Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“____” _____ 2025 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків”

Виконав здобувач вищої освіти
ІІ курсу, групи КН-24М
ОПП «Комп’ютерні науки»
спеціальності 122 «Комп’ютерні науки»
_____ Слабінога Р.Ю.
«____» _____ 2025 р.

Керівник проекту
кандидат технічних наук
_____ Лисенко І.А.
«____» _____ 2025 р.
Рецензент _____
_____

м. Кропивницький

АНОТАЦІЯ

Слабінога Р.Ю. Дослідження та програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків. 122 Комп'ютерні науки. Центральноукраїнський національний технічний університет. Кропивницький. 2025.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи інтелектуального моніторингу та запобігання активності додатків.

Метою розробки є дослідження та програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків.

Об'єктом дослідження є процес інтелектуального моніторингу та запобігання активності додатків.

Предметом дослідження є методи інтелектуального моніторингу та запобігання активності додатків.

Методи дослідження базуються на методах інтелектуального моніторингу, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерні науки, інтелектуальний моніторинг, запобігання активності додатків

ABSTRACT

Slabinoha R.Yu. Research and software implementation of the system of intelligent monitoring and prevention of application activity. 122 Computer Science. Central Ukrainian National Technical University. Kropyvnytskyi. 2025.

In this final qualification work for the second (master's) level of higher education, software has been developed, which is intended for the system of intelligent monitoring and prevention of application activity.

The purpose of the development is the research and software implementation of the system of intelligent monitoring and prevention of application activity.

The object of the research is the process of intellectual monitoring and prevention of application activity.

The subject of the research is the methods of intellectual monitoring and prevention of application activity.

The research methods are based on the methods of intellectual monitoring, methods of mathematical statistics, methods of software development.

The result of the work is the software implementation of the system of intellectual monitoring and prevention of application activity.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software tools are provided.

The program can be used on PCs with Windows 10/11.

The program is developed in the Python environment.

Keywords: computer science, intelligent monitoring, application activity prevention

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

КМ	–	комп’ютерна мережа
КСАЗ	–	комплексна система антивірусного захисту
МЕ	–	міжмережний екран
ПЗ	–	програмне забезпечення
ПК	–	персональний комп’ютер
ACL	–	Access Control List
FTP	–	File Transfer Protocol
http	–	HyperText Transfer Protocol
POP3	–	Post Office Protocol Version 3
SMTP	–	Simple Mail Transfer Protocol
VLAN	–	Virtual Local Area Network

ВСТУП

Актуальність теми. Малоймовірно, що ваш комп'ютер буде заражений вірусом, але в той же час існує небезпека зараження іншими типами шкідливих програм і онлайн-погроз. Трояни-вимагачі зашифровують важливі файли й чекають оплати викупу, перш ніж розблокувати доступ до них. Банківські трояни втручаються в онлайн транзакції й намагаються украсти кошти. Зараження ботнетом зробить ваш комп'ютер ланкою в ланцюзі пристроїв, використовуваних для організації DDoS-атак. З цих і багатьох інших причин, ви повинні захистити свій комп'ютер за допомогою антивірусу.

Багато які із представлених антивірусів є безкоштовними тільки для некомерційного використання. Якщо ви хочете захистити комп'ютери в організації, то прийдеться придбати платну версію. У цьому випадку варто розглянути перехід на повноцінний комплексний антивірус. Зрештою, від цього залежить безпека вашого бізнесу. Якщо захистити потрібно великі підприємства, то на допомогу приходять SaaS-рішення, які дозволяють централізовано виконувати моніторинг і управляти захистом всіх комп'ютерів у компанії.

Ваш антивірус повинен надійно видаляти шкідливі програми, що вкоренилися в системі, але його основне завдання – запобігання нових заражень троянами-шифрувальниками, ботнетами, троянами й іншими видами погроз. Всі представлені в даному рейтингу антивіруси пропонують захист реального часу проти шкідливих атак. Багато продуктів пропонують надійний веб-захист, що блокує доступ до джерел шкідливих об'єктів і запобігає уведення конфіденційних даних на шахрайських сайтах.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

– Огляд існуючих систем інтелектуального моніторингу та запобігання активності додатків.

– Дослідження системи інтелектуального моніторингу та запобігання активності додатків.

– Програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків.

Об'єктом дослідження є процес інтелектуального моніторингу та запобігання активності додатків.

Предметом дослідження є методи інтелектуального моніторингу та запобігання активності додатків.

Методи дослідження базуються на методах інтелектуального моніторингу, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод інтелектуального моніторингу та запобігання активності додатків.

– Розроблено вітчизняний продукт інтелектуального моніторингу та запобігання активності додатків, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі інтелектуального моніторингу та запобігання активності додатків.

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти LV науково-технічної конференції «Наука в ЦНТУ: основні

досягнення та перспективи розвитку» (2025 р.), основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №15.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Безпека ПК є однією з головних умов його ефективної й злагодженої роботи. Будь-яка антивірусна програма призначена для виявлення шкідливих програм і комп'ютерних вірусів, які заражають файли й блокують роботу комп'ютера.

Для злагодженої й чіткої роботи ПК установлювати потрібно тільки одну антивірусну програму. Якщо ж установити два й більше антивіруси, вони почнуть між собою конфліктувати, що приведе до збоїв у комп'ютері. Перш ніж придбати антивірус, можна встановити пробну версію, що пропонують розроблювачі. Вона може бути розрахована від 30 до 90 днів. Однак існують також і безкоштовні антивіруси, які послужать гарним захистом для вашого комп'ютера. Основна відмінність таких антивірусних програм від платних – це спеціальний сканер, що знаходить і видаляє віруси тільки тоді, коли Ви його запускаєте.

1.2 Область застосування

Зростаюча складність сучасних обчислювальних систем підвищує вразливість до нових кіберзагроз. Останні досягнення в галузі безпеки комп'ютерної архітектури використовують регістри лічильників продуктивності апаратного забезпечення (HPC) для моніторингу поведінки програм та доступу до низькорівневих функцій. Інтеграція методів машинного навчання (ML) постає як перспективне рішення, що долає обмеження продуктивності традиційних програмних засобів захисту. Спеціалізовані регістри HPC реєструють різноманітні події, пов'язані з обладнанням, демонструючи ефективність

виявлення шкідливої діяльності за допомогою застосування алгоритмів ML. Це дослідження представляє комплексний та порівняльний аналіз останніх досягнень у новій галузі інтелектуального апаратного виявлення шкідливого програмного забезпечення, теми, яка привернула значну увагу дослідницького співтовариства протягом останнього десятиліття. Крім того, воно окреслює поточні проблеми та прогнозує майбутні тенденції досліджень, пропонуючи розуміння ефективних контрзаходів безпеки на основі лічильників продуктивності апаратного забезпечення.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Avast Free Antivirus 2025

Даний застосунок поширюється безкоштовно й містить велику вірусну базу, завдяки чому він настільки популярний не тільки серед користувачів України, але й в усьому світі. Кількість нових безкоштовних антивірусів постійно росте, деякі з них виходять настільки добротними, що перевершують своїх платних аналогів. В Avast убудовані модулі, які не передбачені навіть у деяких платних програмах.

Основні можливості додатка:

– Захист комп'ютера від вірусів.

– Безпечна робота в мережі Інтернет.

– Захист проведення фінансових і банківських операцій.

– Перевірка й контроль безпеки мережі Wi-Fi.

– Пошук непотрібних плагинів і розширень у веб-браузері.

