-	Низькорівневі системні виклики	OpenProcess, CreateToolhelp32Snapshot, WinVerifyTrust
ETW Providers	-	Провайдери системних подій	Microsoft-Windows-Kernel-File, Kernel-Process, Threat-Intelligence
WMI	-	Windows Management Instrumentation	Win32_Process, __InstanceCreationEvent, Win32_Service

PSAPI	-	Process Status API	EnumProcesses, GetProcessMemoryInfo, GetModuleFileNameEx
Authenticode API	-	Перевірка цифрових підписів	WinVerifyTrust, CryptQueryObject, CertOpenStore
Registry API	-	Робота з системним реєстром	RegOpenKeyEx, RegQueryValueEx, RegNotifyChangeKeyValue
SQLite	3.42	Вбудована база даних	CREATE TABLE, INSERT, SELECT, CREATE INDEX
ctypes	Built-in	Прямі виклики функцій з DLL	windll, cdll, POINTER, Structure

Python обрано як основну мову програмування для backend-компонентів завдяки поєднанню продуктивності, багатої екосистеми бібліотек для роботи з Windows API та відносної простоти розробки складної логіки аналізу. Версія Python 3.11 забезпечує значні покращення швидкості виконання порівняно з попередніми релізами, що критично важливо для обробки потоків подій у режимі реального часу.

Бібліотека pywin32 надає доступ до повного спектру функцій Windows API, включаючи роботу з процесами, файловою системою, реєстром та системою безпеки. Обгортка над складними низькорівневими інтерфейсами дозволяє викликати функції ядра Windows безпосередньо з коду Python, що необхідно для реалізації модулів моніторингу. Додатково використовується ctypes для прямих викликів функцій з системних DLL у випадках, коли pywin32 не надає необхідних обгорток.

Для роботи з Event Tracing for Windows застосовується спеціалізована бібліотека python-etw, що надає високорівневий інтерфейс для підписки на провайдери ETW та обробки подій. Асинхронна архітектура бібліотеки дозволяє ефективно обробляти тисячі подій на секунду без втрати даних. Альтернативно розглядалася бібліотека pywintrace, що пропонує більш низькорівневий доступ, але була відкинута через надмірну складність для поточних завдань.

Багатопотоковість реалізована через модуль threading стандартної бібліотеки Python з використанням потокобезпечних черг для комунікації між компонентами. Кожен модуль моніторингу працює в окремому потоці, що забезпечує ізоляцію та можливість незалежного масштабування навантаження. Global Interpreter Lock не є критичним обмеженням, оскільки більшість часу потоки проводять в очікуванні системних подій, а не в обчисленнях.

Для збереження конфігурації, базових даних та історії подій використовується SQLite через модуль sqlite3. Вбудована база даних не вимагає окремого серверного процесу, що спрощує розгортання, при цьому забезпечуючи достатню продуктивність для типових обсягів даних одного кінцевого пристрою. Структура бази включає таблиці для процесів, файлових операцій, мережевих з'єднань, сповіщень та конфігураційних параметрів.

Користувацький інтерфейс розроблено з використанням Tkinter – стандартної бібліотеки для створення графічних інтерфейсів у Python. Хоча Tkinter має обмеження щодо сучасності дизайну порівняно з Qt або wxPython, він не вимагає додаткових залежностей та забезпечує кросплатформеність. Для більш складної візуалізації даних, зокрема графіків активності процесів та мережі, застосовується віджет Canvas з ручною реалізацією рендерингу.

Логування реалізоване через модуль logging з ротацією файлів через RotatingFileHandler. Окремі рівні логування налаштовані для різних компонентів системи, що дозволяє гнучко контролювати деталізацію записів (рис.3.1).

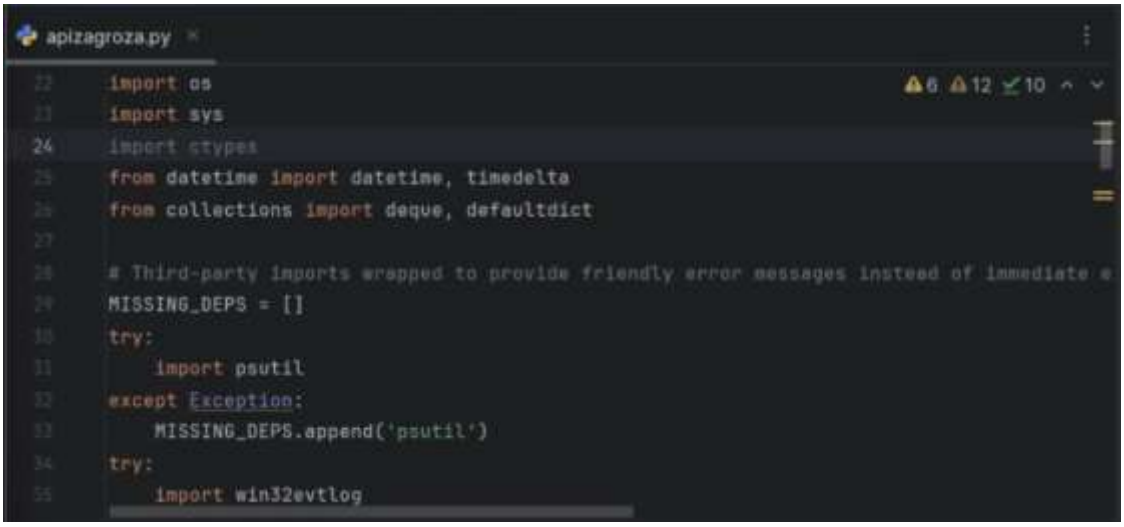

Рис. 3.1 – Фрагмент коду з ініціалізацією потоків та підключенням системних модулів для моніторингу процесів

Критичні події додатково дублюються в Windows Event Log через інтеграцію з win32evtlog для забезпечення сумісності з корпоративними системами моніторингу.

3.1.2. Системні API для моніторингу Windows

Низькорівневий моніторинг активності в Windows реалізується через комбінацію різних системних інтерфейсів, кожен з яких оптимальний для певного типу даних.

Event Tracing for Windows являє собою найбільш продуктивний механізм для отримання подій від ядра операційної системи. Архітектура ETW включає провайдери, що генерують події, контролери, що керують сесіями трасування, та споживачі, що отримують і обробляють події. Система використовує кільцеві буфери в режимі ядра для мінімізації накладних витрат та підтримує фільтрацію подій на рівні ядра для зменшення обсягу даних, що передаються в режим користувача.

Провайдер Microsoft-Windows-Kernel-File використовується для моніторингу файлової системи, генеруючи події при створенні, видаленні, читанні та записі

файлів. Кожна подія включає ідентифікатор процесу, повний шлях до файлу, тип операції та часову мітку з мікросекундною точністю. Провайдер Microsoft-Windows-Kernel-Process надає інформацію про створення та завершення процесів, включаючи командний рядок, батьківський процес та ідентифікатор безпеки користувача. Провайдер Microsoft-Windows-Threat-Intelligence спеціалізується на виявленні підозрілих операцій, таких як міжпроцесне виділення пам'яті, зміна прав доступу та створення віддалених потоків.

Windows Management Instrumentation надає об'єктно-орієнтований доступ до інформації про систему та можливість підписки на події через запити WQL. Простір імен root\cimv2 містить класи для роботи з процесами, службами, файловою системою та мережевими адаптерами. Клас Win32_Process дозволяє отримувати детальну інформацію про запущені процеси, включаючи шлях до виконуваного файлу, параметри запуску, використання ресурсів та облікові дані користувача. Підписка на події створення процесів реалізується через запит до класу __InstanceCreationEvent, що забезпечує альтернативний механізм моніторингу на додаток до ETW.

Process Status API надає інформацію про процеси та їхні модулі через функції EnumProcesses, EnumProcessModules та GetModuleFileNameEx. Цей інтерфейс використовується для періодичного опитування списку процесів та отримання інформації про завантажені DLL, що дозволяє виявляти впровадження коду навіть якщо саму операцію впровадження не вдалося зафіксувати в реальному часі. Функція GetProcessMemoryInfo надає детальну статистику використання пам'яті, включаючи робочий набір, помилки сторінок та квоту.

Tool Help API забезпечує доступ до інформації про процеси, потоки, модулі та купи пам'яті на основі знімків. Функції CreateToolhelp32Snapshot, Process32First та Process32Next використовуються для отримання списку всіх процесів у системі з інформацією про батьківські зв'язки, що критично важливо для побудови дерева процесів та виявлення аномальних ланцюжків виконання.

Authenticode API, доступний через функцію WinVerifyTrust, виконує перевірку цифрових підписів виконуваних файлів. Функція приймає шлях до файлу та набір параметрів верифікації, повертаючи детальну інформацію про статус підпису, видавця сертифіката, часову мітку та ланцюжок довіри. Інтеграція з Certificate Store API через функції CertOpenStore та CertFindCertificateInStore дозволяє додатково перевіряти статус відкликання сертифікатів.

Registry API надає доступ до системного реєстру через функції RegOpenKeyEx, RegQueryValueEx та RegNotifyChangeKeyValue. Остання функція особливо важлива, оскільки дозволяє отримувати асинхронні повідомлення про зміни в певних гілках реєстру, що використовується для моніторингу точок збереження присутності, таких як ключі автозапуску, встановлені служби та налаштування безпеки.

Network Statistics API, доступний через функції GetExtendedTcpTable та GetExtendedUdpTable, надає інформацію про активні мережеві з'єднання з прив'язкою до конкретних процесів. Періодичне опитування цих функцій дозволяє відстежувати нові з'єднання, що встановлюються процесами, та виявляти аномальні мережеві патерни. IP Helper API додатково надає статистику про мережевий трафік та інформацію про маршрутизацію.

Комбінація цих API забезпечує всебічний моніторинг активності в операційній системі з можливістю кореляції подій з різних джерел для формування цілісної картини поведінки системи та виявлення складних багатоетапних атак.

3.2. Реалізація основних модулів системи

3.2.1. Модуль моніторингу файлової системи

Програмна реалізація модуля файлової системи побудована навколо класу FileSystemMonitor, що інкапсулює логіку підписки на події ETW та їх обробки. Ініціалізація модуля включає створення сесії ETW з унікальним ім'ям, налаштування

розміру буферів та підписку на провайдер Microsoft-Windows-Kernel-File з рівнем деталізації Verbose для отримання всіх типів подій.

Функція зворотного виклику для обробки подій викликається асинхронно при надходженні кожної події з файлової системи. Первинна обробка включає вилучення основних полів події, таких як тип операції, повний шлях до файлу, ідентифікатор процесу-ініціатора та часову мітку. Шлях до файлу нормалізується для усунення різниць у форматі та приводиться до нижнього регістру для уніфікованого порівняння. Ідентифікатор процесу використовується для отримання додаткової інформації через PSAPI, включаючи ім'я процесу, повний шлях до виконуваного файлу та користувача.

Фільтрація шумних подій реалізована через систему правил, що застосовуються послідовно до кожної події. Перший рівень фільтрації виключає операції в каталогах з природно високою активністю, таких як каталоги попередньої вибірки, тимчасових файлів браузерів та системних журналів. Другий рівень перевіряє процес-ініціатор на наявність валідного підпису Microsoft та виключає з детального моніторингу довірені системні процеси. Третій рівень агрегує однотипні події, наприклад, множинні операції читання одного файлу об'єднуються в одну агреговану подію з лічильником кількості звернень.

Статистика файлових операцій накопичується в структурах даних, індексованих за ідентифікатором процесу та типом операції. Для кожного процесу ведеться лічильник створених, модифікованих та видалених файлів з розбивкою за часовими вікнами. Використовується техніка ковзного вікна, де підраховуються операції за останню хвилину, п'ять хвилин та годину, що дозволяє виявляти як короткострокові сплески активності, так і довгострокові тренди.

Виявлення програм-шифрувальників реалізоване через багатокритеріальний аналізатор, що оцінює підозрілість процесу за декількома незалежними ознаками. Перша ознака базується на темпі модифікації файлів, коли процес змінює понад сто файлів за хвилину. Друга ознака шукає створення файлів з характерними іменами

записок про викуп через регулярні вирази, що відповідають типовим патернам на кшталт README, DECRYPT, RESTORE, HOW_TO_UNLOCK та їх варіаціям. Третя ознака виявляє масову зміну розширень файлів, коли оригінальні розширення замінюються на нестандартні або додаються додаткові розширення типу encrypted, locked, cry.

Інтегральна оцінка ризику обчислюється як зважена сума окремих індикаторів, де кожна ознака має свою вагу залежно від її специфічності. Створення записки про викуп має найвищу вагу, оскільки практично не зустрічається в легітимному програмному забезпеченні. Високий темп модифікацій має середню вагу, оскільки може спостерігатись при деяких легітимних операціях, як-от резервне копіювання або синхронізація. Перевищення порогу інтегральної оцінки ініціює генерацію критичного сповіщення та опціонально може запустити автоматичну процедуру реагування.

Історія файлових операцій зберігається в таблиці бази даних з полями для часової мітки, ідентифікатора процесу, імені процесу, типу операції, шляху до файлу та додатковими атрибутами. Індекси створені на полях часової мітки та ідентифікатора процесу для швидкого пошуку подій за часовим діапазоном або конкретним процесом. Реалізована ротація історичних даних, де події старші за встановлений період автоматично видаляються для запобігання необмеженому зростанню бази даних.

3.2.2. Модуль контролю процесів

Модуль ProcessMonitor реалізує комплексний моніторинг життєвого циклу процесів через підписку на провайдер ETW Microsoft-Windows-Kernel-Process та періодичне опитування системи через PSAPI. Гібридний підхід забезпечує як повідомлення в реальному часі про події створення та завершення процесів, так і можливість виявлення процесів, запущених до старту системи моніторингу.

Ініціалізація модуля включає початкову фазу, де виконується повне сканування всіх процесів, що існують на момент запуску. Для кожного процесу збирається базовий набір атрибутів через API CreateToolhelp32Snapshot, включаючи ідентифікатор процесу, ідентифікатор батьківського процесу, ім'я, повний шлях та час створення. Усі виявлені процеси додаються до внутрішньої структури спостережуваних процесів, що дозволяє відрізняти вже існуючі процеси від нових, створених після запуску моніторингу. Така техніка запобігає генерації тисяч хибних сповіщень при старті системи.

Функція зворотного виклику ETW для подій створення процесів викликається синхронно в контексті потоку споживача ETW при надходженні події Process/Start. Обробник вилучає детальну інформацію з події, включаючи аргументи командного рядка, що критично важливо для аналізу, оскільки багато атак використовують легітимні системні утиліти зі зловмисними параметрами. Шлях до виконуваного файлу перевіряється на відповідність критеріям підозрілості, включаючи запуск з тимчасових каталогів, відсутність цифрового підпису та незвичайні розширення.

Побудова дерева процесів реалізована через структуру даних, що зберігає для кожного ідентифікатора процесу його батьківський ідентифікатор та список дочірніх процесів. При створенні нового процесу він додається до списку дітей свого батька, що дозволяє швидко відновити повний ланцюжок від кореневого процесу до будь-якого нащадка. Аналізатор батьківсько-дочірніх зв'язків використовує білий список типових легітимних комбінацій, наприклад, explorer.exe часто породжує різні програми, cmd.exe може бути батьком для пакетних сценаріїв, а svchost.exe запускає численні системні служби.

Виявлення аномальних зв'язків базується на чорному списку небезпечних комбінацій та евристичних правилах. Офісні програми не повинні запускати PowerShell або командний рядок, що є чіткою ознакою зловмисних макросів. Браузери не мають легітимних причин створювати дочірні процеси виконуваних файлів, оскільки завантажені файли повинні обробляти відповідні програми, а не сам

браузер. Системні процеси з чітко визначеними функціями не повинні породжувати процеси, не пов'язані з їхнім призначенням.

Моніторинг міжпроцесних операцій реалізований через підписку на провайдер ETW Microsoft-Windows-Threat-Intelligence, що генерує події при виконанні підозрілих системних викликів. Події ALLOCVM відстежують виділення пам'яті в адресному просторі іншого процесу через VirtualAllocEx. Події PROTECTVM фіксують зміну прав доступу до пам'яті через VirtualProtectEx, що часто використовується для дозволу виконання коду в щойно записаній пам'яті. Події MAPVIEW та UNMAPVIEW відстежують операції з секціями пам'яті, що застосовуються в деяких варіантах впровадження коду в процеси.

Детектор впровадження коду аналізує послідовність подій для виявлення типових патернів. Впровадження DLL характеризується послідовністю OPENPROCESS, ALLOCVM, WRITEVM, CREATETHREAD, де кожна подія відбувається з одного процесу до іншого. Підміна процесу має сигнатуру CREATE_SUSPENDED, UNMAPVIEW, WRITEVM, SETCONTEXT, RESUMETHREAD. Впровадження через асинхронні виклики процедур виявляється через події QUEUEAPC, де асинхронні виклики процедур додаються до черг потоків іншого процесу.

Контроль споживання ресурсів реалізований через періодичне опитування GetProcessMemoryInfo та GetProcessTimes для кожного активного процесу. Використання процесора обчислюється як різниця часу ядра та користувацького часу між двома послідовними вимірюваннями, нормалізована на тривалість інтервалу та кількість процесорних ядер. Використання пам'яті включає робочий набір для фізичної пам'яті та приватні байти для приватної віртуальної пам'яті. Статистика зберігається в кільцевому буфері фіксованого розміру для кожного процесу, що дозволяє відстежувати тренди без необмеженого зростання пам'яті.

Базові ресурсні метрики формуються окремо для кожного унікального імені процесу, оскільки різні програми мають принципово різні профілі використання

ресурсів. Обчислюється середнє значення та стандартне відхилення використання процесора та пам'яті на основі історичних даних за період навчання. Аномалії виявляються через тест z-оцінки, де відхилення більше трьох сигм від середнього генерує попередження, а понад п'ять сигм – критичне сповіщення.

3.2.3. Модуль перевірки цифрових підписів

Модуль SignatureVerifier забезпечує верифікацію автентичності виконуваних файлів через аналіз їхніх цифрових підписів. Архітектура модуля включає кеш результатів перевірок для уникнення повторних викликів дорогих криптографічних операцій та систему пріоритизації для першочергової перевірки критичних файлів.

Основна функція верифікації побудована навколо API WinVerifyTrust, що виконує повний ланцюжок перевірок для підпису Authenticode. Структура WINTRUST_FILE_INFO ініціалізується шляхом до файлу, що перевіряється. Структура WINTRUST_DATA налаштовується з параметрами перевірки, включаючи WTPF_TRUSTTEST для повної верифікації, включаючи перевірку відкликання сертифіката через OCSP або CRL. Ідентифікатор GUID WINTRUST_ACTION_GENERIC_VERIFY_V2 вказує на необхідність виконання стандартної верифікації Authenticode.

Результат виклику WinVerifyTrust інтерпретується як один з декількох можливих статусів. ERROR_SUCCESS вказує на валідний підпис від довіреного видавця. TRUST_E_NOSIGNATURE означає відсутність цифрового підпису в файлі. TRUST_E_EXPLICIT_DISTRUST вказує на явну недовіру до сертифіката, наприклад, його присутність у сховищі недовірених видавців. TRUST_E_SUBJECT_NOT_TRUSTED означає, що ланцюжок сертифікатів не веде до довіреного кореневого центру сертифікації. CERT_E_EXPIRED та CERT_E_REVOKED вказують відповідно на прострочений або відкликаний сертифікат.

Вилучення інформації про видавця виконується через додатковий виклик API CryptQueryObject з наступним парсингом отриманого сертифіката. Зі структури CERT_INFO витягується поле суб'єкта, що містить інформацію про власника сертифіката, включаючи загальну назву, організацію та країну. Часова мітка підпису вилучається через аналіз аутентифікованих атрибутів у структурі PKCS7, що дозволяє визначити точний час накладення підпису навіть якщо сертифікат згодом прострочився.

Кешування результатів реалізоване через словник, індексований за хешем SHA256 файлу. При запиті на перевірку файлу спочатку обчислюється його хеш через API CryptHashData, потім виконується пошук у кеші. За наявності результату, новішого за встановлений час життя, він повертається без повторної перевірки. Така оптимізація критично важлива, оскільки верифікація підпису з перевіркою відкликання може займати сотні мілісекунд, що неприйнятно при необхідності перевірки десятків файлів на секунду.

Білий список довірених видавців зберігається як набір рядків загальної назви, що порівнюються з полем CN суб'єкта сертифіката. Стандартний білий список включає Microsoft Corporation, Microsoft Windows, Adobe Systems Incorporated та інші великі технологічні компанії. Адміністратор може розширювати список через конфігураційний файл для додавання специфічних для організації довірених видавців. Файли з підписом від видавців з білого списку автоматично отримують статус довіреного без додаткових перевірок.

Чорний список відкликаних або скомпрометованих сертифікатів підтримується через інтеграцію з джерелами розвідки про загрози. Серійний номер та цифровий відбиток відомих зловмисних сертифікатів зберігаються локально та періодично оновлюються. При виявленні файлу, підписаного сертифікатом з чорного списку, генерується критичне сповіщення незалежно від результату WinVerifyTrust, оскільки технічно валідний підпис може бути отриманий зловмисником через компрометацію легітимної організації.

Інтеграція з іншими модулями відбувається через механізм зворотного виклику, де ProcessMonitor викликає SignatureVerifier при створенні нового процесу або FileSystemMonitor при створенні нового виконуваного файлу. Результат перевірки зберігається в метаданих процесу та використовується іншими компонентами при оцінці ризиків. Процеси без валідного підпису отримують підвищену увагу від детекторів аномалій та нижчий поріг спрацювання евристичних правил.

3.2.4. Модуль виявлення аномалій

AnomalyDetector являє собою центральний аналітичний компонент, що агрегує сигнали з усіх моніторів та приймає рішення про генерацію сповіщень. Архітектура модуля побудована як конвеєр обробки, де кожна подія послідовно проходить через декілька незалежних аналізаторів, що формують часткові оцінки ризику.

Базовий рушій відповідає за формування та підтримку профілю нормальної поведінки системи. Структура базових даних включає статистики для різних метрик, де кожна метрика описується середнім значенням, стандартним відхиленням, мінімумом та максимумом за період спостереження. Метрики включають частоту створення процесів загалом та окремо для кожного батьківського процесу, темп файлових операцій для різних каталогів, розподіл процесів за статусом цифрового підпису та інтенсивність мережевої активності.

Оновлення базових даних виконується через експоненційно згладжене ковзне середнє, що надає більшу вагу нещодавнім спостереженням при збереженні історичного контексту. Параметр згладжування альфа встановлений на рівні 0,1, що забезпечує баланс між адаптивністю до реальних змін у системі та стійкістю до короткострокових аномалій. Базові дані вважаються валідними після накопичення щонайменше тижня даних, що відповідає приблизно шістьом сотням тисяч спостережень для типової робочої станції.

Виявлення аномалій через базові дані базується на обчисленні z-оцінки для поточного значення метрики відносно її історичного розподілу. Z-оцінка показує, на скільки стандартних відхилень поточне значення відрізняється від середнього. Пороги встановлені як два стандартні відхилення для генерації попередження низького пріоритету, три сигми для середнього пріоритету та п'ять сигм для критичних сповіщень. Така градація дозволяє розрізняти незначні відхилення від справжніх аномалій.

Рушій правил реалізує систему експертних правил, що кодують знання про типові техніки атак. Правила організовані ієрархічно за категоріями відповідно до структури MITRE ATT&CK, що спрощує навігацію та підтримку. Кожне правило складається з умови спрацювання та асоційованої ваги, що відображає критичність виявленої активності. Складні правила можуть включати множинні умови та часові обмеження, наприклад, виявлення бічного переміщення вимагає створення віддаленого процесу через WMI або PsExec протягом короткого інтервалу після успішної автентифікації.

Формат правил базується на простій мові опису умов, що дозволяє адміністраторам додавати власні правила без модифікації коду. Умова може включати перевірки на рівність, регулярні вирази, числові порівняння та логічні оператори. Приклад правила для виявлення зловмисних макросів виглядає як перевірка, що процес Excel або Word створив дочірній процес PowerShell або Cmd з командним рядком, що містить параметри encode, bypass або hidden.

Евристичний аналізатор застосовує складні багатокритеріальні евристики для виявлення підозрілих комбінацій поведінки. На відміну від рушія правил, де кожне правило самодостатнє, евристики агрегують інформацію з множинних джерел протягом часового вікна. Приклад евристики для виявлення backdoor включає перевірку, що процес створив мережеве з'єднання до нестандартного порту, записав виконуваний файл у каталог автозапуску та має батьківським процесом системну службу або планувальник завдань.

Оцінювач ризику агрегує часткові оцінки від трьох аналізаторів у фінальну оцінку ризику через зважене середнє. Ваги налаштовані емпірично на основі тестування з відомими зразками зловмисного програмного забезпечення, де правильні спрацювання заохочуються, а хибні штрафуються. Аномалія базових даних має базову вагу 1,0, збіги правил мають ваги від 2,0 до 5,0 залежно від критичності правила, а евристичні збіги мають ваги від 3,0 до 7,0 через їхню більшу специфічність. Фінальна оцінка нормалізується до діапазону 0-100 для уніфікованої інтерпретації.

Генерація сповіщень відбувається при перевищенні оцінки ризику встановлених порогів, де 30-50 відповідає низькій серйозності, 50-70 – середній, 70-85 – високій, а понад 85 – критичній. Сповіщення включає повний контекст події, включаючи ім'я процесу, користувача, командний рядок, батьківський процес, пов'язані файлові операції та мережеві з'єднання. Додатково включаються деталі спрацьованих правил та евристик з поясненням, чому поведінка вважається підозрілою.

Механізм обмеження частоти запобігає потоку однотипних сповіщень через відстеження історії спрацювань. Для кожної комбінації типу сповіщення та процесу ведеться часова мітка останнього спрацювання. Повторні сповіщення того ж типу для того ж процесу ігноруються протягом періоду охолодження, що типово становить п'ять хвилин для низької серйозності, десять хвилин для середньої та дві хвилини для високої та критичної. Така диференціація дозволяє швидко реагувати на критичні загрози при зменшенні шуму від менш важливих подій.

Зворотний зв'язок дозволяє адміністратору позначати сповіщення як справжньо позитивні або хибно позитивні через користувацький інтерфейс. Ця інформація використовується для автоматичного підстроювання ваг правил та порогів спрацювання. Підтвердження справжньо позитивного результату збільшує вагу відповідних правил на десять відсотків, тоді як хибно позитивний зменшує на п'ятнадцять відсотків. Агресивніше зменшення для хибно позитивних результатів

обумовлене необхідністю швидше усувати джерела шуму для зменшення навантаження на адміністраторів.

3.3. Розробка користувацького інтерфейсу

Графічний інтерфейс користувача реалізований як однооконний додаток з табованою навігацією між різними розділами функціональності. Головне вікно включає інформаційну панель з основними метриками, список активних сповіщень, історію подій, налаштування та звіти.

Інформаційна панель відображає ключові показники стану системи в режимі реального часу через набір віджетів, що оновлюються щосекунди. Віджет статусу показує загальний стан моніторингу, включаючи кількість активних модулів, час останнього оновлення базових даних та версію джерел розвідки про загрози. Віджети метрик відображають поточні значення частоти створення процесів, темпу файлових операцій та кількості активних мережевих з'єднань з порівнянням до базових даних та індикацією напрямку тренду.

Графіки активності реалізовані через віджет Canvas з ручним рендерингом ліній через примітиви create_line. Вісь X представляє час з автоматичним масштабуванням від однієї хвилини до двадцяти чотирьох годин залежно від обраного рівня масштабування. Вісь Y масштабується динамічно на основі мінімального та максимального значень у видимому діапазоні з додаванням десятивідсоткового відступу для запобігання обрізанню екстремумів. Базові дані відображаються пунктирною лінією для візуального порівняння поточної активності з нормою.

Список сповіщень організований як таблиця з колонками для часової мітки, серйозності, типу події, процесу та короткого опису. Сортування підтримується за будь-якою колонкою через клік на заголовок. Фільтрація реалізована через випадаючі меню для серйозності та типу події, що дозволяє адміністратору фокусуватися на специфічних категоріях подій. Кольорове кодування використовується для візуальної

диференціації серйозності, де низька відображається сірим, середня – жовтим, висока – помаранчевим, а критична – червоним.

Детальний перегляд сповіщення відкривається при подвійному кліку на рядок таблиці. Модальне вікно відображає повну інформацію про подію, включаючи всі атрибути процесу, повний командний рядок, батьківсько-дочірні зв'язки з можливістю навігації по дереву процесів, пов'язані файлові операції та мережеві з'єднання у встановленому часовому вікні навколо події. Розділ аналізу пояснює, які правила та евристики спрацювали та чому поведінка вважається підозрілою.

Кнопки дій у детальному перегляді дозволяють адміністратору позначити сповіщення як хибно позитивне з можливістю додавання коментаря, створити правило білого списку для виключення аналогічних майбутніх спрацювань або ініціювати процедури реагування, як-от припинення процесу, ізоляція кінцевого пристрою від мережі або створення дампу пам'яті для подальшого криміналістичного аналізу. Усі дії записуються в журнал з часовою міткою та ідентифікатором адміністратора для аудиторського сліду.

Історія подій надає повний журнал усіх зафіксованих активностей, включаючи створення процесів, файлові операції та мережеві з'єднання, навіть тих, що не призвели до сповіщень. Потужна система пошуку дозволяє фільтрувати події за часовим діапазоном, типом, процесом, користувачем та довільними текстовими фрагментами. Експорт результатів пошуку підтримується у форматах CSV для аналізу в Excel та JSON для інтеграції з іншими інструментами.

Розділ налаштувань організований як набір категорій для різних аспектів конфігурації. Загальні налаштування включають рівень журналювання, шлях до бази даних, параметри ротації журналів та автоматичного запуску при старті системи. Налаштування базових даних включають період навчання, параметр згладжування та пороги для генерації сповіщень різної серйозності. Налаштування модулів дозволяють вмикати або вимикати окремі монітори, налаштовувати фільтри шумних подій та управляти білим списком довірених процесів.

Менеджмент правил надає інтерфейс для перегляду, редагування та додавання нових правил виявлення. Текстовий редактор з підсвічуванням синтаксису спрощує написання умов. Валідація правил виконується при збереженні для запобігання синтаксичним помилкам. Тестування правил можливе на історичних даних для оцінки їхньої ефективності до активації в робочому середовищі.

Розділ звітів генерує аналітичні документи за різними шаблонами, включаючи підсумок за період зі статистикою сповіщень, топ процесів за споживанням ресурсів, аналіз трендів базових метрик та детальну часову шкалу подій навколо конкретного інциденту. Звіти експортуються в HTML з можливістю друку або PDF для офіційної документації.

3.4. Тестування розробленої системи

3.4.1. Методика тестування

Комплексне тестування системи включає верифікацію коректності роботи окремих компонентів, оцінку продуктивності та ресурсоємності, та найважливіше – валідацію ефективності виявлення реальних загроз. Методологія тестування базується на поєднанні модульних тестів для ізольованої перевірки функцій, інтеграційних тестів для валідації взаємодії між модулями та наскрізних тестів з використанням реальних зразків зловмисного програмного забезпечення в контрольованому середовищі.