І це далеко не всі функції Avast Free Antivirus 2025, що по праву вважається одним із кращих антивірусних додатків навіть серед платних аналогів. Як і антивірус Касперського, дане застосунок сумісний з більшістю сучасних ОС, у тому числі й мобільних.

360 Total Security

Розроблювачем даного додатка є компанія Qihoo з Китаю. Він містить кілька потужних антивірусних движків, швидкий у роботі й постійно нарощує свій арсенал функцій з виходом нових версій.

Розроблювачі розширили функціонала антивірусу настільки, що він здатний виконувати наступні завдання:

– Захищати веб-браузер від вірусних атак.

– Стискати дані в постійній пам'яті комп'ютера.

– Виконувати перевірку й очищення системного реєстру.

– Захищати мережу Wi-Fi від зовнішніх погроз.

Ще однією перевагою цієї програми є її безкоштовність. Вона, як і описані вище конкуренти, сумісна з основними операційними системами. Даний антивірус підійде тим користувачам, які шукають багатофункціональний і швидкий застосунок з безкоштовною ліцензією.

NANO Антивірус

Даний застосунок робить компанія, утворена в 2009 році. У ньому використані новітні технології й наробітки компанії, завдяки яким антивірус працює швидко, ефективно й має більше число різних функцій. На відміну від більшості інших антивірусів, які просто видаляють заражені файли, NANO намагається їх вилікувати завдяки використанню технології глибокої модуляції.

Основні особливості даного антивірусу:

– Швидка перевірка самих уразливих областей на наявність погроз.

– Хмарна технологія захисту.

– Прямий доступ до серверів для відновлення вірусної бази.

– Захист від будь-якого виду шкідливих додатків.

Застосунок має безкоштовну версію із трохи обмеженими можливостями й платну – PRO (вартість 1000 днів захисту за 1000 рублів). Недоліком програми є її високі вимоги до системи, тому вона здатне сповільнювати роботу щодо слабких машин.

Avira Antivirus Pro

Даний застосунок очолив рейтинг антивірусів 2025 року для Windows 10 за даними фахівців, і зараз спробуємо розібратися, чому. Застосунок має платну й

безкоштовну версії, завантажити які можна на офіційному сайті. Воно має гарні механізми захисту й досить швидко працює.

Переваги Avira полягають у наступному:

– Установлено модуль для захисту від погроз, які приходять із мережі.

– Є убудований фаєрвол.

– Дозволяє налаштовувати механізм обробки даних, установлюючи фільтри для деяких додатків і контенту

– Може бути використаний у якості LiveCD для очищення ПК, на якому немає антивірусної програми, від вірусів.

– Містить модулі для поліпшення продуктивності роботи ОС Windows.

– Містить ігровий режим, при включенні якого навантаження на ресурси системи різко падає.

Єдиним недоліком додатка є той факт, що воно поширюється платно. Безкоштовна версія має обмежені можливості й уже не здається таким привабливим варіантом для комп'ютера, що працює на Windows.

AVG Internet Security 2025

Дана версія антивірусу від компанії AVG значно перевершує застосунок минулого року. Програма платна, але досить придбати одну копію й установлювати її на необмежену кількість пристроїв. Антивірус підтримує такі операційні системи, як Android, Mac і Windows.

Він має наступні можливості:

– Виконує захист особистої інформації.

– Контролює роботу в мережі інтернет, запобігаючи вірусним атакам.

– Дозволяє здійснювати контроль над пристроями через інтернет-з'єднання, які захищені даним антивірусом.

Застосунок працює досить швидко, але при роботі з іншими антивірусними застосунками можуть виникати конфлікти (наприклад, із захисником Windows). Ще один недолік додатка – використання істотної кількості ресурсів системи.

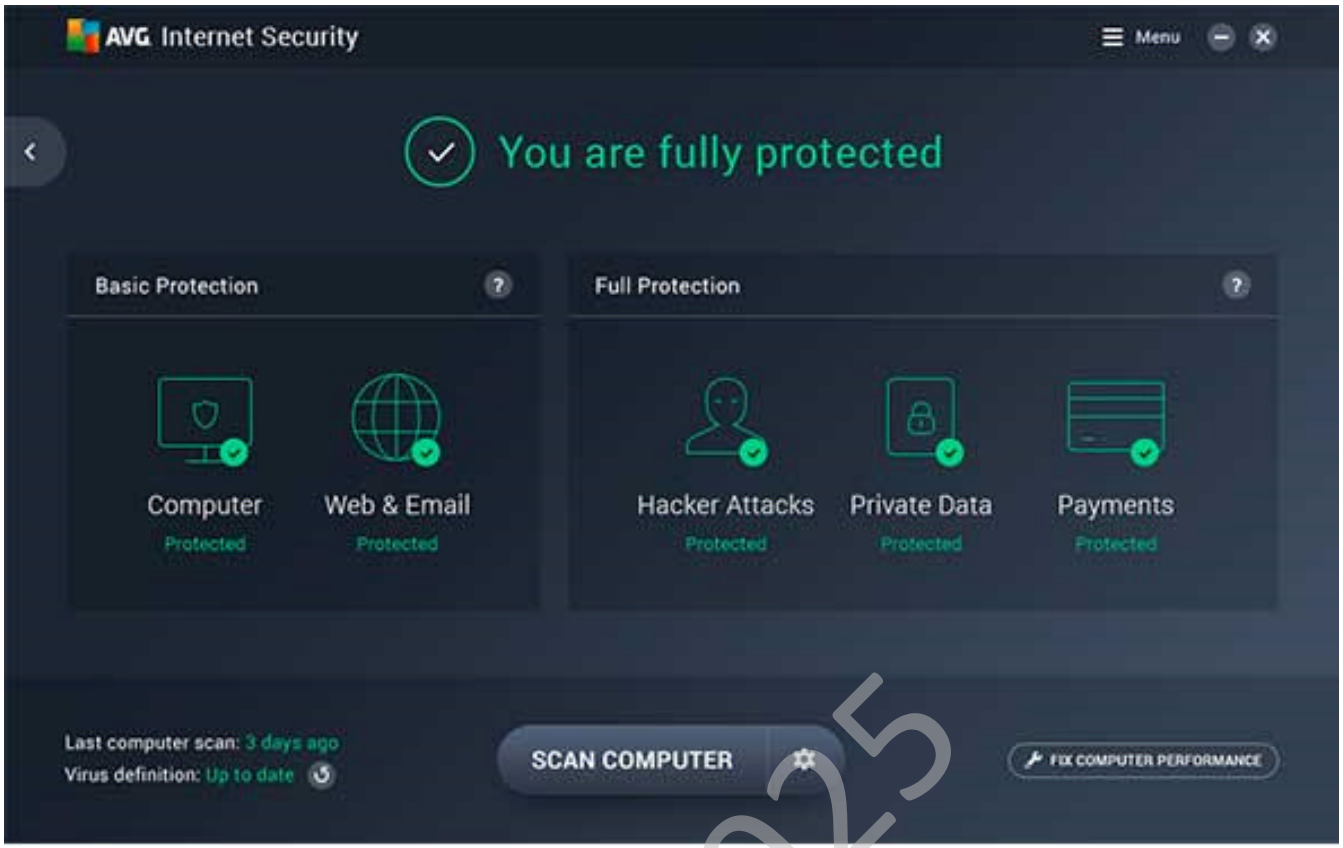

Рисунок 2.1 – Скріншот AVG Internet Security 2025

Panda Global Protection 2025

Даний застосунок також заслуговує на увагу й місце в списку кращих антивірусів 2025 для Windows 10 завдяки розмаїтості надаваних їм функцій, швидкості роботи й використанню системних ресурсів. Крім версії для Windows, існують ще Panda Global Protection 2025 для Android і Mac OS. Цей антивірус є платним.