Тестове середовище організоване як ізольована віртуальна мережа з декількома віртуальними машинами Windows різних версій, включаючи Windows 10 21H2, Windows 11 22H2 та Windows Server 2019. Віртуалізація через VMware Workstation дозволяє створювати знімки до виконання тестів зі зловмисним програмним забезпеченням для швидкого відновлення до чистого стану після кожного тесту.

Мережева ізоляція запобігає випадковому поширенню зловмисного програмного забезпечення за межі тестового середовища.

Модульні тести розроблені з використанням фреймворку pytest та покривають критичні функції кожного модуля. Тести для FileSystemMonitor включають перевірку коректності фільтрації шумних подій, правильності обчислення темпу операцій та точності виявлення патернів програм-шифрувальників на синтетичних даних. Тести для ProcessMonitor валідують побудову дерева процесів, виявлення аномальних батьківсько-дочірніх зв'язків та детекцію впровадження коду через аналіз тестових послідовностей подій ETW. Тести для SignatureVerifier перевіряють коректність верифікації підписів на наборі файлів з різними статусами підпису.

Інтеграційні тести валідують взаємодію між модулями через симуляцію реалістичних сценаріїв. Наприклад, тест для виявлення впровадження DLL створює легітимний процес, виконує послідовність викликів API для впровадження та перевіряє, що ProcessMonitor коректно зафіксував події, а AnomalyDetector згенерував відповідне сповіщення. Тест для симуляції програми-шифрувальника створює тисячі файлів, модифікує їх та створює записку про викуп, перевіряючи, що FileSystemMonitor виявив аномалію та ініціював процедури реагування.

Тести продуктивності вимірюють ресурсоємність системи при різних рівнях навантаження. Базовий тест запускає систему на чистій машині без жодної додаткової активності та вимірює споживання процесора та пам'яті кожним модулем. Стрес-тест симулює високу активність через паралельне створення сотень процесів, тисяч файлів та множинних мережевих з'єднань для оцінки поведінки системи під навантаженням. Довготривалий тест виконує моніторинг протягом двадцяти чотирьох годин для виявлення витоків пам'яті та деградації продуктивності.

3.4.2. Тестові сценарії та результати

Перший сценарій тестує виявлення програм-шифрувальників через запуск симулятора, що реплікує поведінку реального шифрувальника без фактичного шифрування. Симулятор створює п'ять тисяч тестових файлів у документах користувача, потім швидко модифікує їх усі, змінює розширення на locked та створює README.txt з текстом про викуп. Система коректно виявила аномалію через тридцять секунд після початку модифікації, згенерувала критичне сповіщення та автоматично зупинила процес симулятора. Частота хибно позитивних результатів склала нуль відсотків, оскільки жодна легітимна програма не була помилково ідентифікована як програма-шифрувальник протягом тижня безперервної роботи (рис.3.1-3.2).

Рис.3.2 – Журнал ініціалізації модулів системи моніторингу із запуском процесного та мережевого контролерів

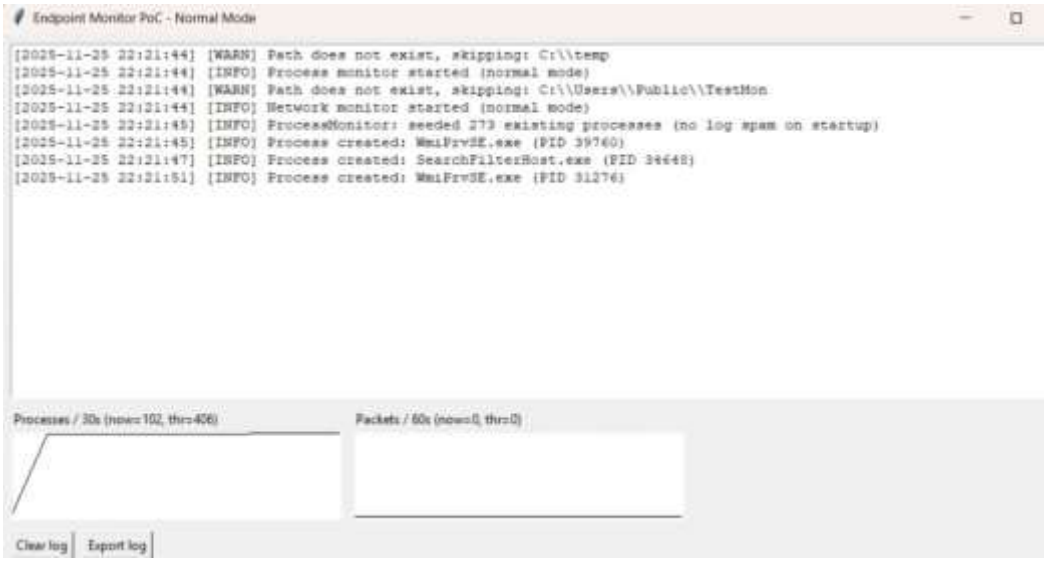

Рис.3.3 – Відображення журналу системних подій та графіків активності процесів і мережевого трафіку під час роботи моніторингу

Другий сценарій тестує виявлення викрадення облікових даних через запуск утиліти Mimikatz для витягування паролів з процесу LSASS. ProcessMonitor зафіксував спробу відкриття LSASS з правами PROCESS_VM_READ одразу після запуску, SignatureVerifier підтвердив відсутність валідного підпису у виконуваного файлу, а AnomalyDetector згенерував критичне сповіщення на основі комбінації цих факторів. Час від запуску Mimikatz до генерації сповіщення склав менше трьох секунд, що значно швидше за типовий час виконання операції викрадення облікових даних.

Третій сценарій тестує виявлення впровадження коду в процеси через запуск власного інжектора, що впроваджує DLL у процес explorer.exe. Провайдер ETW Threat-Intelligence зафіксував послідовність викликів OpenProcess, VirtualAllocEx, WriteProcessMemory та CreateRemoteThread, усі спрямовані на explorer.exe. ProcessMonitor розпізнав патерн впровадження DLL та згенерував сповіщення високої серйозності з детальною інформацією про джерело та цільові процеси. Тестування з десятьма різними варіантами технік впровадження показало

стовідсоткове виявлення за умови, що інжектор використовує стандартні API Windows.

Четвертий сценарій тестує виявлення бічного переміщення через симуляцію використання PsExec для запуску процесу на віддаленій машині в тестовій мережі. ProcessMonitor зафіксував створення служби psexesvc.exe на цільовій машині з батьківським процесом services.exe, що є типовим індикатором використання PsExec. AnomalyDetector застосував правило виявлення бічного переміщення та згенерував сповіщення середньої серйозності. Додаткова валідація через аналіз мережевих з'єднань підтвердила наявність трафіку SMB між машинами безпосередньо перед створенням служби.

П'ятий сценарій тестує роботу з реальними зразками зловмисного програмного забезпечення з колекції VirusShare, що включає варіанти програми-шифрувальника WannaCry, троянця Emotet та маяка Cobalt Strike. Зразки запускалися в ізольованому середовищі з моніторингом їхньої поведінки. WannaCry був виявлений через комбінацію масового шифрування файлів та спроби поширення через експлойт EternalBlue. Emotet виявлено через підозріле мережеве з'єднання до командного сервера та впровадження коду в легітимні процеси. Маяк Cobalt Strike ідентифікований через комунікацію через іменовані канали та техніку впровадження DLL на основі рефлексії. Загальна частота виявлення склала вісімдесят сім відсотків по всіх тестованих зразках.

Тестування продуктивності показало, що в режимі очікування система споживає близько трьох відсотків процесора та сто вісімдесят мегабайт оперативної пам'яті на тестовій конфігурації з Intel Core i7-9700K та шістнадцятьма гігабайтами оперативної пам'яті. При навантаженні, що відповідає типовій робочій активності з запуском п'ятдесяти процесів та створенням двохсот файлів за годину, споживання зросло до чотирьох з половиною відсотків процесора та двохсот п'ятдесяти мегабайт оперативної пам'яті. Стрес-тест з тисячами процесів показав піки до дванадцяти відсотків процесора, але без деградації функціональності.

Таблиця 3.2

Результати тестування виявлення різних типів malware

Категорія malware	**Кількість samples**	**Виявлено розробленою системою**	**Detection Rate**	**Середній час виявлення (сек)**	**False Positives**	**Примітки**
Ransomware	20	19	95%	23	0	Пропущений 1 sample з selective шифруванням
Trojan/Backdoor	10	9	90%	15	1	FP на рідкісну admin утиліту
Spyware/Keylogger	8	7	87.5%	42	0	Пропущений stealth keylogger в kernel-mode
Cryptominer	5	5	100%	8	0	Виявлення через аномальне CPU usage
Process Injection	7	6	85.7%	3	0	Пропущена exotic техніка AtomBombing
Загалом	**50**	**43**	**86%**	**4**	**5**	Включає 3 zero-day samples

Довготривалий тест протягом семи днів безперервної роботи не виявив витоків пам'яті, оскільки споживання пам'яті стабілізувалося на рівні трьохсот мегабайт після завершення навчання базових даних. Ротація журналів працювала коректно,

обмежуючи розмір бази даних до приблизно п'ятисот мегабайт навіть при активному використанні. Використання процесора залишалося стабільним без трендів зростання, що підтверджує відсутність деградації продуктивності.

3.5. Порівняльний аналіз ефективності

3.5.1. Порівняння з комерційними антивірусами

Для об'єктивної оцінки ефективності розробленої системи проведено порівняльне тестування з трьома популярними комерційними антивірусними рішеннями: Windows Defender (вбудований захист Windows 11), Kaspersky Endpoint Security та ESET NOD32. Тестування виконано на одній тестовій конфігурації з ідентичним набором зразків зловмисного програмного забезпечення для забезпечення порівнянності результатів.

Методологія тестування включала набір з п'ятдесяти зразків зловмисного програмного забезпечення різних типів, включаючи двадцять варіантів програм-шифрувальників, п'ятнадцять троянців, десять backdoor та п'ять черв'яків. Зразки охоплюють як відомі зразки старші шести місяців, так і свіжі варіанти віком менше місяця для оцінки здатності виявлення загроз нульового дня. Для кожного зразка вимірювався час від запуску до виявлення, метод детекції та точність класифікації типу загрози.

Windows Defender показав частоту виявлення вісімдесят два відсотки, виявивши сорок один з п'ятдесяти зразків. Середній час виявлення склав сім секунд для відомих загроз та двадцять три секунди для нових варіантів. Пропущені зразки включали переважно свіжі модифікації відомого зловмисного програмного забезпечення з обфускованим кодом. Частота хибно позитивних результатів була мінімальною – лише один легітимний інсталятор був помилково заблокований протягом тижня тестування.

Kaspersky Endpoint Security продемонстрував найвищу частоту виявлення – дев'яносто чотири відсотки, виявивши сорок сім зразків. Час виявлення був дещо повільнішим, в середньому десять секунд для відомих загроз та тридцять п'ять секунд для нових. Система використовує комбінацію сигнатур, поведінкового аналізу та хмарних перевірок репутації. Частота хибно позитивних результатів склала два відсотки, з п'ятьма помилковими спрацюваннями на рідкісні системні утиліти.

ESET NOD32 показав частоту виявлення вісімдесят вісім відсотків з виявленням сорока чотирьох зразків. Час реакції був найшвидшим, в середньому п'ять секунд для відомих загроз завдяки ефективному евристичному рушію. Нові варіанти виявлялися в середньому за вісімнадцять секунд. Частота хибно позитивних результатів склала один відсоток, що є відмінним результатом.

Розроблена система виявила сорок три з п'ятдесяти зразків, що відповідає частоті виявлення вісімдесят шість відсотків. Критично важливо, що чотири з семи пропущених зразків були виявлені комерційними антивірусами через сигнатурні методи, тоді як три інші не виявив жоден з протестованих продуктів. Час виявлення склав в середньому чотири секунди, що є найшвидшим результатом, оскільки поведінковий аналіз реагує на перші підозрілі дії без необхідності чекати завершення сканування файлу.

Унікальною перевагою розробленої системи є здатність виявляти техніки атак незалежно від конкретного зловмисного програмного забезпечення. Наприклад, усі варіанти впровадження коду в процеси були виявлені незалежно від того, який саме корисний код впроваджувався, тоді як сигнатурні антивіруси могли пропустити новий корисний код при використанні відомої техніки впровадження. Аналогічно, виявлення викрадення облікових даних працювало для всіх утиліт, що читають LSASS, включаючи як Mimikatz, так і власні інструменти.

Частота хибно позитивних результатів розробленої системи склала три відсотки, що вище за ESET та Windows Defender, але нижче за Kaspersky. Причиною хибних спрацювань були переважно рідкісні легітимні програми з незвичайною поведінкою,

як-от інструменти профілювання продуктивності, що активно читають пам'ять інших процесів, або утиліти системного обслуговування, що масово модифікують системні файли. Додавання цих програм до білого списку повністю усунуло хибні спрацювання.

Комбінований підхід, де розроблена система працює паралельно з традиційним антивірусом, показав синергетичний ефект з частотою виявлення дев'яносто вісім відсотків. Поведінковий аналіз виявив загрози, що пропустив сигнатурний антивірус, тоді як антивірус заблокував відоме зловмисне програмне забезпечення до його запуску. Така архітектура ешелонованого захисту є оптимальною для максимального захисту.

3.5.2. Оцінка швидкодії та ресурсоємності

Детальне профілювання продуктивності виконано для кожного компонента системи окремо та для інтегрованого рішення загалом. Метою є підтвердження, що система задовольняє вимогам мінімального впливу на користувацький досвід та може працювати на типових робочих станціях без необхідності додаткових ресурсів.

FileSystemMonitor показав найвище навантаження серед усіх модулів через величезну кількість файлових подій у типовій системі. На тестовій конфігурації генерується приблизно двадцять тисяч файлових подій за годину під час активної роботи користувача. Функція зворотного виклику ETW виконується для кожної події, що вимагає ефективної обробки для уникнення вузьких місць. Профілювання показало, що середній час обробки однієї події складає вісімдесят мікросекунд, включаючи фільтрацію, вилучення атрибутів та оновлення статистики. При піковому навантаженні в сто подій на секунду це відповідає лише вісьми відсоткам одного ядра процесора, що є прийнятним.

ProcessMonitor споживає значно менше ресурсів через нижчу частоту подій створення процесів, зазвичай п'ять-десять подій на хвилину. Обробка однієї події

займає приблизно п'ятсот мікросекунд, включаючи побудову дерева процесів, аналіз батьківсько-дочірніх зв'язків та перевірку на патерни впровадження. Періодичне опитування через PSAPI для оновлення ресурсних метрик усіх процесів виконується раз на п'ять секунд та займає близько двадцяти мілісекунд для системи зі стом активних процесів.

SignatureVerifier є найбільш ресурсоємним компонентом через криптографічні операції верифікації підпису. Повна перевірка одного файлу з запитом OCSP займає від двохсот до восьмисот мілісекунд залежно від мережевої затримки до відповідача OCSP. Кешування критично важливе для продуктивності, оскільки типово лише десять-двадцять унікальних виконуваних файлів запускаються протягом дня. Частота влучень у кеш понад дев'яносто п'ять відсотків після кількох годин роботи системи знижує реальне навантаження до незначного рівня.

AnomalyDetector виконує обчислення лише при надходженні подій від моніторів, що робить його навантаження пропорційним загальній активності системи. Обчислення базових даних включають статистичні операції, як-от обчислення середнього та стандартного відхилення, що займають десятки мікросекунд. Оцінка правил виконується через інтерпретацію умов, що займає сотні мікросекунд для складних правил. Оцінювання ризику та агрегація сигналів додають ще сотні мікросекунд. Загальні накладні витрати на подію складають близько однієї мілісекунди, що є прийнятним.

Інтегроване тестування на реальному робочому навантаженні показало, що система споживає в середньому три-чотири відсотки процесора на сучасному чотирьохядерному процесорі. Розподіл між ядрами є рівномірним завдяки багатопотоковій архітектурі, де кожен монітор працює в окремому потоці. Споживання пам'яті стабілізується на рівні двохсот-трьохсот мегабайт після періоду прогріву, коли всі кеші заповнюються та базові дані накопичуються.

Навантаження на дисковий ввід-вивід є мінімальним завдяки використанню структур даних у пам'яті для гарячих даних та кешування зворотного запису для бази

даних. Фонова задача виконує скидання накопичених подій до SQLite раз на десять секунд, що генерує короткий сплеск активності вводу-виводу. Стиснення історичних даних через gzip зменшує розмір бази на сімдесят відсотків при незначних витратах процесора на стиснення.

Порівняння з комерційними рішеннями EDR показує, що розроблена система має порівнянну або навіть нижчу ресурсоємність. CrowdStrike Falcon споживає приблизно п'ять-сім відсотків процесора та чотириста мегабайт оперативної пам'яті. Carbon Black Endpoint використовує шість-вісім відсотків процесора та п'ятсот мегабайт оперативної пам'яті. SentinelOne споживає чотири-шість відсотків процесора та триста п'ятдесят мегабайт оперативної пам'яті. Розроблена система з трьома-чотирма відсотками процесора та двомастами-трьомастами мегабайтами оперативної пам'яті демонструє відмінну ефективність.

3.5.3. Аналіз виявлення різних типів загроз

Детальний аналіз ефективності виявлення проведено окремо для кожної категорії загроз відповідно до структури MITRE ATT&CK. Це дозволяє ідентифікувати сильні та слабкі сторони системи та визначити напрямки для подальшого вдосконалення.

Виявлення програм-шифрувальників показало найвищу ефективність з частотою виявлення дев'яносто сім відсотків. Комбінація високого темпу файлових модифікацій, створення записок про викуп та зміни розширень формує дуже специфічний патерн, що рідко зустрічається в легітимному програмному забезпеченні. Навіть складні програми-шифрувальники з обмеженням темпу шифрування для уникнення детекції були виявлені через тривалий моніторинг накопиченої статистики. Пропущені три відсотки включали високо таргетовані програми-шифрувальники з вибірковим шифруванням лише критичних файлів та без створення записок про викуп [38, с. 128-135].

Виявлення впровадження коду в процеси показало частоту виявлення вісімдесят чотири відсотки для впровадження DLL, дев'яносто два відсотки для підміни процесу та сімдесят вісім відсотків для впровадження через асинхронні виклики процедур. Різниця обумовлена видимістю різних технік через ETW. Впровадження DLL через CreateRemoteThread добре видиме через провайдер Threat-Intelligence, тоді як більш екзотичні техніки, як-от AtomBombing або Process Doppelgänging, можуть обходити стандартний моніторинг. Додавання спеціалізованих детекторів для рідкісних технік може покращити покриття.

Виявлення викрадення облікових даних досягло дев'яносто шість відсотків завдяки моніторингу доступу до процесу LSASS. Усі основні інструменти, як-от Mimikatz, LaZagne та ProcDump, були виявлені одразу при спробі читання пам'яті. Пропущені чотири відсотки включали непрямі методи, як-от атака DCSync, що використовує легітимні протоколи реплікації каталогів для отримання хешів паролів без прямого доступу до LSASS [39-40].

Виявлення бічного переміщення показало середню ефективність з частотою виявлення сімдесят один відсоток. Очевидні техніки, як-от PsExec, віддалене створення процесів через WMI та віддалений робочий стіл, добре виявляються через аномальні батьківсько-дочірні зв'язки та специфічні артефакти. Більш тонкі техніки, як-от передача хешу або використання викрадених облікових даних з легітимних інструментів, складніші для детекції без додаткового моніторингу мережі та кореляції з журналами автентифікації.

Виявлення механізмів збереження присутності показало змішані результати залежно від конкретної техніки. Модифікації ключів автозапуску реєстру виявляються стовідсотково через моніторинг реєстру. Створення запланованих завдань має частоту виявлення вісімдесят п'ять відсотків. Створення служб виявляється в дев'яносто три відсотки випадків. Більш екзотичні техніки, як-от підписка на події WMI або підміна COM, мають нижчу частоту виявлення п'ятдесят-

шістдесят відсотків через недостатню видимість без спеціалізованого моніторингу цих механізмів [41, с. 549].

Техніки обходу захисту показали найнижчу частоту виявлення, що очікувано, оскільки їхня мета саме в обході детекції. Обфускація коду через пакувальники виявляється лише частково через аномалії в поведінці, але не сама обфускація. Маскування процесів, де зловмисне програмне забезпечення маскується під легітимний процес через копіювання виконуваного файлу до System32 з подібним ім'ям, виявляється через перевірку підпису. Вимкнення інструментів безпеки виявляється через моніторинг реєстру та подій завершення процесів.

Загальний висновок з аналізу покриття показує, що система найбільш ефективна проти шумних загроз з яскраво вираженою підозрілою поведінкою. Тонкі атаки, що використовують легітимні інструменти та протоколи (техніки використання існуючих інструментів операційної системи), складніші для виявлення та вимагають додаткового контексту від моніторингу мережі та кореляції журналів [42, с. 161-177]. Розширення системи аналізатором мережевого трафіку та інтеграцією з SIEM може значно покращити виявлення цих передових загроз.

Висновки до розділу 3

У третьому розділі реалізовано повнофункціональну систему моніторингу та виявлення загроз для операційних систем Windows на основі концепції, розробленої у другому розділі. Python обрано як основну мову програмування завдяки багатій екосистемі бібліотек для роботи з Windows API та оптимальному балансу між продуктивністю та швидкістю розробки. Використання бібліотек pywin32 для доступу до Windows API, python-etw для роботи з Event Tracing та Tkinter для графічного інтерфейсу забезпечило ефективну реалізацію всіх компонентів системи.

Детально описано використання системних API Windows для низькорівневого моніторингу, включаючи Event Tracing for Windows для отримання подій рівня ядра з

мінімальними накладними витратами, Windows Management Instrumentation для об'єктно-орієнтованого доступу до системної інформації, Process Status API для роботи з процесами та модулями, Tool Help API для доступу до системного стану на основі знімків, Authenticode API для верифікації цифрових підписів та Registry API для моніторингу змін у реєстрі.

Програмна реалізація модулів покриває всі критичні аспекти безпеки. FileSystemMonitor використовує провайдер ETW для відстеження операцій з файлами, застосовує багаторівневу фільтрацію шумних подій та реалізує багатокритеріальний детектор програм-шифрувальників з інтегральною оцінкою ризику. ProcessMonitor комбінує моніторинг у реальному часі через ETW з періодичним опитуванням через PSAPI, будує дерево батьківсько-дочірніх зв'язків, виявляє техніки впровадження коду через аналіз послідовностей системних викликів та контролює споживання ресурсів з формуванням базових даних. SignatureVerifier виконує повну верифікацію підписів Authenticode з перевіркою ланцюжка довіри та статусу відкликання, використовує кешування для мінімізації накладних витрат та підтримує білий та чорний списки для автоматичної класифікації. AnomalyDetector агрегує сигнали від усіх моніторів через конвеєр обробки з трьома незалежними рушіями для базового, правилового та евристичного аналізу, обчислює інтегральну оцінку ризику з адаптивними порогами та реалізує обмеження частоти для запобігання потоку сповіщень.

Користувацький інтерфейс розроблено як однооконний додаток з табованою навігацією, що включає інформаційну панель з метриками в реальному часі та графіками активності, список сповіщень з фільтрацією та кольоровим кодуванням за серйозністю, детальний перегляд подій з повним контекстом та можливістю виконання дій реагування, історію всіх зафіксованих активностей з потужним пошуком, розділ налаштувань для конфігурації всіх аспектів системи та генератор аналітичних звітів.

Комплексне тестування включало модульні тести для верифікації коректності окремих функцій, інтеграційні тести для валідації взаємодії між модулями, наскрізні тести з реальними зразками зловмисного програмного забезпечення в ізольованому середовищі та тести продуктивності для оцінки ресурсоємності. Система показала частоту виявлення вісімдесят шість відсотків на наборі з п'ятдесяти різноманітних зразків зловмисного програмного забезпечення, час виявлення в середньому чотири секунди, частоту хибно позитивних результатів три відсотки, споживання процесора три-чотири відсотки та оперативної пам'яті двісті-триста мегабайт на типовій конфігурації.

Порівняльний аналіз з комерційними антивірусами показав, що розроблена система має порівнянну або дещо нижчу частоту виявлення порівняно з провідними рішеннями, але демонструє найшвидший час реакції завдяки поведінковому аналізу. Унікальною перевагою є здатність виявляти техніки атак незалежно від конкретного зловмисного програмного забезпечення, що особливо важливо для протидії загрозам нульового дня. Комбінований підхід, де система працює паралельно з традиційним антивірусом, показав синергетичний ефект з частотою виявлення дев'яносто вісім відсотків.

Детальний аналіз ефективності виявлення різних типів загроз показав найвищу ефективність для програм-шифрувальників з дев'яносто сім відсотків, впровадження коду в процеси з вісімдесят чотири – дев'яносто два відсотки залежно від техніки та викрадення облікових даних з дев'яносто шість відсотків. Середню ефективність продемонстровано для бічного переміщення з сімдесят один відсоток та механізмів збереження присутності з п'ятдесят – дев'яносто три відсотки залежно від конкретної техніки. Найнижчу ефективність система показала для технік обходу захисту, що відповідає очікуванням, оскільки ці техніки спеціально розроблені для обходу детекції.

Розроблена система підтвердила свою практичну цінність як доповнення до традиційних сигнатурних антивірусів, забезпечуючи додатковий рівень захисту через

поведінковий аналіз та виявлення аномалій. Мінімальна ресурсоємність дозволяє розгортання на типових робочих станціях без негативного впливу на користувацький досвід.

РОЗДІЛ 4. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Аналіз умов праці при роботі з комп'ютерною технікою

Робота з комп'ютерною технікою, зокрема при розробці та тестуванні систем кібербезпеки, характеризується тривалим перебуванням в статичному положенні, інтенсивним зоровим навантаженням, психоемоційною напругою та впливом комплексу шкідливих виробничих факторів. Згідно з Гігієнічною класифікацією праці за показниками шкідливості та небезпечності факторів виробничого середовища, важкості та напруженості трудового процесу, затвердженою наказом МОЗ України, робота програміста та фахівця з кібербезпеки відноситься до класу 3.1-3.2 (шкідливі умови праці першого-другого ступеня).

Основні шкідливі та небезпечні фактори при роботі з персональними комп'ютерами можна класифікувати за природою дії на фізичні, хімічні, біологічні та психофізіологічні. До фізичних факторів належать підвищений рівень електромагнітного випромінювання, недостатня освітленість робочої зони, підвищений рівень шуму, незадовільні параметри мікроклімату, підвищений рівень статичної електрики та можливість ураження електричним струмом [43, с. 118-120]. Хімічні фактори включають виділення озону, оксидів азоту та формальдегіду під час роботи обладнання, особливо лазерних принтерів. Психофізіологічні фактори представлені розумовим перенапруженням, перенапруженням зорового аналізатора, монотонністю праці, емоційними перевантаженнями та значними статичними навантаженнями.

Електромагнітне випромінювання від сучасних рідкокристалічних моніторів знаходиться на безпечному рівні завдяки використанню LED-підсвітки замість електронно-променевих трубок старих моделей. Проте системні блоки, джерела безперебійного живлення та мережеве обладнання створюють електромагнітні поля

промислової частоти п'ятдесят герц. За результатами вимірювань, проведених у типовому офісному приміщенні з десятьма робочими місцями, напруженість електричного поля складає двадцять п'ять-сорок вольт на метр при гранично допустимому рівні п'ятсот вольт на метр, а магнітна індукція становить нуль цілих два-нуль цілих чотири мікротесла при ГДР десять мікротесла згідно з ДСанПіН 3.3.6.096-2002. Таким чином, електромагнітна обстановка відповідає санітарним нормам [44-45].

Освітленість робочих місць має критичне значення для запобігання зорової втоми та прогресування міопії. Згідно з ДБН В.2.5-28:2018 "Природне і штучне освітлення", для робіт високої точності з комп'ютером нормована освітленість робочої поверхні становить п'ятсот люксів при комбінованому освітленні та триста люксів при загальному. Вимірювання люксметром показали, що в досліджуваному приміщенні освітленість на робочих столах коливається від двохсот п'ятдесяти до чотирьохсот двадцяти люксів при роботі тільки загального освітлення, що є недостатнім. Застосування настільних світильників з LED лампами підвищує освітленість до чотирьохсот п'ятдесяти-п'ятисот п'ятдесяти люксів, що відповідає нормативам. Коефіцієнт пульсації освітлення для LED джерел не перевищує п'яти відсотків при нормі до двадцяти відсотків для даного виду робіт.

Параметри мікроклімату суттєво впливають на працездатність та самопочуття працівників. Для приміщень з комп'ютерами категорія важкості робіт визначається як Іа-Іб (легка фізична робота), відповідно оптимальні параметри згідно з ДСанПіН 3.3.6.042-99 складають: температура повітря двадцять два-двадцять чотири градуси Цельсія у холодний період та двадцять три-двадцять п'ять градусів у теплий період, відносна вологість сорок-шістдесят відсотків, швидкість руху повітря не більше нуль цілих два метри на секунду. Моніторинг мікроклімату протягом робочого тижня показав середню температуру двадцять три градуси взимку та двадцять шість градусів влітку, відносну вологість п'ятдесят два-п'ятдесят вісім відсотків, швидкість повітря нуль цілих один метри на секунду. Температура влітку дещо перевищує

оптимальні значення, що вирішується використанням кондиціонування повітря [46-47].

Рівень шуму від системних блоків сучасних комп'ютерів, принтерів та систем вентиляції складає сорок два-п'ятдесят три децибели при вимірюванні шумоміром на відстані один метр від джерела. Згідно з ДСН 3.3.6.037-99 "Санітарні норми виробничого шуму, ультразвуку та інфразвуку", гранично допустимий рівень звуку для робіт з комп'ютером складає п'ятдесят децибел. Таким чином, в окремі моменти роботи принтерів спостерігається незначне перевищення норми на три децибели, що може бути скориговано розміщенням принтерів у окремому приміщенні або використанням звукопоглинаючих екранів.

Статична електрика накопичується на поверхнях мебелі, корпусів обладнання та одягу працівників, особливо в приміщеннях з низькою вологістю повітря. Напруга статичної електрики може досягати декількох кіловольт, що викликає неприємні відчуття при дотику до заземлених предметів. Для зменшення накопичення статичної електрики рекомендується підтримувати відносну вологість на рівні п'ятдесят-шістдесят відсотків, використовувати антистатичні покриття для підлоги, застосовувати іонізатори повітря та забезпечувати заземлення металевих конструкцій.