Застосунок має наступні функції:

– Захист комп'ютера в реальному часі від вірусних атак.

– Захист мережі Wi-Fi.

– Батьківський контроль для обмеження перегляду дитиною матеріалів для дорослих.

Єдиний недолік цього антивірусу – його вартість. Але якщо вибирати серед платних додатків, Panda Global Protection є одним з антивірусів, що рекомендуються, поряд з Kaspersky і Norton, що буде розглянутий нижче.

Norton Security

Компанія Norton тривалий час займається розробкою ПЗ для захисту пристроїв від вірусних атак. Її наробітку лягли в основу й нової версії антивірусу, що забезпечує найвищий рівень захисту пристроїв. Він підтримується будь-який сучасної ОС, здійснює захист комп'ютерів у реальному часі. Застосунок є платним.

Антивірус захищає дані банківських карт і паролі користувача від їхнього розкрадання зловмисниками, перевіряє отримані дані з мережі на наявність вірусів, сканує що підключаються флешки та інші знімні накопичувачі.

Виробники даного додатка настільки упевнений в ефективності його роботи, що пропонують покупцям досить цікаві умови для його використання. Якщо антивірусу не вдалося автоматично видалити вірус на комп'ютері, це спробує зробити один з фахівців Norton. Якщо і йому це не вдасться зробити, тоді користувач одержує назад гроші, витрачені на покупку ліцензії!

BitDefender Antivirus

Даний антивірус використовує «движок», розроблений фахівцями компанії BitDefender, і здатний захистити пристрій від вірусів, які перебувають у його базі, і від невідомих шкідливих додатків. Захист виробляється в реальному часі. Антивірус запобігає завантаженню шахрайських і фішінгових сайтів, від погроз із мережі й вірусів, які можуть потрапити на ПК із флеш-накопичувача або переносного вінчестера. Користувачеві доступно величезна кількість налаштувань, але при цьому інтерфейс приємний і зручний. Відразу після установки програма починає свою роботу в оптимальному режимі, здійснюючи якісний захист пристрою від шкідливих програм.

Передбачено дві версії програми: безкоштовна й платна. Перша має обмеженого функціонала. Наприклад, у ній не можна внести налаштування рівня безпеки, чого буде не вистачати досвідченим користувачам.

Comodo Antivirus

Ця програма також потрапила в список кращих антивірусів 2025, рейтинг яких я надав у цьому огляді. Вона забезпечує базовий захист пристроїв, розпізнає й нейтралізує велику кількість різних вірусів і шкідливих додатків. Одна з основних переваг антивірусу Comodo – невисокі вимоги до ресурсів системи. Програма містить модуль, що дозволяє запобігти запуску потенційно небезпечного файлу. Якщо антивірусом був виявлений такий файл, користувач побачить відповідне повідомлення про погрозу на екрані. Поповнення бази вірусів виробляється швидко завдяки підтримці хмарної технології.

Рисунок 2.2 – Скріншот Comodo Antivirus

Даний антивірус поширюється безкоштовно й містить деякі недоліки:

– Відсутність функції батьківського контролю.

– Погана служба підтримки. При обігу користувач не може розраховувати на швидке рішення виниклої проблеми.

Як вибрати антивірус – поради

Кожний виробник антивірусу намагається виставити в кращому світлі свій застосунок, розписує його унікальні можливості й функції. Звичайному користувачеві може виявитися не просто зробити вибір на користь того або іншого програмного продукту. Один із критеріїв, на який варто звертати увагу при виборі програми, – здійснення захисту від вірусів, які «невідомі» антивірусу. Раніше такі програми могли лише впоратися з тими шкідливими застосунками, відомості про які перебували в їхній вірусній базі. Алі з виходом нового вірусу розроблювачам не завжди вдавалося оперативно випускати відновлення. Тому наявність системи «розумного сканування» – істотна перевага для антивірусної програми.

Також варто звертати увагу на системні вимоги додатка, наявність модуля «батьківський контроль», вартість ліцензії та інші фактори. Якщо немає бажання витрачати гроші на покупку ліцензії, можна звернути увагу на кращі безкоштовні антивіруси 2025 року, які я також розглянув у цьому огляді. Безкоштовні програми найчастіше не уступають по функціоналі платним.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Як мова програмування обрана Python. Python – високорівнева мова програмування загального призначення з акцентом на продуктивність розроблювача й читаність коду. Синтаксис ядра Python мінімалістичний. У той же час стандартна бібліотека включає великий обсяг корисних функцій.

Python підтримує кілька парадигм програмування, у тому числі структурне, об'єктно-орієнтоване, функціональне, імперативне й аспектно-орієнтоване. Основні архітектурні риси – динамічна типізація, автоматичне керування пам'яттю, повна інтроспекція, механізм обробки виключень, підтримка багатопоточні обчислень і зручні високорівневі структури даних. Код у Python організовується у функції й класи, які можуть поєднуватися в модулі (які у свою чергу можуть бути об'єднані в пакети).

Еталонною реалізацією Python є інтерпретатор CPython, що підтримує більшість активно використовуваних платформ. Він поширюється вільно під дуже ліберальною ліцензією, що дозволяє використовувати його без обмежень у будь-яких застосунках, включаючи пропрієтарні. Є реалізації інтерпретаторів для JVM (з можливістю компіляції), MSIL (з можливістю компіляції), LLVM і інших. Проект PyPy пропонує реалізацію Python на самому Python, що зменшує витрати на зміни мови й постановку експериментів над новими можливостями.

Python – мова програмування, що активно розвивається, нові версії (з додаванням/зміною язикових властивостей) виходять приблизно раз у два з половиною року. Внаслідок цього й деяких інших причин на Python відсутні ANSI, ISO або інші офіційні стандарти, їхня роль виконує CPython.

Python портований і працює майже на всіх відомих платформах – від КПК до мейнфреймів. Існують порти під Microsoft Windows, практично всі варіанти UNIX (включаючи FreeBSD і Linux), Plan 9, Mac OS і Mac OS X, iPhone OS 2.0 і вище, Palm OS, OS/2, Amiga, AS/400 і навіть OS/390, Symbian і Android.

При цьому, на відміну від багатьох портуємих систем, для всіх основних платформ Python має підтримку характерних для даної платформи технологій (наприклад, Microsoft COM/DCOM). Більше того, існує спеціальна версія Python для віртуальної машини Java – Jython, що дозволяє інтерпретаторові виконуватися на будь-якій системі, що підтримує Java, при цьому класи Java можуть безпосередньо використовуватися з Python й навіть бути написаними на

Python. Також кілька проектів забезпечують інтеграцію із платформою Microsoft .NET, основні з яких – IronPython і Python.Net.

Python підтримує динамічну типізацію, тобто тип змінної визначається тільки під час виконання. Тому замість «присвоювання значення змінної» краще говорити про «зв'язування значення з деяким ім'ям». У Python є убудовані типи: бульові, рядки, Unicode-рядки, цілі числа довільної точності, числа із плаваючої коми, комплексні числа й деякі інші. З колекцій Python підтримує кортежі (*tuples*), списки, словники (асоціативні масиви) і, починаючи з версії 2.4, безлічі. Всі значення в Python є об'єктами, у тому числі функції, методи, модулі, класи.

Додати новий тип можна або написавши клас (class), або визначивши новий тип у модулі розширення (наприклад, написаному мовою C). Система класів підтримує спадкування (одиночне й множинне) і метапрограмування. Можливе спадкування від більшості убудованих типів і типів розширень.

Всі об'єкти діляться на посилальні й атомарні. До атомарного ставляться int, long, complex і деякі інші. При присвоюванні атомарних об'єктів копіюється їхнє значення, у той час як для посилальних копіюється тільки покажчик на об'єкт, таким чином, обидві змінні після присвоювання використовують те саме значення. Посилальні об'єкти бувають змінювані й незмінні. Наприклад, рядки й кортежі є незмінними, а списки, словники й багато інших об'єктів – змінюваними. Кортеж у Python є, по суті, незмінним списком. У багатьох випадках кортежі працюють швидше списків, тому якщо ви не плануєте змінювати послідовність, то краще використовувати саме їх.

Мова має чіткий і послідовний синтаксис, продуману модульність й масштабованість, завдяки чому вихідний код написаних на Python програм легко читаємий.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускну кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи інтелектуального моніторингу та запобігання активності додатків.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформувати висновки про виконаний обсяг робіт та одержані результати.

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Розглянемо реалізацію системи моніторингу й запобігання активності додатків, що була реалізована у вигляді антивірусу.

Автономний антивірус включає систему запобігання вторгнень HIPS. У тесті протидії 30 експлойтам, програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, виявив і заблокував більше половини з них, що краще результатів Bitdefender і Kaspersky. Norton заблокував дві-третини атак, причому все з них на мережному рівні.

Складна система контролю пристроїв більше підходить для корпоративних середовищ, чим для звичайних споживачів, хоча сам продукт більше орієнтований на домашніх користувачів. Технічно підковані люди можуть запобігти підключенню зовнішніх пристроїв, включаючи кард-рідери, Bluetooth і зовнішні USB-пристрої. Для довірених пристроїв можна створювати виключення.

На сторінці “Сервіс” користувачеві доступні файли журналу подій і список доданих у карантин файлів. Інші інструменти призначені для агентів технічної підтримки при віддаленому усуненні проблем. Серед таких утиліт – запущені процеси, графік активності файлової системи й інструмент для створення знімків стану системи для наступного порівняння.

Базовий фаєрвол

Під час тестування фаєрвол коректно перевів всі системні порти в схований режим і успішно протистояв веб-атакам. Проте, програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, не впорався з тестом, у якому використовувалися запити команди ping – це означає, що кіберзлочинець зможе з'ясувати реальну IP-адресу комп'ютера.

Ще однією перевіркою роботи двостороннього фаєрвола є випробування на блокування спроб зловживання мережними підключеннями. Програмний контроль програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, може працювати в декількох режимах. Стандартний автоматичний режим дозволяє весь вихідний трафік і блокує підозрілий вхідний трафік.

При перемиканні в інтерактивний режим, програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, поводиться як традиційний фаєрвол, тобто при кожній спробі доступу в мережу з боку невідомої програми, продукт виводить користувальницький запит подальшої дії. Проте. функція контролю має кілька розширених опцій. Фаєрвол може застосувати ваш вибір один раз, створити на його основі постійне правило або запам'ятати вибір до завершення роботи програми. За замовчуванням, вибір не запам'ятовується, тобто користувачеві прийде щораз реагувати на запити.

Після натискання по посиланню докладної інформації, користувачеві показується інформація про видавця, репутацію файлу й віддаленому комп'ютері й портах. При виборі розширених налаштувань можна редагувати правила фаєрвола за допомогою окремих IP-адрес і номерів портів. Звичайним користувачам ці можливості не придадуться.

Багато продуктів, зокрема Norton і Kaspersky, приймають рішення програмного контролю самостійно й не покладаються на недосвідчених користувачів. ZoneAlarm Extreme Security 2017 для обробки додатків використовує масивну базу даних відомих надійних файлів і автоматично налаштовує програмний контроль. В інтерактивному режимі фаєрволу програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, потрібно вручну визначати правила для всіх програм і навіть для компонентів Windows – не найкращий підхід.

Один зі способів уникнути нескінченного потоку запитів – використовувати режим навчання. У даному режимі програмний продукт

моніторингу й запобігання активності додатків, що був розроблений у даній роботі, виконує моніторинг всіх програм, які одержують доступ у мережу й створює правила дозволу доступу. Режим навчання автоматично завершиться через два тижні, хоча даний період можна змінювати. Після цього запитів буде помітно менше. Також можна розглянути вибір режиму на основі політик, що блокує всі підключення за винятком тих, які дозволені правилами фаєрвола.

Мережний захист у програмному продукті моніторингу й запобігання активності додатків, що був розроблений у даній роботі, NOD32 Internet Security є розширеною в порівнянні з можливостями автономного антивірусу. Проте, у тесті експлойтів, обидва продукти показали однакові результати.

По вкрай мері, фаєрвол програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, успішно пручався прямим таргетованим атакам. Комплексний антивірус має два видимих процеси й одну службу, але в процесі тестування знайти спосіб для їхнього відключення за допомогою шкідливих технік не вдалося.

Фаєрвол програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, відверто розчарував. Він не зміг захистити від витоку реальної IP-адреси, а програмний контроль не повністю справляється зі своїм завданням. В інтерактивному режимі програма завалює користувача запитами. Це традиційний фаєрвол на базі застарілих технологій. Багато сучасних продуктів пропонують більше передові рішення.

Захист домашньої мережі

При виборі панелі “Захист домашньої мережі” у головному вікні антивірусу з'являється зображення карти мережі. Виявлені пристрої відображаються у вигляді іконок у концентричних колах. При цьому маршрутизатор і локальний пристрій показуються в самому центрі. Наступне коло показує недавно підключені пристрої, а саме далеке коло – пристрою, підключені за останній місяць.

Якщо програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, не може одержати ім'я, то відображає IP-адреса. Майстер мережі дозволяє з'ясувати, що ховається за IP-адресу й допомагає додати зрозумілу назву. З режиму детального перегляду можна вибрати посилання усунення проблем, щоб подивитися заблокований фаєрволом трафік з даного пристрою.

Натисніть кнопку “Сканувати маршрутизатор”, щоб програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, перевірив налаштування безпеки вашого роутера. Буде запущено кілька тестів на проникнення, націлених на роутер. При виявленні проблем, програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, відразу ж запропонує їх усунути.

Захист банківських платежів

Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, приділяє особливу увагу даної функції, тому що одна із трьох панелей на головному екрані присвячена захисту банкінгу. При виборі панелі відкривається захищена версія вашого браузера за замовчуванням із зеленою рамкою й баннером “Захищене” на верхній панелі. Браузер відкриває сторінку, що пояснює мета даної функції й рекомендує використовувати її тільки для інтернет-банкінгу й проведення фінансові транзакції, а не для звичайного серфінгу. Після установки антивірусу потрібно виконувати перезавантаження комп'ютера, щоб захист банківської оплати запрацював. Функція підтримує роботу з Chrome, Firefox і Internet Explorer. Користувачі Opera, Vivaldi і інших браузерів не зможуть їй скористатися.

Також, як і функція "Безпечні платежі" у продуктах "Лабораторії Касперського", захист банківських платежів автоматично активується при відвідуванні відомого фінансового сайту у звичайному, незахищеному браузері. Програмний продукт моніторингу й запобігання активності додатків, що був

розроблений у даній роботі, запропонує запустити безпечний браузер і запитає, потрібно чи запам'ятати цей вибір на постійній основі.