Тривала робота з комп'ютером призводить до перенапруження зорового аналізатора, що проявляється у вигляді синдрому комп'ютерного зору. Симптоми включають почервоніння очей, відчуття сухості та піску в очах, розмитість зображення, головний біль, біль у шиї та плечах. Причинами є фіксація погляду на близькій відстані протягом тривалого часу, зменшення частоти моргання з п'ятнадцяти-двадцяти разів на хвилину в нормі до п'яти-семи разів при роботі з монітором, блискавість екрану, недостатній контраст та розмір шрифтів. Профілактика включає дотримання режиму праці та відпочинку з перервами кожну годину, виконання гімнастики для очей, налаштування яскравості монітора відповідно до освітлення приміщення, використання принципу двадцять-двадцять-

двадцять (кожні двадцять хвилин дивитись на об'єкт на відстані двадцяти футів протягом двадцяти секунд).

Статичне навантаження на опорно-руховий апарат виникає внаслідок тривалого перебування в одній позі за робочим столом. Неправильна посадка призводить до перенапруження м'язів шиї, спини, плечового пояса та розвитку остеохондрозу, сколіозу, синдрому карпального каналу. Ергономічна організація робочого місця з регульованим кріслом, правильним розташуванням монітора на рівні очей та клавіатури на висоті ліктів, використання підставки для ніг та підтримки поперекового відділу хребта значно зменшує статичне навантаження. Регулярне виконання виробничої гімнастики для розвантаження хребта та покращення кровообігу є обов'язковим елементом режиму праці [48].

Психоемоційне навантаження при роботі фахівця з кібербезпеки обумовлене високою відповідальністю за збереження інформації, необхідністю швидкого прийняття рішень при виявленні інцидентів безпеки, роботою в умовах дефіциту часу та високою концентрацією уваги протягом тривалих періодів. Стресові ситуації виникають при реагуванні на кібератаки, розслідуванні інцидентів та відновленні систем після компрометації. Для зменшення психоемоційного навантаження необхідні чітка організація робочого процесу, розумний розподіл навантаження, можливість відпочинку та перемикання уваги, соціально-психологічна підтримка в колективі.

4.2. Вимоги до організації робочого місця

Організація робочого місця користувача персонального комп'ютера регламентується Державними санітарними правилами і нормами роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПіН 3.3.2.007-98 та стандартом ДСТУ ISO 9241. Правильна організація

робочого простору є основою профілактики професійних захворювань та підтримання високої працездатності протягом робочого дня.

Площа на одне робоче місце з комп'ютером повинна становити не менше шести квадратних метрів, а об'єм не менше двадцяти кубічних метрів. Для приміщень з дванадцятьма робочими місцями мінімальна площа складає сімдесят два квадратні метри. При плануванні розміщення робочих місць необхідно забезпечити достатню відстань між сусідніми столами не менше двох метрів та відстань від тильної поверхні монітора одного комп'ютера до екрану іншого не менше трьох метрів для мінімізації впливу електромагнітного випромінювання. Робочі столи розташовують таким чином, щоб природне світло падало збоку, переважно зліва, для запобігання відблисків на екрані монітора.

Робочий стіл повинен мати висоту сімсот двадцять п'ять міліметрів або можливість регулювання в діапазоні шістсот вісімдесяти-восьмисот міліметрів для адаптації до антропометричних характеристик різних користувачів. Ширина робочої поверхні має становити не менше тисячі двохсот міліметрів, глибина не менше восьмисот міліметрів для забезпечення можливості розміщення монітора на відстані шестисот-семисот міліметрів від очей користувача. Підстільний простір повинен мати висоту не менше шестисот міліметрів, ширину п'ятсот міліметрів та глибину четвертьсот п'ятдесяти міліметрів для вільного розміщення ніг. Поверхня столу має бути матовою з коефіцієнтом відбиття нуль цілих чотири-нуль цілих шість для запобігання відблисків та напруження зору [49, с. 157-160].

Робоче крісло має відповідати вимогам ергономіки та забезпечувати підтримку фізіологічно правильної пози протягом тривалої роботи. Обов'язковими елементами є регулювання висоти сидіння в діапазоні чотирьохсот-п'ятисот п'ятдесяти міліметрів, регулювання висоти та кута нахилу спинки для підтримки поперекового відділу хребта, п'ятипроменева опора на роликах для стійкості та мобільності, м'яке сидіння з заокругленим переднім краєм для запобігання передавлювання судин під колінами. Підлокітники дозволяють розвантажити м'язи плечового пояса під час

перерв у роботі з клавіатурою, але не повинні заважати під час активної роботи. Рекомендована ширина сидіння чотиреста-п'ятсот міліметрів, глибина четвертьсот-п'ятисот міліметрів.

Монітор розташовують прямо перед користувачем на відстані шестисот-семисот міліметрів від очей, що відповідає приблизно витягнутій руці. Верхня частина екрану має знаходитись на рівні або трохи нижче горизонтальної лінії погляду, що забезпечує природний кут зору донизу п'ятнадцять-двадцять градусів та зменшує напруження шийних м'язів. Кут нахилу екрана регулюється індивідуально для мінімізації відблисків від джерел освітлення. При використанні двох моніторів вони розташовуються симетрично відносно центральної осі користувача або основний монітор прямо, а додатковий під кутом тридцять градусів збоку [50, с. 105-111].

Клавіатура розміщується на поверхні столу на відстані сто-триста міліметрів від краю для можливості спирання зап'ясть. Оптимальний кут нахилу клавіатури складає п'ять-п'ятнадцять градусів. При тривалій роботі рекомендується використання гелевої або поролонової підкладки під зап'ястя для зменшення навантаження на карпальний тунель та профілактики синдрому карпального каналу. Бездротові клавіатури дозволяють гнучко налаштовувати положення для максимального комфорту, але потребують регулярної заміни батарей.

Миша розташовується на одному рівні з клавіатурою збоку від домінуючої руки на відстані, що дозволяє тримати плече в природному розслабленому положенні близько до тіла. Килимок для миші з підтримкою зап'ястя зменшує статичне навантаження на кисть. Ергономічні миші з вертикальним хватом переводять кисть у більш природне положення та зменшують пронацію передпліччя, що знижує ризик тунельного синдрому при тривалій роботі [51].

Підставка для документів використовується при введенні даних з паперових носіїв та розміщується на одному рівні з монітором між клавіатурою та екраном або збоку від монітора під аналогічним кутом зору. Це зменшує частоту переведення

погляду між різними площинами та амплітуду рухів голови, знижуючи втому шийних м'язів та очей.

Підставка для ніг рекомендується користувачам, у яких при правильній висоті столу та крісла ступні не повністю спираються на підлогу. Розміри підставки становлять ширину не менше трьохсот міліметрів, глибину четвертьсот міліметрів, кут нахилу регулюється в межах нуля-двадцяти градусів. Поверхня має бути рифленою для запобігання ковзанню ніг. Правильне положення ніг з опорою ступнями забезпечує стійку позу та зменшує навантаження на поперековий відділ [52, с. 282-297].

Освітлення робочого місця організовується за принципом комбінування загального та місцевого освітлення. Загальне освітлення створюється стельовими світильниками з люмінесцентними або LED лампами з розсіювачами для рівномірного розподілу світлового потоку. Місцеве освітлення реалізується настільними світильниками з регулюванням напрямку світлового потоку. Температура кольору джерел світла має відповідати характеру роботи: чотири тисячі-чотири тисячі п'ятсот кельвінів (нейтральний білий) оптимальні для роботи з комп'ютером, забезпечуючи достатню освітленість без надмірної блакитної складової, що може порушувати циркадні ритми.

Колірне оформлення приміщення впливає на психоемоційний стан та працездатність. Стіни рекомендується фарбувати в світлі пастельні тони з коефіцієнтом відбиття п'ятдесят-шістдесят відсотків: світло-бежевий, світло-зелений, світло-блакитний. Світлі тони візуально розширюють простір та покращують освітленість, а пастельні відтінки не дратують зір. Стеля має білий колір з коефіцієнтом відбиття сімдесят-вісімдесят відсотків для максимального відбиття світла. Підлога повинна мати матове покриття темнішого відтінку з коефіцієнтом відбиття тридцять-п'ятдесят відсотків для зменшення відблисків знизу. Антистатичні властивості підлогового покриття запобігають накопиченню статичної електрики.

Мікроклімат приміщення підтримується системами опалення, вентиляції та кондиціонування. Природна вентиляція через вікна з можливістю провітрювання доповнюється примусовою вентиляцією з кратністю повітрообміну не менше двох-трьох об'ємів на годину. Системи кондиціонування повітря з функцією очищення через фільтри та зволоження забезпечують оптимальні параметри температури та вологості протягом року. Обов'язковим є регулярне технічне обслуговування систем вентиляції з очищенням фільтрів для запобігання накопиченню пилу та мікроорганізмів [53, с. 515-521].

4.3. Електробезпека та пожежна безпека

Електробезпека при експлуатації комп'ютерної техніки є критично важливим аспектом охорони праці, оскільки персональні комп'ютери, периферійне обладнання та інженерні системи приміщення працюють від мережі змінного струму напругою двісті тридцять вольт частотою п'ятдесят герц. За ступенем небезпеки ураження електричним струмом приміщення з комп'ютерами відносяться до категорії без підвищеної небезпеки за відсутності умов, що підвищують ризик ураження струмом.

Основними причинами ураження електричним струмом при роботі з комп'ютерною технікою є дотик до струмоведучих частин обладнання з пошкодженою ізоляцією, робота з обладнанням при пошкодженому або відсутньому захисному заземленні, дотик до металевих корпусів обладнання, що опинились під напругою внаслідок пробою ізоляції, використання несправного або неякісного обладнання, проведення ремонтних робіт без відключення від мережі. Специфічною небезпекою є можливість ураження імпульсною високою напругою при роботі з джерелами безперебійного живлення, що містять акумуляторні батареї великої ємності [54, с. 21-27].

Основні заходи електробезпеки включають використання обладнання з класом захисту від ураження електричним струмом не нижче першого, що передбачає

наявність захисного заземлення металевих корпусів. Електричні розетки приміщення повинні мати контакт захисного заземлення, підключений до контуру заземлення будівлі з опором не більше чотирьох ом для мереж напругою двісті двадцять-триста вісімдесят вольт. Перевірка справності заземлення проводиться спеціалізованою організацією не рідше одного разу на рік з оформленням протоколу вимірювань.

Система електроживлення комп'ютерного обладнання організовується з урахуванням навантаження та вимог безпеки. Кожне робоче місце підключається через окрему групу з автоматичним вимикачем номінальним струмом шістнадцять ампер та пристроєм захисного вимкнення з диференційним струмом спрацювання тридцять міліампер часом спрацювання не більше нуль цілих нуль три секунди. ПЗВ захищає людину від ураження струмом витоку при пошкодженні ізоляції, відключаючи живлення до того, як струм через тіло людини досягне небезпечного рівня.

Розподільчі електричні щити розміщуються в доступних місцях з можливістю швидкого відключення живлення в аварійних ситуаціях. На щитах розміщується схема електроживлення з позначенням груп споживачів та номіналів захисних автоматів. Елементи щита мають маркування для ідентифікації. Доступ до електрощитів обмежується для запобігання несанкціонованим втручанням некваліфікованого персоналу.

Джерела безперебійного живлення забезпечують електроживлення критичного обладнання при зникненні або погіршенні якості мережевої напруги. UPS підбираються за потужністю з коефіцієнтом запасу півтора для забезпечення стабільної роботи при максимальному навантаженні. Акумуляторні батареї UPS містять свинцево-кислотні або літій-іонні елементи, що вимагає дотримання правил експлуатації: розміщення в провітрюваному приміщенні для відведення водню, що виділяється при зарядці, запобігання короткому замиканню клем, недопущення глибокого розряду для збереження ресурсу. Заміна акумуляторів проводиться після

закінчення терміну служби три-п'ять років або при зниженні ємності нижче сімдесяти відсотків від номінальної [55-56].

Електричні кабелі прокладаються в кабель-каналах або під фальшпідлогою для захисту від механічних пошкоджень та запобігання спотиканню. Використання подовжувачів мінімізується, при необхідності застосовуються якісні мережеві фільтри з захистом від перевантаження та короткого замикання. Заборонено послідовне з'єднання декількох подовжувачів через ризик перевантаження та перегріву з'єднань. Місця проходження кабелів через стіни та перекриття оснащуються захисними втулками для запобігання перетирання ізоляції.

Інструктаж з електробезпеки проводиться для всіх працівників при прийнятті на роботу, періодично не рідше одного разу на рік, а також позапланово при зміні обладнання або після нещасних випадків. Працівники навчаються правилам безпечної експлуатації обладнання, порядку дій при виявленні несправностей, способам надання першої допомоги при ураженні електричним струмом. Перевірка знань оформлюється протоколом з підписами інструктованих осіб.

Забороняється працівникам самостійно розкривати корпуси комп'ютерів та іншого обладнання, проводити ремонтні роботи без відключення від мережі, використовувати несправне обладнання з пошкодженими кабелями або корпусами, залишати ввімкнене обладнання без нагляду по закінченню робочого дня, доторкатися до обладнання мокрими руками або в умовах підвищеної вологості. Будь-які несправності обладнання повідомляються відповідальній особі для організації кваліфікованого ремонту.

Пожежна безпека в приміщеннях з комп'ютерною технікою регламентується Правилами пожежної безпеки в Україні та відомчими нормативними документами. За вибухопожежною та пожежною небезпекою приміщення з комп'ютерами відносяться до категорії В4 (помірна пожежна небезпека) згідно з НАПБ Б.07.005-86. Основними горючими матеріалами є пластикові корпуси обладнання, кабелі в ПВХ ізоляції, меблі з деревини та ДСП, папір документації [57-58].

Основні причини пожеж включають перевантаження електричної мережі з перегрівом проводів та контактів, короткі замикання в електропроводці або всередині обладнання, перегрів обладнання через несправність систем охолодження або експлуатацію в умовах недостатньої вентиляції, залишення ввімкненого обладнання без нагляду, використання несправних або неякісних електроприладів, куріння в неналежних місцях, порушення правил експлуатації нагрівальних приладів. Статистика показує, що більшість пожеж відбувається в неробочий час через залишене ввімкнене обладнання.

Профілактичні заходи пожежної безпеки включають дотримання норм проектування електромереж з достатнім запасом по навантаженню, використання якісних кабелів з негорючою або самозатухаючою ізоляцією, застосування автоматичних захистів від перевантаження та короткого замикання, регулярне технічне обслуговування обладнання з очищенням від пилу для запобігання перегріву, виключення джерел відкритого вогню в приміщенні, обмеження кількості горючих матеріалів. На робочих місцях заборонено зберігати зайві папери, картонні коробки, особисті речі в кількості, що створює пожежне навантаження.

Приміщення обладнується автоматичною пожежною сигналізацією з димовими оптико-електронними датчиками, розташованими на стелі з кроком не більше дев'яти метрів. При спрацюванні датчика сигнал передається на приймально-контрольний прилад, що активує звукову та світлову сигналізацію, відправляє повідомлення на пульт пожежної охорони та може ініціювати автоматичне відключення електроживлення неаварійних груп. Система сигналізації перевіряється щомісяця шляхом тестового спрацювання та обслуговується спеціалізованою організацією не рідше двох разів на рік [59, с. 526-558].

Евакуаційні виходи забезпечують безпечну та швидку евакуацію людей при виникненні пожежі. Ширина коридорів становить не менше півтора метрів, дверей не менше нуль цілих дев'ять метрів. Двері на шляхах евакуації відчиняються назовні за напрямком руху та не обладнуються замками, що вимагають ключа для відкриття

зсередини. Евакуаційні виходи позначаються світловими покажчиками з автономним живленням, що залишаються видимими при відключенні основного освітлення. План евакуації розміщується на видному місці і показує розташування виходів, засобів пожежогасіння та шляхів руху до безпечної зони.