Обмежений батьківський контроль

Доступ до системи батьківського контролю програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, схований глибоко в налаштуваннях. У лівому навігаційному меню потрібно вибрати пункт “Налаштування”, а потім потрібно перейти в розділ “Засоби безпеки”. Після включення ви побачите список облікових записів Windows. Щоб завершити конфігурацію, потрібно вказати, які аккаунти належать дорослим, а які – дітям.

Залежно від віку дитини, програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, визначає, які із трьох десятків категорій умісту будуть піддаватися блокуванню. При детальному розгляді правил можна побачити, що кожна категорія має вікові обмеження – для всіх, 12+, 18+ або заборонене. Зверніть увагу, що навіть для облікових записів дорослих будуть блокуватися категорії Кримінал і Шкідливе ПЗ. Одночасно в області видимості перебуває всього три категорії, тому налаштування системи викликає серйозні труднощі.

При тестуванні контент-фільтр працював надійно. Фільтр працює в будь-якому браузері й не відключається простими мережними командами, які були успішні з деякими конкурентами. Під час випробування не вдалося виявити сайти, які уникли блокування. При блокуванні програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, показує в браузері просте попередження, схожі оповіщення відображаються при виявленні фішінг-погрози або шкідливого сайту. Дитина не зможе запросити перегляд заблокованого сайту, як це можна зробити в Symantec Norton Family Premier.

Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, обробляє захищені HTTPS сайти іншим образом. Він не може підмінити заблокований ресурс інформаційною сторінкою, тому в

браузері з'являється повідомлення про помилку, а спливаюче повідомлення пояснює, що відбулося.

В основному додатку програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, батьки можуть переглядати список заблокованих сайтів із вказівкою дати й часу спроби відвідування, обліковому запису й категорії контенту.

На цьому можливості батьківського контролю закінчуються. Тут ви не знайдете функцію керування контактами в мессенджерах, аналітики соціальних мереж або контролю відеоігор по їхньому рейтингу. Крім того, недоступне керування часом використання Інтернету й комп'ютера. Батьківський контроль сильно урізаний, але працює добре.

Точний антиспам

Автономний антивірус програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, має ефективний убудований захист електронної пошти. Компонент сканує вхідну пошту POP3 і IMAP і вихідну пошту SMTP на предмет шкідливих программ. Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, підтримує інтеграцію з Outlook, Outlook Express / Windows Mail або Windows Live Mail для розширеного контролю. При виявленні шкідливого ПЗ інструмент видаляє небезпечне вкладення й додає його назва до теми повідомлення. У комплексному антивірусі є ще повноцінний спам-фільтр.

Щоб включити антиспам, перейдіть у Налаштування > Захист Інтернету й виберіть іконку шестірні поруч із секцією “Захист від спаму”. Налаштувань не так багато. Користувач може включити розширений захист від спаму, що повинна поліпшити точність. Дана функція відключена за замовчуванням, тому що серйозно знижує швидкість обробки пошти. Можна поміняти стандартну мітку [SPAM] і включити оцінку спаму. За замовчуванням антиспам імпортує список контактів і адресатів ваших повідомлень у виключення. Також туди містяться листи, які були вручну визнані не спамом. Доступно ручне редагування

білого й чорного списку адресатів. У сучасних умовах спам-фільтр потрібний не всім, тому що багато поштових провайдерів уже мають дану функцію. Якщо вам усе ще потрібний окремий антиспам, то програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, прекрасно справляється із цим завданням.

Захист веб-камери

Повідомлення про випадки стеження за допомогою веб-камери з'являються із тривожною регулярністю. Звичайно, можна заклеїти камеру изолентой, але якщо ви часто берете участь у веб-конференціях, постійне заклеювання й відклеювання може стомити.

Захист веб-камери в програмному продукті моніторингу й запобігання активності додатків, що був розроблений у даній роботі, є доповненням програмного контролю, але більше простим у використанні. Користувач може заблокувати будь-які спроби доступ до камери й відключати захист тільки під час користування Skype. Якщо витратити небагато часу на налаштування, то можна вказати, які програми можуть одержувати доступ до камери й включити відображення оповіщення перед наданням доступу. Перевірити функцію в дії не вдалося, але для користувачів веб-камер вона може бути реально корисної.

Невеликий вплив на продуктивність

Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, показав середній вплив на ресурси системи. У тесті виміру часу завантаження системи, антивірус сповільнив час завантаження всього на 11 відсотків, що краще середнього показника.

Розширений фаєрвол

Фаєрвол програмного продукту моніторингу й запобігання активності додатків, що був розроблений у даній роботі, справляється з базовими завданнями – відбиття атак на порти й перемикання всіх системних портів у схований режим. За замовчуванням програмний контроль обмежується дозволом вихідного трафіка й блокуванням підозрілого вхідного трафіка. В інтерактивному

режимі при кожній спробі виходу в Інтернет з боку нової програми з'являється діалогове вікно із запитом подальшої дії. Фаєрволи в Kaspersky і Norton уміє автоматично обробляти дозволу для відомих надійних і шкідливих програм і пильно стежать за невідомими застосунками. Такий підхід є більше передовим, чим просто покласти ці обов'язки на плечі користувача. З позитивної сторони, не вдалося знайти спосіб для програмного відключення захисту ESET.

При запуску інструмента “Захист домашньої мережі” з'являється пророблена карта домашньої мережі з усіма підключеними пристроями. Ваш пристрій і роутер показуються в центрі, а інші пристрої розташовуються в концентричних колах залежно від залежності від дати підключення. При виборі кнопки сканування маршрутизатора програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, перевірить налаштування безпеки роутера й запропонує виправлення потенційних проблем.

Також як "Безпечні платежі" в Kaspersky, захист банківських операцій у програмному продукті моніторингу й запобігання активності додатків, що був розроблений у даній роботі, автоматично вступає в дію при відвідуванні фінансового сайту. Продукт пропонує запустити сайт у захищеному браузері з можливістю запам'ятати цей вибір.

Система батьківського контролю налаштовується окремо для кожного облікового запису Windows. Залежно від віку дитини застосовуються профілі, які дозволяють блокувати деякі з більше 30 категорій умісту. Можна самостійно налаштовувати фільтрацію. Категорії Кримінал і Шкідливе ПЗ блокуються навіть для аккаунтів дорослих. Компонент добре відробив при тестуванні, але не пропонує розширених функцій, а займається лише фільтрацією контенту й реєстрацією активності.

Спам-фільтр обробляє поштові аккаунти POP3 і IMAP і підтримує інтеграцію з популярними поштовими клієнтами. При використанні підтримуваного поштовика потрібно налаштувати правило для автоматичної переадресації небажаної пошти в папку спаму. Ваші контакти автоматично

попадають у список виключень і ніколи не блокуються. Також доступний чорний список для джерел спаму.

Боїтеся стеження за допомогою веб-камери? Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, вам готовий допомогти. Ви можете відключити використання веб-камери, коли вона вам не потрібна й дозволити доступ для окремих программ. Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, може також попереджати при кожній спробі доступу до веб-камери.

Споконвічно здавалося, що додаткові функції повинні не кращим образом позначитися на продуктивності в порівнянні з комплексом початкового рівня. Однак, проведені випробування не виявили помітної різниці. Продукт небагато знижує швидкість завантаження системи й файлових ситуацій, але при реальному використання помітна з не відчувалася.

Потужний менеджер паролів

Менеджер паролів заснований на Sticky Password Premium, але одержав далеко не всі функції Sticky Password. Наприклад, користувач може синхронізувати паролі між декількома комп'ютерами й Mac, але не пропонує підтримку мобільних пристроїв. Також відсутня функція синхронізація паролів у бездротовій мережі Wi-Fi.

Спочатку потрібно створити майстер-пароль. Спливаюча підказка повідомляє, що майстер-пароль повинен складатися із заголовних, малих літер і цифр, а його довжина повинна бути не менш 8 знаків. Увести пароль можна за допомогою віртуальної клавіатури, щоб перестрахуватися від кейлоггерів. Під час моніторингу були перехоплені лише координати курсору, тому небажаний глядач ніколи не довідається, по яких клавішах ви кликали.

Можна налаштувати розблокування менеджера паролів при підключенні якого-небудь USB або Bluetooth пристрою. Проте, це не двофакторна автентифікація, тому що даний спосіб повністю заміняє введення майстр-пароля.

Менеджер паролів підтримують інтеграцію з Chrome, Firefox, Internet Explorer і значним списком менш популярних браузерів: Pale Moon, Comodo Dragon і SeaMonkey. Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, уміє імпортувати паролі із браузерів, але на відміну від оригінальної версії Sticky Password, не імпортує закладки. Також підтримується імпорт даних з RoboForm, KeePass, LastPass, Dashlane, Kaspersky Password Manager, і 1Password.

Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, захоплює облікові дані при авторизації на безпечних сайтах. Під час захвата користувач може задати зрозумілу назву для запису й визначити підходящу для неї категорію. Нову групу в цей момент створити не вийде на відміну від LastPass. Однак, в основному вікні менеджера паролів можна створити будь-яка кількість груп, включаючи вкладені групи. Ці категорії стануть пунктами й підпунктами основного меню списку паролів, що розкривається при натисканні по кнопці браузерного розширення.

При повторному відвідуванні програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, пропонує заповнити ваші облікові дані. Продукт упевнено обробляє події зміни паролів. Проте, при відвідуванні сайту з нестандартною формою авторизації, просто зберегти уведені дані не вийде, хоча дана можливість передбачена в LastPass і Sticky Password.

Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, також захоплює облікові дані при створенні нового аккаунта. Убудований генератор допомагає створити захищений пароль для облікового запису. За замовчуванням програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, генерує 15-значний пароль, що містить заголовні букви, малі літери й цифри. Для посиленого захисту можна додати спецсимволи.

Також як RoboForm, LastPass і деякі інші продукти, програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, уміє захоплювати й підставляти паролі додатків. Просто перетягнете курсор у вигляді хрестика у вікно уведення пароля, уведіть дані й збережете запис. Потім програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, сам подбає про автоматичний вхід.

Користувач може створювати кілька профілів – колекцій персональних даних для заповнення веб-форм. Для кожного профілю потрібно ввести персональну, контактну інформацію й ділові дані. При бажанні можна додати реквізити банківських рахунків або кредитних карт. При відвідуванні сторінки з веб-формою, програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, виводить червону границю навколо полів заповнення. Просто натисніть по кнопці панелі інструментів і виберіть бажаний профіль заповнення. Як і багато інших програм, програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, заповнює не всі форми, але в цілому справляється успішно. RoboForm Everywhere 7 спеціалізується на заповненні форм, підтримує більше типів полів, чим конкуренти й допускає кілька записів для одного типу поля.

Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, не пропонує наочної аналітики паролів, як у випадку з LastPass, Dashlane, або LogMeOnce Password Management. Замість цього він попереджає про паролі з низьким рівнем безпеки. Але не варто всерйоз покладатися на даний рейтинг, тому що 6-значний пароль, що використовує два набори символів буде вважатися нормальним. Слабкий пароль “Monkey” не з'явиться в списку попереджень.

У програмному продукті моніторингу й запобігання активності додатків, що був розроблений у даній роботі, відсутній ще кілька функцій оригінального Sticky Password. Тут недоступна онлайн консоль для керування пристроями. Наприклад, не можна заблокувати менеджер на загубленому пристрої або

заборонити додавання нових пристроїв. Функція захищених заміток дозволяє зберігати важливу інформацію, що буде синхронізуватися між пристроями. Портативну версію менеджера паролів теж не вдасться створити в ESET. Навіть без цих функцій, менеджер паролів програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, є потужних інструмент, що по своїх можливостях перевершує багато автономних рішень.

Безпечне шифрування даних

Існує велика кількість ризиків втрати даних. Деякі трояни крадуть приватні дані й відправляють їх кіберзлочинцям. Загублений ноутбук може привести до витоку даних. Нарешті, ваші документи можуть бути доступні цікавим членам родини або колегам по роботі. Кращий спосіб захистити важливі дані – використовувати захищене шифрування. Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, саме пропонує таку функцію.

Як і схожа функція в Bitdefender Internet Security 2017, Захист даних у програмному продукті моніторингу й запобігання активності додатків, що був розроблений у даній роботі, створює будь-яку кількість віртуальних зашифрованих дисків. Вам залишається ввести назву диска й розташування файлу, що буде містити дані віртуального диска. Ви можете вибрати доступні значення ємності диска: 500 Мб, 1 Гб, 5 Гб, 10 Гб, або 100 Гб або можете вказати довільний обсяг. Додайте пароль, і все готово до роботи.

Сторінка для уведення паролів нагадує, що у випадку втрати пароля, ви втратите доступ до ваших файлів, і навіть фахівці програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, не зможуть допомогти. Проте, доступна опція для автоматичної розшифровки диска для поточного облікового запису Windows. Для посиленої безпеки рекомендується відключити цю опцію, якщо ваш аккаунт Windows не захищений дуже сильним паролем.

При розблокуванні, віртуальний диск працює, як і будь-який інший локальний диск. Ви можете переміщати файли на диск і з диска, редагувати вміст і виконувати будь-які інші дії. На відміну від інших аналогічних компонентів в інших продуктах, програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, не має функції примусового блокування захищеного сховища. Це відбувається автоматично при виході з облікового запису або при перезавантаженні комп'ютера.

Немає ніякого сенсу в шифруванні файлів, якщо ви залишаєте незашифровані оригінали. Багато рішень для шифрування поставляються разом з файловими шредерами, які безпечно видаляють оригінальний об'єкт без можливості відновлення. Kaspersky уміє автоматично видаляти оригінали при шифруванні їхніх копій. На жаль, але в програмному продукті моніторингу й запобігання активності додатків, що був розроблений у даній роботі, інструмент безпечного очищення відсутній. Можна використовувати клавіатурне сполучення Shift+Del, у цьому випадку файл буде видалений минаючи кошик, але це не убезпечить від потенційного відновлення.

Створити віртуальне зашифроване сховище можна й на переносному USB-накопичувачі. Процес аналогічний, але не потрібно вказувати ім'я й обсяг. Нова папка з назвою Encrypted з'явиться на флешці, і програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, буде автоматично запитувати пароль при підключенні пристрою. Також можна налаштувати автоматичне розблокування для поточного облікового запису.

Антизлодій

Антизлодій – розповсюджена функція для мобільних антивірусів, у десктопних версіях захист від крадіжки зустрічається нечасто. Bitdefender є одним з деяких конкурентів, які можуть визначати місце розташування, блокувати й стирати дані на пристроях Windows. Функція Антизлодій у програмному продукті моніторингу й запобігання активності додатків, що був розроблений у даній роботі, не підтримує віддалене очищення, але вміє визначати

геолокацію пристрою, блокувати його й знімати зображення з веб-камери й скріншоти екрана.