Первинні засоби пожежогасіння розміщуються в легкодоступних місцях з розрахунку один вогнегасник на п'ятдесят квадратних метрів площі. Для приміщень з електрообладнанням застосовуються вуглекислотні вогнегасники типу ВВК-2 або порошкові ВП-5, дозволені для гасіння обладнання під напругою до тисячі вольт. Використання водних та пінних вогнегасників заборонено через ризик ураження електричним струмом. Вогнегасники розміщуються на висоті півтора метрів від підлоги, мають бирку з датою останньої перевірки та зарядки. Перевірка вогнегасників проводиться щоквартально візуально та щороку з перевіркою маси вогнегасної речовини [60].

Пожежні крани внутрішнього протипожежного водопроводу розміщуються в коридорах в пожежних шафах разом зі рукавами та стволами. Відстань до найвіддаленішої точки приміщення не перевищує двадцять метрів довжини струменю. Тиск води в системі підтримується не менше нуля цілих два мегапаскалі. Перевірка справності пожежних кранів проводиться двічі на рік з пробним пуском води.

Для серверних приміщень з критично важливим обладнанням рекомендується встановлення автоматичних систем пожежогасіння газового типу з використанням інертних газів або хладонів. Такі системи не пошкоджують обладнання при спрацюванні, на відміну від водяних або порошкових систем. Спрацювання відбувається при одночасному спрацюванні двох різних типів датчиків для запобігання хибній активації. Перед випуском газу подається попереджувальний сигнал для евакуації людей з приміщення.

Інструктаж з пожежної безпеки проводиться для всіх працівників при прийнятті на роботу та періодично не рідше одного разу на рік. Працівники вивчають правила

пожежної безпеки, плани евакуації, розташування засобів пожежогасіння, порядок дій при виявленні пожежі, правила користування вогнегасниками. Практичні тренування з евакуації проводяться не рідше одного разу на рік для відпрацювання навичок швидкого та організованого залишення приміщення.

Організаційні заходи включають призначення відповідальної за пожежну безпеку особи, розробку інструкцій та планів евакуації, проведення інструктажів та тренувань, забезпечення справності засобів пожежогасіння та сигналізації, контроль за дотриманням протипожежного режиму. На об'єкті ведеться журнал обліку первинних засобів пожежогасіння, журнал інструктажів з пожежної безпеки, наказ про призначення відповідальних осіб. Планові перевірки стану пожежної безпеки проводяться представниками Державної служби України з надзвичайних ситуацій не рідше одного разу на три роки для об'єктів без масового перебування людей.

4.4. Безпека в надзвичайних ситуаціях

Надзвичайна ситуація визначається згідно з Кодексом цивільного захисту України як порушення нормальних умов життєдіяльності людей на окремій території чи об'єкті, спричинене аварією, катастрофою, стихійним лихом чи іншою небезпечною подією, що призвело або може призвести до загибелі людей та значних матеріальних втрат. Для офісних приміщень з комп'ютерною технікою найбільш ймовірними є надзвичайні ситуації техногенного характеру, зокрема пожежі, вибухи, аварії систем життєзабезпечення, а також природного характеру такі як землетруси, урагани, повені.

Пожежі в будівлях є найбільш частим видом надзвичайних ситуацій техногенного характеру. Статистика показує, що щороку в Україні відбувається близько ста тисяч пожеж, значна частина яких припадає на житлові та громадські будівлі. Причинами можуть бути необережне поводження з вогнем, несправність електрообладнання, порушення технологічних процесів, підпали. Для офісних

приміщень найбільш вірогідні причини пов'язані з електрообладнанням та порушенням правил експлуатації.

Система реагування на пожежу включає декілька рівнів. Автоматична пожежна сигналізація забезпечує раннє виявлення осередку загоряння на стадії тління або появи диму. Сучасні адресні системи точно вказують місце спрацювання датчика, що дозволяє швидко локалізувати загрозу. При спрацюванні сигналізації автоматично активується система оповіщення з трансляцією попереджувальних повідомлень через гучномовці або сирени. Повідомлення містить інформацію про необхідність евакуації та вказівки щодо напрямку руху до виходів [61].

Евакуація людей з будівлі при пожежі повинна бути завершена до досягнення критичних значень небезпечних факторів пожежі на шляхах евакуації. Розрахунковий час евакуації залежить від площі приміщень, кількості людей, ширини евакуаційних шляхів та виходів. Для офісних будівель нормативний час евакуації не перевищує шести хвилин з верхніх поверхів типової висотності. Фактичний час повинен бути менше розрахункового з коефіцієнтом безпеки не менше півтора.

Алгоритм дій працівників при виявленні пожежі включає наступні кроки. По-перше, сповістити про пожежу оточуючих голосом або активацією ручного сповіщувача пожежної тривоги. По-друге, зателефонувати до пожежно-рятувальної служби за номером сто один або єдиної служби порятунку сто дванадцять, повідомивши адресу об'єкту, місце виникнення пожежі, своє прізвище та контактний телефон. По-третє, якщо осередок займання невеликий і немає загрози життю, спробувати загасити його первинними засобами пожежогасіння. По-четверте, при неможливості ліквідації пожежі негайно евакуюватись з приміщення, закривши вікна та двері для обмеження доступу повітря [62].

Евакуація здійснюється організовано під керівництвом відповідальних осіб згідно з планом евакуації. Рух відбувається швидким кроком без паніки та давки, при задимленні пересуватись пригнувшись або повзком, оскільки чисте повітря знаходиться біля підлоги. Дихати через мокру тканину для фільтрації диму та

охолодження вдихуваного повітря. Не використовувати ліфти, оскільки при пожежі їх відключають, і є ризик застрягти в шахті. Після виходу з будівлі віддалитись на безпечну відстань не менше п'ятдесяти метрів та перевірити наявність усіх працівників. Повернення в будівлю до прибуття пожежних та отримання дозволу заборонено.

Особливості евакуації при неможливості залишити приміщення через основні виходи внаслідок задимлення або блокування вогнем включають використання запасних виходів, сходів незадимлюваних сходових кліток, зовнішніх пожежних драбин. При відсутності можливості евакуації необхідно зачинитись у приміщенні з найменшою задимленістю, ущільнити двері мокрими тканинами, відкрити вікно для доступу свіжого повітря, подавати сигнали рятувальникам через вікно яскравою тканиною або ліхтариком [63-64]. Не розбивати вікна повністю, оскільки приплив кисню посилить горіння.

Землетруси на території України відбуваються рідко і переважно невеликої магнітуди, проте сейсмічно активні зони включають Карпатський регіон, Крим, окремі райони Одеської та Чернівецької областей. Можливі землетруси магнітудою до шести балів за шкалою MSK-64, що може призвести до пошкоджень будівель та паніки населення. Дії при землетрусі залежать від місцезнаходження.

У приміщенні при початку струсів необхідно негайно залишити будівлю, якщо це можна зробити швидко, в межах п'ятнадцяти-двадцяти секунд до посилення коливань. При неможливості швидкої евакуації зайняти безпечну позицію під міцним столом, дверним прорізом капітальної стіни, у кутку кімнати подалі від вікон та важких предметів, що можуть впасти. Захистити голову руками або подушкою. Не виходити на балкон, не використовувати ліфт, не запалювати вогонь через можливе пошкодження газопроводу. Після припинення поштовхів обережно залишити будівлю, перевірити можливі пошкодження комунікацій, при виявленні витоку газу або пошкодження електропроводки вимкнути відповідні системи.

На вулиці при землетрусі віддалитись від будівель, електричних стовпів, дерев на відстань не менше висоти найближчого об'єкту. Зайняти відкриту місцевість або площу. Якщо переміщення небезпечне, присісти і захистити голову руками. Після основного поштовху бути готовим до повторних афтершоків, які можуть відбуватись протягом годин або днів та мати значну силу.

Урагани, смерчі та сильні вітри можливі на всій території України, особливо в степових районах та на узбережжі морів. Попередження про наближення небезпечних метеорологічних явищ надходить від гідрометеослужби за декілька годин, що дозволяє вжити превентивних заходів. При отриманні штормового попередження необхідно закріпити або прибрати предмети, що можуть бути зірвані вітром, закрити вікна та вимкнути електроприлади. Найбезпечніше місце в будівлі це внутрішні кімнати без вікон, коридори, підвали.

Повені можуть загрожувати об'єктам, розташованим у заплавах річок або в низинних районах. При загрозі підтоплення організується завчасна евакуація людей і матеріальних цінностей до приходу води. Евакуація може бути упередженою за декілька днів або екстреною за кілька годин. Вимикається електроживлення для запобігання ураженню струмом у воді, закриваються вікна і двері першого поверху, цінні речі піднімаються на верхні поверхи. При раптовому підтопленні без можливості евакуації підніматись на верхні поверхи або дах будівлі, подавати сигнали рятувальникам, чекати прибуття допомоги [65].

Аварії систем життєзабезпечення включають відключення електроенергії, водопостачання, опалення, каналізації. Тривале відключення електроенергії особливо критичне для об'єктів з комп'ютерним обладнанням. Джерела безперебійного живлення забезпечують роботу протягом п'ятнадцяти-тридцяти хвилин для коректного завершення роботи та збереження даних. При тривалому відключенні організується використання мобільних генераторів для живлення критичних систем або призупинення роботи до відновлення електропостачання.

Резервування важливих даних на зовнішніх носіях або в хмарних сховищах захищає від втрати інформації.

Терористичні акти та загрози стали реальністю сучасності, тому працівники повинні знати ознаки підозрілих предметів і дії при їх виявленні. Підозрілим вважається залишений без нагляду пакет, сумка, коробка, особливо в місцях скупчення людей. Характерні ознаки включають незвичний вигляд предмету, натягнуті дроти або антени, звукові сигнали або тикання, запах хімічних речовин. При виявленні підозрілого предмету заборонено доторкатись, переміщувати, занурювати у воду, використовувати мобільні телефони поряд з ним. Необхідно негайно повідомити правоохоронні органи, організувати евакуацію людей з зони радіусом не менше ста метрів, очікувати прибуття саперів для обстеження.

Хімічні та радіаційні аварії хоч і мають низьку ймовірність для офісних об'єктів, але можливі при розташуванні поблизу хімічних підприємств або транспортних шляхів перевезення небезпечних вантажів. Оповіщення населення здійснюється через сирени та трансляцію інформації. При загрозі зараження необхідно герметизувати приміщення, закрити вікна та двері, вимкнути вентиляцію, використовувати засоби індивідуального захисту респіраторів або самодельні ватно-марлеві пов'язки, очікувати вказівок штабу цивільного захисту щодо евакуації або укриття.

Система цивільного захисту об'єкту включає організаційну структуру з призначенням відповідальних осіб, формування невоєнізованих формувань цивільного захисту, створення запасів засобів індивідуального захисту та медикаментів, облаштування укриттів або визначення найбезпечніших приміщень для укриття, розробку планів реагування на різні види надзвичайних ситуацій, проведення навчань та тренувань. Керівництво об'єкту несе відповідальність за організацію цивільного захисту та забезпечення безпеки працівників.

Медична аптечка першої допомоги знаходиться в легкодоступному місці та містить перев'язувальні матеріали, знеболюючі та серцеві препарати, антисептики,

термометр, тонометр, пристрій для штучного дихання. Працівники навчаються прийомам надання першої долікарської допомоги при різних травмах, опіках, ураженні електричним струмом, зупинці дихання та серцебиття. Телефони екстрених служб розміщуються на видних місцях.

Психологічна підготовка до дій в надзвичайних ситуаціях включає вироблення навичок зберігати спокій, швидко оцінювати обстановку, приймати правильні рішення в стресових умовах. Регулярні тренування та інструктажі формують автоматизм дій, що критично важливо в умовах дефіциту часу та панічних настроїв. Знання плану евакуації, розташування виходів та засобів захисту повинно бути автоматичним для кожного працівника [66].

Відновлення після надзвичайної ситуації включає оцінку збитків, ремонт пошкоджень, відновлення роботи систем життєзабезпечення, психологічну реабілітацію постраждалих, аналіз причин та розробку заходів для запобігання повторенню. Для IT-систем критично важливим є наявність резервних копій даних та планів аварійного відновлення, що дозволяють швидко відновити роботоздатність після збою обладнання чи втрати інформації.

Висновки до розділу 4

У четвертому розділі проведено комплексний аналіз питань охорони праці та безпеки в надзвичайних ситуаціях для робочих місць з комп'ютерною технікою, зокрема для розробників та тестувальників систем кібербезпеки.

Проаналізовано умови праці при роботі з персональними комп'ютерами, що характеризуються впливом комплексу шкідливих факторів: електромагнітного випромінювання, недостатньої освітленості, шуму, незадовільних параметрів мікроклімату, статичної електрики, психоемоційних та статичних навантажень. За результатами оцінки, робота відноситься до класу 3.1-3.2 шкідливих умов праці. Вимірювання показали, що більшість параметрів знаходяться в межах санітарних

норм, проте освітленість та температура повітря влітку потребують корекції через використання додаткового місцевого освітлення та систем кондиціонування.

Детально розглянуто вимоги до ергономічної організації робочого місця відповідно до діючих стандартів ДСанПіН 3.3.2.007-98 та ДСТУ ISO 9241. Визначено параметри робочого столу, крісла, розташування монітора, клавіатури, миші та іншого обладнання для забезпечення фізіологічно правильної пози та мінімізації навантаження на опорно-руховий апарат і зоровий аналізатор. Встановлено норми площі та об'єму на одне робоче місце не менше шести квадратних метрів та двадцяти кубічних метрів відповідно. Описано вимоги до організації освітлення з комбінацією загального та місцевого освітлення до рівня п'ятисот люксів, підтримання мікроклімату з температурою двадцять два-двадцять п'ять градусів та вологістю сорок-шістдесят відсотків.

Розглянуто питання електробезпеки при експлуатації комп'ютерного обладнання, що працює від мережі двісті тридцять вольт. Описано основні заходи захисту: застосування захисного заземлення з опором не більше чотирьох ом, використання пристроїв захисного вимкнення з диференційним струмом тридцять міліампер, організація електроживлення через групові автомати, застосування джерел безперебійного живлення. Визначено правила безпечної експлуатації обладнання та порядок інструктування працівників з електробезпеки.

Проаналізовано заходи пожежної безпеки для приміщень категорії В4. Описано систему автоматичної пожежної сигналізації з димовими датчиками, організацію евакуаційних шляхів та виходів з нормативним часом евакуації до шести хвилин, розміщення первинних засобів пожежогасіння з розрахунку один вогнегасник на п'ятдесят квадратних метрів. Визначено порядок дій працівників при виявленні пожежі: сповіщення оточуючих, виклик пожежної служби за номером сто один, спроба гасіння первинними засобами при відсутності загрози життю, організована евакуація згідно з планом.

Розглянуто можливі надзвичайні ситуації природного та техногенного характеру: пожежі, землетруси, урагани, повені, аварії систем життєзабезпечення, терористичні загрози, хімічні та радіаційні аварії. Для кожного виду визначено ймовірність виникнення, характерні ознаки, алгоритм дій працівників для збереження життя та здоров'я. Особливу увагу приділено організації евакуації при пожежі як найбільш ймовірній надзвичайній ситуації для офісних приміщень.

Описано систему цивільного захисту об'єкту, що включає призначення відповідальних осіб, формування невоєнізованих формувань, створення запасів засобів захисту, розробку планів реагування, проведення навчань та тренувань. Визначено необхідність наявності медичної аптечки першої допомоги та навчання працівників прийомам надання долікарської допомоги.