Керування функціями захисту від крадіжки здійснюється на веб-порталі. Спочатку буде запропоновано пройти кроки оптимізації. Якщо на вашім ноутбуці або планшеті Windows настроєний автоматичний вхід без уведення пароля, то буде запитана активація входу по паролі. Крім того, буде створений “фантомний аккаунт”.

Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, кожних 10 мінут буде одержувати дані про статус пристрої. Якщо ви пометете пристрій як загублене, то програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, виконає комплекс мір: система буде перезавантажена й буде виконаний автоматичний вхід в “фантомний аккаунт”. Почнеться збір даних про геолокації й передача скріншотів екрана. При бажанні можна вивести повідомлення, наприклад, із проханням повернути пристрій по контактній адресі. Моніторинг буде тривати протягом 14 днів. Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, повідомить по електронній пошті про закінчення періоду моніторингу.

Під час тестування функція працювала максимально коректно. Після перезавантаження список реальних аккаунтів не відображався й був доступний тільки “фантомний”. Одержати доступ до реальних користувальницьких файлів не вдалося. Система дійсно була заблокована. Програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, кожних 10 хвилин захоплював зображення екрана, і актуальні скріншоти відразу ж з'являлися в онлайн консолі. Роботу функції зняття фотографій з веб-камери перевірити не вдалося через відсутність пристрою.

Для визначення місця розташування програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, аналізує сигнали бездротових мереж. У випадку підключення через Ethernet локація не

буде розпізнана. У довідковій системі повідомляється, що в цьому випадку можна перевірити список IP-адрес, до яких підключається пристрій і визначити зону пошуку. Під час тестування місця розташування пристрої було визначено некоректно – посередині ставка в 13 кілометрах від його реальної локації.

Захист від крадіжки – дійсно корисна функція для ноутбуків. Всі ноутбуки оснащуються модулями бездротового зв'язку, а значить ви зможете відстежити їхнє знаходження. Якщо ви встановили програмний продукт моніторингу й запобігання активності додатків, що був розроблений у даній роботі, на ноутбук, не забудьте пройти кроки оптимізації, просто на випадок, якщо вам знадобляться функції Антизлодія.

3.2 Розробка структурної схеми

Кібербезпека стала першочерговою проблемою в комерційних системах введення даних, чому зловмисники все частіше потребують програмне забезпечення-вразливості програмного та апаратного забезпечення для компрометації інформаційно-технологічної інфраструктури. Нещодавні досягнення свідчать про сплеск використання нових вразливостей для зловмисної діяльності, підкреслюючи критичну потребу в надійному захисті від зловмиснихзагроз програмного забезпечення

Шкідливе програмне забезпечення – широкий термін, що охоплює шкідливе програмне забезпечення – це фрагмент коду, розроблений кіберзлочинцями для проникнення в комп’ютерні системи без згоди користувача, що є несанкціонованим. доступ до даних, знищення файлів та інші шкідливі дії [1], [2]. Зростання інформаційних технологій посилило серйозність шкідливого програмного забезпечення як основну загрозу.

Традиційні методики визначення на основі програмного забезпечення, які базуються на аналізі сигнатур стикаються з такими недоліками продуктивності,

як обмеження статичного аналізу, неможливість виявлення обфускованих та важких обчислювальних витрат, особливо на системи з обмеженими ресурсами.

Ці проблеми сприяють через обмеження обчислювальної потужності та пропускної спроможності зв'язку середовища інтегрованих систем [3], [4], [5]. Для вирішення цих викликів вкрай необхідно розвивати ефективні та економічно вигідні контрзаходи кібербезпеки, зосереджуючись на захисті інформаційних користувачів та пом'якшення нових кіберзагроз [6].

Це забезпечує парадигму переходу до інтеграції заходів безпеки в базове апаратне забезпечення, встановлюючи підхід «знизу вгору» для зміцнення комп'ютерного використання пристроїв, а не ставлення до безпеки як до другорядної [7]. У системах, дали успішних програм у різних галузях, особливо в підвищенні системи безпеки [8], [9], [10].

Нещодавні дослідження підкреслюють важливість визначення шкідливих дій на рівнях апаратного забезпечення та архітектури процесорів через його швидкість, ефективність та меншу видимість для наявних експлуатацій зловмисника.

З цим призначено апаратно-допоміжне шкідливе програмне забезпечення Методи виявлення (HMD), зокрема, з використанням машинного навчання з використанням функції продуктивності апаратного лічильника (HPC), з'являється як рішення недоліків традиційного програмного виявлення шкідливих програм [11], [12], [13], [14], [15].

Високопродуктивні процесори (HPC) – це спеціалізовані регістри в межах системи продуктивності. Блок моніторингу (PMU), вбудований в сучасні мікропроцесори, відстежують події апаратного забезпечення програми (наприклад, кількість виконаних циклів, інструкцій, пов'язаних з ними промахів кешу тощо)[16], [17], [18], [19].

Крім того, технології машинного навчання продемонстрували ефективність виявлення та класифікації аномалій у межах простору низького рівня оцінки. Використовуючи машинне навчання, системи можуть ефективно

розпізнавати потенціал загрози та проактивно реагувати на зміни в поведінці в реальний час [7], [20].

Поточні дослідження в галузі інтелектуального виявлення шкідливих програм на Рівень апаратного забезпечення охоплюють різні обчислювальні платформи, такі як вбудовані системи, Інтернет речей і високонавантажені системи продуктивності.

Переважно використовуємо найсучасніші дослідження HMD-особливо наголошуємо на розробці та розробці стандартів передових методів машинного навчання для протидії еволюції загрози шкідливого програмного забезпечення.

У даній роботі наведено поглиблений аналіз апаратно-допоміжних методи виявлення шкідливого програмного забезпечення, зосереджені на останніх досягненнях у використанні штучного інтелекту та машинного навчання для покращення захисту систем від шкідливих нападів.

Зростання кількості та складності інфекцій шкідливим програмним забезпеченням впливає на окремих осіб та організації. Ці шкідливі програми з різноманітними функціональними можливостями розроблені для шкідливих цілей, таких як дистанційне керування, крадіжка даних, несанкціонований доступ, знищення файлів та проведення атак типу «відмова в обслуговуванні» [1], [10].

Наведемо огляд процесу підходів на основі машинного навчання, розроблених для підвищення кібербезпеки, зокрема в контексті апаратного виявлення шкідливого програмного забезпечення. Цей процес охоплює етапи від моніторингу програм для профілювання даних високопродуктивних обчислень, розробки функцій та навчання детекторів на основі машинного навчання та онлайн-виводу. Безперервне навчання моделей машинного навчання шляхом аналізу низькорівневих мікроархітектурних особливостей має на меті розпізнавання та протидію шкідливим шаблонам. Цей проактивний та інтелектуальний підхід захищає архітектуру процесора від потенційних загроз,

що охоплюють не лише шкідливе програмне забезпечення, але й атаки по мікроархітектурних бічних каналах [21], [22].