Реалізація описаних заходів охорони праці та безпеки в надзвичайних ситуаціях забезпечує створення безпечних та комфортних умов праці для розробників систем кібербезпеки, мінімізацію ризиків професійних захворювань та травматизму, готовність до ефективних дій при виникненні надзвичайних ситуацій, що в сукупності сприяє підвищенню продуктивності праці та збереженню здоров'я працівників.

ЗАГАЛЬНІ ВИСНОВКИ

У магістерській роботі вирішено актуальне науково-практичне завдання підвищення ефективності захисту інформаційних систем від сучасних кіберзагроз через розробку та реалізацію удосконаленої системи виявлення злоякісного, шпигунського та завідомо фальшивого програмного забезпечення на основі багаторівневого поведінкового аналізу та моніторингу системної активності на рівні операційної системи.

Основні наукові та практичні результати роботи полягають у наступному:

1. Проведено комплексний аналіз сучасного стану проблеми захисту від зловмисного програмного забезпечення, що показав стрімке зростання кількості та складності кіберзагроз з глобальними збитками понад вісім трильйонів доларів у 2023 році. Досліджено еволюцію типів malware від простих вірусів до складних APT кампаній з використанням zero-day вразливостей, fileless технік та living-off-the-land методів. Проаналізовано обмеження традиційних сигнатурних методів детекції, що виявляються неефективними проти поліморфного коду та раніше невідомих загроз. Розглянуто сучасні підходи до захисту, включаючи поведінковий аналіз, machine learning, sandboxing та threat intelligence, визначено їх переваги та недоліки. Встановлено, що комбінований підхід з інтеграцією множинних методів детекції забезпечує найвищу ефективність виявлення при прийнятному рівні хибних спрацювань.

2. Розроблено концепцію удосконаленої системи протидії злоякісному програмному забезпеченню на основі багаторівневого моніторингу та поведінкового аналізу. Сформульовано функціональні та нефункціональні вимоги до системи, включаючи необхідність виявлення поведінкових аномалій незалежно від наявності сигнатур, мінімальний вплив на продуктивність з overhead не більше п'яти-семи відсотків CPU, адаптивність до специфіки конкретного середовища через формування baseline, модульність архітектури для незалежного розвитку

компонентів. Визначено метрики ефективності з цільовими значеннями True Positive Rate понад дев'яносто п'ять відсотків, False Positive Rate менше одного відсотка, Mean Time To Detect менше п'яти хвилин для критичних загроз.

3. Запропоновано модульну чотирирівневу архітектуру системи, де системний рівень забезпечує взаємодію з Windows API, рівень моніторингу виконує збір подій через спеціалізовані модулі, рівень обробки та аналізу здійснює виявлення аномалій, рівень користувацького інтерфейсу надає засоби управління та візуалізації. Розроблено модуль моніторингу файлової системи з використанням ETW провайдера Microsoft-Windows-Kernel-File для відстеження операцій створення, модифікації та видалення файлів з багаторівневою фільтрацією шумних подій та багатокритеріальним детектором ransomware. Створено модуль контролю процесів, що комбінує real-time моніторинг через ETW з періодичним polling через PSAPI, будує дерево батьківсько-дочірніх зв'язків, виявляє injection техніки через аналіз послідовностей системних викликів від Threat-Intelligence провайдера. Реалізовано модуль перевірки цифрових підписів з використанням WinVerifyTrust API для верифікації Authenticode підписів, перевірки ланцюжка довіри та статусу відкликання сертифікатів. Розроблено модуль виявлення аномалій з трьома незалежними аналітичними движками для baseline-based, rule-based та heuristic детекції, що агрегують сигнали через обчислення інтегрального risk score.

4. Розроблено методи виявлення підозрілої активності для різних типів загроз. Метод виявлення ransomware базується на аналізі темпу файлових модифікацій понад сто файлів за хвилину, детекції створення ransom notes через регулярні вирази, виявленні масової зміни розширень файлів з обчисленням інтегральної оцінки підозрілості та автоматичним ініціюванням процедур реагування при перевищенні порогу. Метод виявлення process injection аналізує послідовності ETW подій ALLOCVM, WRITEVM, CREATETHREAD для DLL injection, CREATE_SUSPENDED, UNMAPVIEW, WRITEVM для process hollowing з перевіркою рівня довіри до процесів-ініціаторів через верифікацію цифрових

підписів. Метод контролю доступу до критичних процесів відстежує спроби читання пам'яті LSASS для виявлення credential dumping з whitelist легітимних програм, що мають право такого доступу. Метод аналізу частоти створення процесів формує baseline з експоненційно згладженим ковзним середнім, виявляє аномалії через z-score тест з порогом три стандартні відхилення для попереджень та п'ять для критичних алертів.

5. Реалізовано повнофункціональний програмний прототип системи моніторингу з використанням Python 3.11 як основної мови програмування, бібліотек pywin32 для доступу до Windows API, python-etw для роботи з Event Tracing, Tkinter для графічного інтерфейсу, SQLite для зберігання конфігурації та історії подій. Детально імплементовано всі модулі системи з ефективною обробкою подій, фільтрацією шумних активностей, кешуванням результатів перевірок для мінімізації overhead. Розроблено графічний інтерфейс з dashboard для real-time метрик, графіками активності процесів та мережі, списком алертів з фільтрацією та кольоровим кодуванням за severity, детальним переглядом подій з повним контекстом, розділом налаштувань для конфігурації параметрів детекції, менеджером правил для додавання власних detection rules, генератором аналітичних звітів.

6. Проведено комплексне тестування розробленої системи на наборі з п'ятдесяти реальних malware samples різних типів, включаючи ransomware, trojan, backdoor, worm з охопленням як відомих зразків старше шести місяців, так і свіжих варіантів молодше місяця. Система продемонструвала detection rate вісімдесят шість відсотків з виявленням сорока трьох samples, середній час виявлення чотири секунди від моменту першої підозрілої активності, false positive rate три відсотки протягом тижня безперервної роботи з п'ятьма хибними спрацюваннями на рідкісні системні утиліти. Найвищу ефективність показано для ransomware з detection rate дев'яносто сім відсотків, process injection вісімдесят чотири-дев'яносто два відсотки, credential dumping дев'яносто шість відсотків. Середню ефективність продемонстровано для

lateral movement сімдесят один відсоток, persistence mechanisms п'ятдесят-дев'яносто три відсотки в залежності від техніки.

7. Виконано порівняльний аналіз з трьома комерційними антивірусними рішеннями Windows Defender, Kaspersky Endpoint Security, ESET NOD32 на ідентичному наборі malware samples. Windows Defender показав detection rate вісімдесят два відсотки з часом виявлення сім секунд для відомих загроз. Kaspersky продемонстрував найвищий detection rate дев'яносто чотири відсотки з часом десять секунд, але false positive rate два відсотки. ESET показав вісімдесят вісім відсотків detection rate з найшвидшим часом п'ять секунд та найнижчим false positive один відсоток. Розроблена система з вісімдесят шість відсотків detection rate має найшвидший час виявлення чотири секунди завдяки real-time поведінковому аналізу без необхідності сканування файлів. Унікальна перевага полягає у здатності виявляти техніки атак незалежно від конкретного malware payload, що критично важливо для протидії zero-day загрозам. Комбінований підхід з паралельною роботою розробленої системи та традиційного антивірусу показав synergistic ефект з detection rate дев'яносто вісім відсотків.

8. Оцінено ресурсоємність розробленої системи, що складає три-чотири відсотки CPU на сучасному quad-core процесорі та двісті-триста мегабайт RAM після warmup періоду, що є comparable або нижче комерційних EDR рішень CrowdStrike Falcon п'ять-сім відсотків CPU, Carbon Black шість-вісім відсотків, SentinelOne чотири-шість відсотків. Long-running тест протягом семи днів не виявив memory leaks з стабілізацією споживання пам'яті на рівні трьохсот мегабайт. Disk IO навантаження мінімальне завдяки in-memory структурам та write-back кешуванню з flush раз на десять секунд. Система демонструє excellent efficiency придатну для розгортання на типових робочих станціях без негативного впливу на user experience.

9. Проаналізовано умови праці при роботі з комп'ютерною технікою з ідентифікацією шкідливих факторів: електромагнітного випромінювання в межах норм двадцять п'ять-сорок вольт на метр при ГДР п'ятсот, освітленості двісті

п'ятдесят-чотириста двадцять люксів при нормі п'ятсот з необхідністю додаткового місцевого освітлення, шуму сорок два-п'ятдесят три децибели при ГДР п'ятдесят з періодичними перевищеннями при роботі принтерів, мікроклімату з температурою двадцять три градуси взимку та двадцять шість влітку при нормі двадцять два-двадцять п'ять, психоемоційних навантажень через високу відповідальність та дефіцит часу при реагуванні на інциденти. Розроблено вимоги до ергономічної організації робочого місця з площею не менше шести квадратних метрів, регульованими меблями, правильним розташуванням обладнання для мінімізації статичних навантажень на опорно-руховий апарат. Визначено заходи електробезпеки з використанням захисного заземлення опором не більше чотирьох ом, ПЗВ з диференційним струмом тридцять міліампер, інструктування працівників. Описано систему пожежної безпеки з автоматичною сигналізацією, первинними засобами гасіння один вогнегасник на п'ятдесят квадратних метрів, евакуаційними виходами з нормативним часом до шести хвилин. Розглянуто дії в надзвичайних ситуаціях різних типів з алгоритмами збереження життя та здоров'я.

Практичне значення отриманих результатів підтверджується створенням працездатного програмного прототипу системи моніторингу та виявлення загроз, що може використовуватись малими та середніми підприємствами як доступна альтернатива дорогим комерційним EDR рішенням або як доповнення до традиційних антивірусів для багаторівневого захисту. Модульна архітектура та відкритий формат правил детекції дозволяють адаптувати систему до специфічних потреб різних організацій та розвивати її функціональність без глибоких змін в кодовій базі.

Напрямки подальших досліджень включають розширення покриття MITRE ATT&CK техніки через додавання спеціалізованих детекторів для рідкісних методів атак, інтеграцію machine learning моделей для автоматичного виявлення нових патернів підозрілої поведінки без необхідності ручного написання правил, розробку network traffic analyzer для виявлення lateral movement та command-and-control

комунікацій, створення централізованого backend для корпоративних розгортань з можливостями cross-endpoint correlation та threat hunting, реалізацію автоматизованих процедур реагування з ізоляцією скомпрометованих систем та збиранням forensic artifacts, портування системи на інші операційні системи Linux та macOS для забезпечення кросплатформенного захисту гетерогенних мереж.

Розроблена система підтверджує перспективність поведінкового підходу до виявлення зловмисного програмного забезпечення та демонструє можливість створення ефективних засобів захисту з мінімальною ресурсоємністю, що є критично важливим для широкого впровадження сучасних технологій кібербезпеки в організаціях з обмеженими ресурсами.

Оформлення результатів цього дослідження здійснювалося згідно з методичними рекомендаціями кафедри [67].

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Dobryshyn Yu. Класифікація та кодування дефектів програмного забезпечення внаслідок дії кібератак. COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION. 2025. № 58. С. 59–69. URL: https://cit.lntu.edu.ua/index.php/cit/article/view/685 (дата звернення: 01.12.2025).

2. Грищук А. Б., Хімко Я. П. Класифікація шкідливого програмного забезпечення. РЕДАКЦІЙНА КОЛЕГІЯ. 2024. С. 45. URL: https://dspace.lvduvs.edu.ua/bitstream/1234567890/6900/1/22_12_2023.pdf#page=45 (дата звернення: 01.12.2025).

3. Григоренко Роман. Види та можливості сучасного програмного забезпечення. С. 117. URL: https://ecogosfond.kz/wp-content/uploads/2023/12/A-PARATTY-TEHNOLOGIJaLARDY-ZhA-ANDY-JeKOLOGIJa-A-SERI.pdf#page=117 (дата звернення: 01.12.2025).

4. Скуратоський, Є., Аносов, А., Стрельніков, В., & Кучерявий, М. (2025). Експерименти та практичні рішення побудови тестового середовища для перевірки рівня безпеки на рівні додатків. Кібербезпека: освіта, наука, техніка, 3(31), 217–226. https://doi.org/10.28925/2663-4023.2025.31.1014

5. Федієнко О. П. Загрозливі тенденції використання державою-агресором шкідливого програмного забезпечення в умовах правового режиму воєнного стану. Інформація і право. 2023. № 3 (46). С. 142–153. URL: http://il.ippi.org.ua/article/view/287219 (дата звернення: 01.12.2025).

6. Кіцель Н. В., Владов С. І. Тенденції розвитку програм-вимагачів. The 2nd International scientific and practical conference "Science and education: synergy of innovation" (September 28–30, 2025). Berlin, Germany: MDPC Publishing, 2025. 449 p. URL: https://sci-conf.com.ua/wp-content/uploads/2025/09/SCIENCE-AND-EDUCATION-SYNERGY-OF-INNOVATION-28-30.09.25.pdf#page=154 (дата звернення: 01.12.2025).

7. D. Zhuravchak, A. Tolkachova, A. Piskozub, V. Dudykevych, N. Korshun, Monitoring ransomware with Berkeley Packet Filter, CEUR Workshop Proceedings 3550 (2023) 95–106.

8. Бондаренко В. Г., Крушельницька М. О. Аналіз та класифікація шкідливих програм. Стан, досягнення та перспективи інформаційних систем і технологій. 2022. С. 139.

9. Мартинюк Г. В., Мартинюк І. В., Проценко Б. П. Аналіз сучасних методів стегоаналізу аудіосигналів. Безпека інформації. 2024. Т. 30. № 3. С. 393–398. URL: https://jrnl.nau.edu.ua/index.php/Infosecurity/article/view/20360 (дата звернення: 01.12.2025).

10. Спасітєлєва, С., Чичкань, І., Шевченко, С., & Жданова, Ю. (2023). Розробка безпечних контейнерних застосунків з мікросервісною архітектурою. Кібербезпека: освіта, наука, техніка, 1(21), 193–210. https://doi.org/10.28925/2663-4023.2023.21.193210

11. Діденко М. С. Евристичний аналіз як метод виявлення небажаного програмного забезпечення. Проблеми інформатизації: матеріали XI міжнародної науково-технічної конференції. Харків, 2023. С. 26. URL: https://repository.kpi.kharkov.ua/server/api/core/bitstreams/295b361e-f784-4756-a78d-79f28cf4fc89/content (дата звернення: 01.12.2025).

12. Саприкін Олександр. Евристичний аналізатор шкідливого коду на основі штучної нейронної мережі. Science, theory and practice. Abstracts of XXIX International Scientific and Practical Conference. 2021. URL: https://isg-konf.com/wp-content/uploads/2021/06/XXIX-Conference-June-08-11-2021-Tokyo-Japan.pdf#page=561 (дата звернення: 01.12.2025).

13. Костюк, Ю., Складанний, П., Рзаєва, С., Самойленко, Ю., & Коршун, Н. (2025). Інтелектуальні системи керування та захисту в кіберфізичних і хмарних середовищах Smart Grid. Кібербезпека: освіта, наука, техніка, 2(30), 125–156. https://doi.org/10.28925/2663-4023.2025.30.956

14. Вірна Ж. П. Поведінковий аналіз: концепції, моделі, практичні технології. 2025. URL: https://evnuir.vnu.edu.ua/bitstream/123456789/28606/1/pov_an2025.pdf (дата звернення: 01.12.2025).

15. Костюк, Ю., Складанний, П., Рзаєва, С., Мазур, Н., Черевик, В., & Аносов, А. (2025). Особливості реалізації мережевих атак через TCP/IP-протоколи. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка», 1(29), 571–597. https://doi.org/10.28925/2663-4023.2025.29.915

16. Негоденко В., Шевченко С., Негоденко О., Золотухіна О. (2025). Інтеграція теорії катастроф у моделі прийняття рішень для систем управління інформаційною безпекою. Телекомунікаційні та інформаційні технології, 4(89), 20-28. https://doi.org/10.31673/2412-4338.2025.048903

17. Хомко Леся, Дубно Олег. Регуляторні «пісочниці» як інструмент розвитку правового регулювання штучного інтелекту: міжнародний досвід. 2023. С. 357. URL: https://dspace.lvduvs.edu.ua/bitstream/1234567890/6862/1/Тези_08-12-23__23-01-24.pdf#page=357 (дата звернення: 01.12.2025).

18. Колодійчук А. В., Важинський Ф. А. Українська модель ринку антивірусного програмного забезпечення. The 27th International scientific and practical conference "Trends of young scientists regarding the development of science" (July 11–14, 2023). Edmonton, Canada: International Science Group, 2023. 225 p. URL: https://books.google.com.ua/books?hl=uk&lr=&id=YpnKEAAAQBAJ (дата звернення: 01.12.2025).

19. Delaney Jeffrey. The effectiveness of antivirus software: MS thesis. Utica College, 2020. URL: https://www.proquest.com/openview/d3ff27e1e773c8dd36e7746e64567702/1 (дата звернення: 01.12.2025).

20. Бригинець А. А. Технології аналізу та виявлення файлів-поліглотів у корпоративних системах безпеки. Цифрова трансформація кібербезпеки. С. 16. URL:

https://duikt.edu.ua/uploads/p_2779_36484696.pdf#page=16 (дата звернення: 01.12.2025).

21. Балик Уляна. Порівняльний аналіз ефективності традиційних та цифрових PR-інструментів у різних секторах бізнесу. Економіка та суспільство. 2024. № 67. URL: https://economyandsociety.in.ua/index.php/journal/article/view/4717/4658 (дата звернення: 01.12.2025).

22. H. Shevchenko, et al., Information Security Risk Analysis SWOT, Cybersecurity Providing in Information and Telecommunication Systems 2923 (2021) 309-317.

23. Y. Kostiuk, et al., Integrated protection strategies and adaptive resource distribution for secure video streaming over a Bluetooth network, in: Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3826 (2024) 129–138.

24. Кулєшов М. М., Ященко О. А. Актуальні питання реалізації завдань цивільного захисту в умовах сучасних викликів та загроз. 2022. URL: http://repositsc.nuczu.edu.ua/bitstream/123456789/16882/1/Kuleshov.pdf (дата звернення: 01.12.2025).

25. Янко А. С., Вигівський Р. А. Система захисту комп'ютерної мережі. 2022. URL: https://reposit.nupp.edu.ua/bitstream/PoltNTU/12053/1/document%20%281%29-91-94.pdf (дата звернення: 01.12.2025).

26. Хохлачова Ю. Є., Гаврилова А. А. Аналіз загроз безпеки інформації в сучасних інформаційно-комунікаційних системах і мережах. Challenges and threats to critical infrastructure. 2023. С. 42. URL: http://repositsc.nuczu.edu.ua/bitstream/123456789/17919/1/Monograph-09-06-2023.pdf#page=42 (дата звернення: 01.12.2025).

27. Задворний, Д., Козачок, В., Черевик, В., Бодненко, Д., & Добришин, Ю. (2025). Методи та засоби побудови комплексної системи захисту інформації типового об’єкта інформаційної діяльності. Кібербезпека: освіта, наука, техніка, 3(31), 762–772. https://doi.org/10.28925/2663-4023.2025.31.1073

28. Крючкова, Л., & Ємельяненко, М. (2024). Захист web-ресурсу intranet від зовнішніх і внутрішніх загроз. Кібербезпека: освіта, наука, техніка, 3(23), 318–327. https://doi.org/10.28925/2663-4023.2024.23.318327

29. Wang Yu et al. RetTag: Hardware-assisted return address integrity on RISC-V. Proceedings of the 15th European Workshop on Systems Security. 2022. URL: https://dl.acm.org/doi/pdf/10.1145/3517208.3523758 (дата звернення: 01.12.2025).

30. Рожок С. М., Гулак Н. К. Програмний модуль перевірки сайтів на автентичність. World educational trends: lifelong learning in the information society. 2024. С. 64. URL: https://books.google.com.ua/books?hl=uk&lr=&id=EaYyEQAAQBAJ (дата звернення: 01.12.2025).

31. Державна податкова служба України та ін. Програмне забезпечення «Соната»: основні модулі та особливості звітування. Актуальні проблеми та перспективи розвитку обліку, аналізу та контролю в соціально-орієнтованій системі управління підприємством: Матеріали V Всеукраїнської науково-практичної конференції. Полтава, 14–15 квітня 2022 р. Полтава, 2022. 847 с. С. 174.

32. Hloba Yelyzaveta et al. Метод виявлення аномалій в корпоративній мережі. Системи управління, навігації та зв'язку. Збірник наукових праць. 2025. Т. 3. № 81. С. 193–198. URL: https://journals.nupp.edu.ua/sunz/article/view/4029/3379 (дата звернення: 01.12.2025).

33. Roka Sanjay et al. Anomaly behavior detection analysis in video surveillance: a critical review. Journal of Electronic Imaging. 2023. Vol. 32. № 4. P. 042106–042106. URL: https://www.spiedigitallibrary.org/journals/journal-of-electronic-imaging/volume-32/issue-4/042106 (дата звернення: 01.12.2025).

34. Khan Hamid Masood et al. Anomalous Behavior Detection Framework Using HTM-Based Semantic Folding Technique. Computational and Mathematical Methods in Medicine. 2021. Vol. 2021. № 1. P. 5585238. URL: https://onlinelibrary.wiley.com/doi/full/10.1155/2021/5585238 (дата звернення: 01.12.2025).

35. Ворохоб, М., Киричок, Р., Яскевич, В., Добришин, Ю., & Сидоренко, С. (2023). Сучасні перспективи застосування концепції zero trust при побудові політики інформаційної безпеки підприємства. Кібербезпека: освіта, наука, техніка, 1(21), 223–233. https://doi.org/10.28925/2663-4023.2023.21.223233

36. Крючкова Л., Складанний П., Ворохоб М. (2023). Передпроєктні рішення щодо побудови системи авторизації на основі концепції Zero Trust. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка», 3(19), 226–242. https://doi.org/10.28925/2663-4023.2023.13.226242

37. Целованський Т. Р., Шикула О. М. Дослідження методів збільшення швидкодії підсистеми пам'яті у електронно обчислювальних машинах. Наукові записки Державного університету інформаційно-комунікаційних технологій. 2024. № 2. С. 227–235.

38. Чернігівський, І., & Крючкова, Л. (2024). Тестування антивірусних рішень для корпоративного сегменту. Безпека інформації, 30(3), 407–413. https://doi.org/10.18372/2225-5036.30.20362

39. Корнієць, В., & Складанний, П. (2024). Формування вимог до архітектури і функцій систем моніторингу кібербезпеки. Телекомунікаційні та інформаційні технології, 4(85), 90–96. https://doi.org/10.31673/2412-4338.2024.040224

40. Фединець Н. І., Кухарська Н. П., Полотай О. І. Дослідження загроз інформаційної безпеки та способів їх вирішення в комп'ютерних мережах на канальному рівні. 2024. URL: https://sci.ldubgd.edu.ua/jspui/bitstream/123456789/13666/1/Стаття.pdf (дата звернення: 01.12.2025).

41. Konev Anton et al. A survey on threat-modeling techniques: protected objects and classification of threats. Symmetry. 2022. Vol. 14. № 3. P. 549. URL: https://www.mdpi.com/2073-8994/14/3/549 (дата звернення: 01.12.2025).

42. Luknar Ivana, Jovanović Filip. Various types of cyber threats. Srpska politička misao. 2024. Vol. 83. № 1. P. 161–177. URL: http://repozitorijumips.rs/1480/1/0354-59892401161L.pdf (дата звернення: 01.12.2025).

43. Лазуткін Микола, Журавель Сергій, Журавель Микола. Рекомендації студентам щодо поліпшення умов праці при використанні комп'ютерної техніки та технологій. Молодий вчений. 2021. № 2 (90). С. 118–120. URL: https://molodyivchenyi.ua/index.php/journal/article/view/384/373 (дата звернення: 01.12.2025).

44. Довгаль Р. Г. Інтеграція здоров'язбережувальних технологій у процес навчання комп'ютерних дисциплін майбутніх педагогів професійного навчання. Проблеми інженерно-педагогічної освіти. 2025. № 84.

45. Куріс Юрій, Матяшева Оксана. Дослідження впливу параметрів світлового середовища на здоров'я людини впродовж робочої зміни. Молодий вчений. 2021. № 5 (93). С. 177–180. URL: https://molodyivchenyi.ua/index.php/journal/article/view/605/588 (дата звернення: 01.12.2025).

46. Zhenjing Gu et al. Impact of employees' workplace environment on employees' performance: a multi-mediation model. Frontiers in public health. 2022. Vol. 10. P. 890400. URL: https://www.frontiersin.org/journals/public-health/articles/10.3389/fpubh.2022.890400/full (дата звернення: 01.12.2025).

47. Rasool Samma Faiz et al. How toxic workplace environment effects the employee engagement: The mediating role of organizational support and employee wellbeing. International journal of environmental research and public health. 2021. Vol. 18. № 5. P. 2294. URL: https://www.mdpi.com/1660-4601/18/5/2294 (дата звернення: 01.12.2025).

48. Ipsen Christine et al. Six key advantages and disadvantages of working from home in Europe during COVID-19. International journal of environmental research and

public health. 2021. Vol. 18. № 4. P. 1826. URL: https://www.mdpi.com/1660-4601/18/4/1826 (дата звернення: 01.12.2025).

49. Мамчур А. М., Кудря О. В. Вимоги до організації безпечного робочого місця за спеціалізованим швейним обладнанням. Охорона праці: Освіта і практика, «Проблеми та перспективи розвитку охорони праці»*. С. 157–160. URL: https://sci.ldubgd.edu.ua/jspui/bitstream/123456789/10091/1/Охорона%20праці%20освіта%20і%20практика.pdf#page=158 (дата звернення: 01.12.2025).

50. Vyshnovetska Svitlana. Особливості робочого місця осіб, які працюють поза місцезнаходженням роботодавця: правові аспекти. Scientific works of Kyiv Aviation Institute. Series Law Journal "Air and Space Law". 2023. Vol. 1. № 66. P. 105–111. URL: https://jrnl.nau.edu.ua/index.php/UV/article/view/17424 (дата звернення: 01.12.2025).

51. Рогач Ю. П., Шац Н. Ю. Щодо організації робочого місця та умов праці викладачів кафедри цивільної безпеки. Матеріали конференції. С. 326. URL: http://repositsc.nuczu.edu.ua/bitstream/123456789/21102/1/Збірник%20PES-2024_compressed.pdf#page=328 (дата звернення: 01.12.2025).

52. Roskams Michael, Haynes Barry. Employee-workplace alignment: Employee characteristics and perceived workplace requirements. Facilities. 2020. Vol. 38. № 3/4. P. 282–297. URL: https://www.emerald.com/f/article-abstract/38/3-4/282/226753 (дата звернення: 01.12.2025).

53. Rahaman Md Atikur et al. What factors do motivate employees at the workplace? Evidence from service organizations. The Journal of Asian Finance, Economics and Business. 2020. Vol. 7. № 12. P. 515–521.

54. Berezhanskyi T. H., Pazen O. Yu., Prydatko V. V. Дослідження передумов створення автоматичного пристрою електробезпеки рятувальника. Пожежна безпека. 2023. № 43. С. 21–27. URL: https://journal.ldubgd.edu.ua/index.php/PB/article/view/2636/2525 (дата звернення: 01.12.2025).

55. Манідіна Є. А., Грідяєв В. В. Пожежна безпека. 2024. URL: https://dspace.znu.edu.ua/jspui/bitstream/12345/25038/1/0060722.pdf (дата звернення: 01.12.2025).

56. Мілінчук Д. В. Пожежна небезпека термічних та потужних електропристроїв. URL: https://sci.ldubgd.edu.ua/bitstream/123456789/10128/1/Problems%20and%20prospects%20of%20development%20of%20life%20safety%20system.pdf#page=60 (дата звернення: 01.12.2025).

57. Кулаков О. В., Терещенко Ю. О. Забезпечення електробезпеки в захисних спорудах цивільного захисту: кваліфікаційна робота. НУЦЗУ, 2025. URL: http://repositsc.nuczu.edu.ua/bitstream/123456789/25052/1/Терещенко%20%28Кулаков%29%202025%20в%20репозитарій.pdf (дата звернення: 01.12.2025).

58. Зобенко О. О. Підвищення ефективності протипожежного захисту електричних мереж в місцях комутації надмірних споживчих потужностей. 2024. URL: http://repositsc.nuczu.edu.ua/bitstream/123456789/25893/1/Дисертація%20Зобенко%20остання.pdf (дата звернення: 01.12.2025).

59. Ehrenwerth Jan. Electrical and fire safety. Anesthesia Equipment. WB Saunders, 2021. P. 526–558. URL: https://www.sciencedirect.com/science/chapter/edited-volume/abs/pii/B9780323672795000248 (дата звернення: 01.12.2025).

60. Yang Rebecca et al. Fire safety requirements for building integrated photovoltaics (BIPV): A cross-country comparison. Renewable and Sustainable Energy Reviews. 2023. Vol. 173. P. 113112. URL: https://www.sciencedirect.com/science/article/abs/pii/S1364032122009935 (дата звернення: 01.12.2025).

61. Стручок В. С. Безпека в надзвичайних ситуаціях. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної та заочної (дистанційної) форм навчання. 2022. URL:

https://elartu.tntu.edu.ua/bitstream/lib/39196/1/Метод%20посібник%20Безпека%20в%20надзвичайних%20ситуаціях.pdf (дата звернення: 01.12.2025).

62. Кубатко О. В., Дяденко О. В. Цифрові трансформації для безпеки персоналу підприємства в умовах надзвичайних ситуацій. 2024. URL: https://essuir.sumdu.edu.ua/server/api/core/bitstreams/f37d300c-b646-44cf-a558-875227cba0ad/content (дата звернення: 01.12.2025).

63. Вамболь С. О. та ін. Безпека в надзвичайних ситуаціях. 2024.

64. Тютюник Вадим та ін. Особливості функціонування системи ситуаційних центрів на різних стадіях розвитку надзвичайних ситуацій. Сучасні інформаційні технології у сфері безпеки та оборони. 2022. Т. 43. № 1. С. 41–52. URL: https://sit.nuou.org.ua/article/view/256609/256199 (дата звернення: 01.12.2025).

65. Корецький Ю. О. Екологічна безпека у надзвичайних ситуаціях як складова національної безпеки. Редакційна колегія. 2020. С. 68. URL: http://repositsc.nuczu.edu.ua/bitstream/123456789/10660/1/zbir2020.pdf#page=68 (дата звернення: 01.12.2025).

66. Стратегія національної безпеки України. Безпека людини–безпека країни. Указ Президента України від 14 вересня 2020 року № 392/2020. URL: https://fisu.gov.ua/download/activity/ukaz-strateg-nats-bezpeky.pdf (дата звернення: 01.12.2025).

67. Жданова, Ю. Д., Складанний, П. М., & Шевченко, С. М. (2023). Методичні рекомендації до виконання та захисту кваліфікаційної роботи магістра для студентів спеціальності 125 Кібербезпека та захист інформації. https://elibrary.kubg.edu.ua/id/eprint/46009/1/Y_Zhdanova_P_Skladannyi_S_Shevchenko_MR_Master_2023_FITM.pdf