Розробка ефективних апаратних детекторів шкідливого програмного забезпечення на основі машинного навчання починається з таких ключових кроків, як збір даних та вибір функцій [7], [11], [14], [16]. У сучасних мікропроцесорах можна збирати численні мікроархітектурні події, але вибір відповідних низькорівневих функцій є важливим для уникнення обчислювальної складності та затримок, пов'язаних з високовимірними наборами даних. Зокрема, ідентифікація суттєвих низькорівневих мікроархітектурних функцій є критично важливою для апаратного виявлення шкідливого програмного забезпечення з кількох причин:

– Велика кількість мікроархітектурних подій (наприклад, 100+ в Intel Xeon) призводить до високовимірних даних [14].

– Обробка необроблених наборів даних передбачає обчислювальну складність та спричиняє затримки [23].

– Вибір відповідних мікроархітектурних подій створює труднощі у визначенні нетривіальних подій для різних класів шкідливих програм [15].

Ця проблема ускладнюється обмеженою доступністю регістрів HPC у різних процесорах, зазвичай від 2 до 8.

Проблема обмеженої кількості регістрів HPC, тісно пов'язана з виявленням шкідливого програмного забезпечення під час виконання, обговорює значну проблему HMD, розглянуту в нещодавніх роботах [14], [15]. Вона включає визначення мінімального набору HPC, які точно фіксують характеристики шкідливих атак, тим самим мінімізуючи непотрібні обчислювальні витрати. Це прагнення забезпечує розробку ефективного контрзаходу безпеки на основі машинного навчання з мінімальним впливом на продуктивність системи. Щодо обмежень архітектури базового процесора, особливо в обчислювальних платформах з обмеженими ресурсами, таких як вбудовані системи та пристрої Інтернету речей з обмеженими регістрами HPC,

ефективне, але точне виявлення під час виконання залежить від вибору критичних ознак. Нещодавні дослідження HMD, такі як [14], [15], [19], розглядали ефективне HMD під час виконання, визначаючи мінімальний набір основних подій лічильника продуктивності, необхідних для збору даних за один прогін.

Існує чотири основні кроки процесу розробки ознак:

– Очищення ознак включає аналіз необроблених даних для пошуку порожніх записів, викидів та будь-яких інших аномальних записів даних, щоб їх можна було видалити з процесу машинного навчання. Це також може забезпечити зворотний зв'язок для покращення збору даних.

– Нормалізація ознак є критичним кроком для масштабування табличних даних вздовж значень стовпців або рядків, запобігаючи домінуванню деяких даних або ознак з великими значеннями в процесі навчання. Цей метод є ефективним, особливо для алгоритмів машинного навчання, які чутливі до значень відстані між ознаками. Поширені методи нормалізації включають нормалізацію L1/L2 та нормалізацію MinMax.

– Вибір ознак включає аналіз важливості ознак, аналіз кореляції ознак та вибір найкращих ознак. Цей процес зазвичай виконується офлайн та ефективно тестується для цільової моделі машинного навчання.

– Вилучення ознак полягає у вилученні записів даних з найкращими ознаками для формулювання навчального набору даних. На етапі онлайн-виведення це означає вилучення онлайн-даних, які мають ті ж найкращі ознаки та розмірність, що й навчальний набір для обробки висновків за допомогою детекторів машинного навчання.

Вибрані ознаки високопродуктивних обчислень використовуються для навчання окремих детекторів на основі машинного навчання. Класифікатор прагне встановити кореляцію між значеннями ознак та поведінкою програми, прагнучи передбачити наявність шкідливих шаблонів (доброякісних або типу атаки). Кілька методів вибору ознак відіграли важливу роль у попередніх

зусиллях HMD на основі машинного навчання. До них належать такі методи, як оцінка атрибутів кореляції [14], [24], [25], [15], [26], [27], аналіз головних компонент [15], [26], [28], оцінка коефіцієнта підсилення [16], [19], [29] та оцінка Фішера [30], [12].

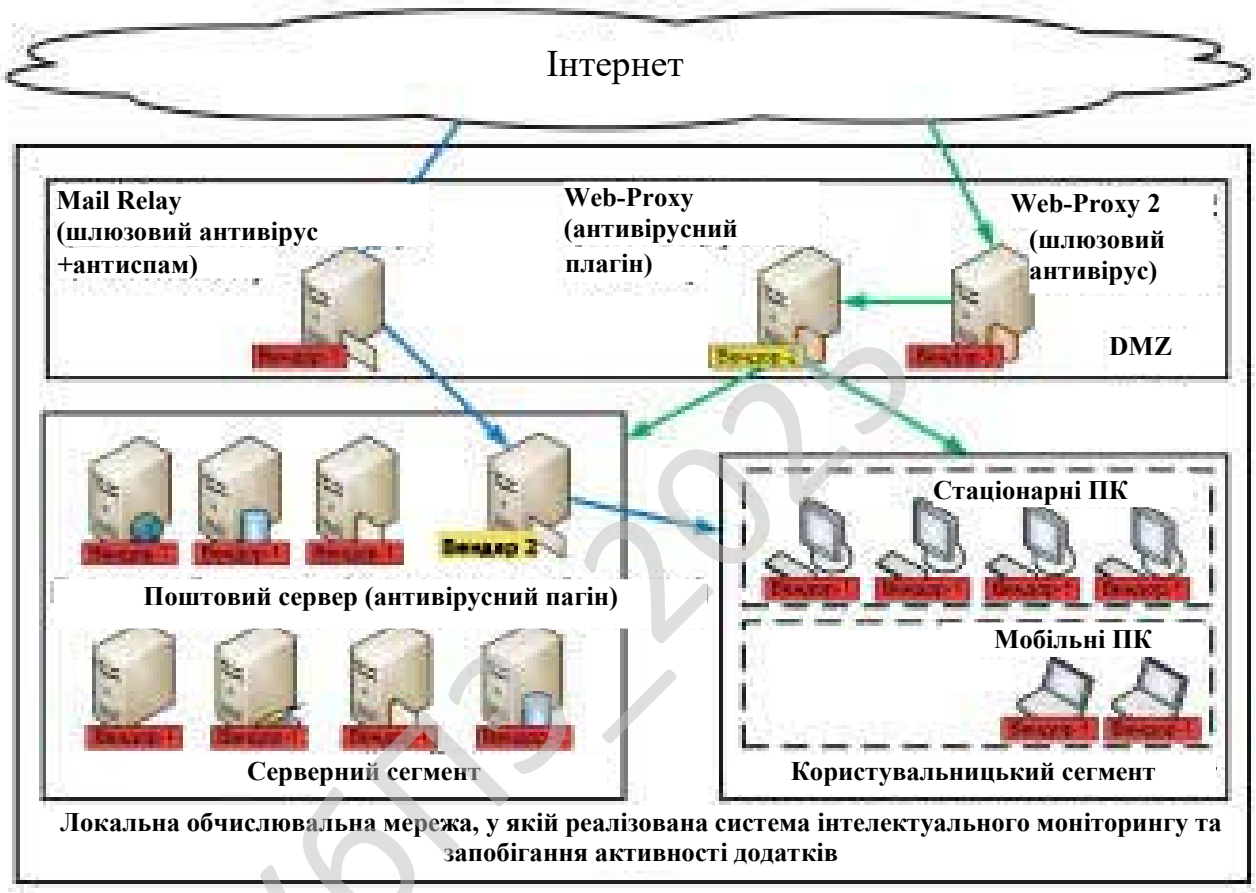

Рисунок 3.1 – Структурна схема системи

3.3 Розробка функціональної схеми

Система інтелектуального моніторингу та запобігання активності додатків, у вигляді антивірусного комплексу, що розроблена у результаті виконання магістерської роботи – захист вашого комп'ютера від шкідливих програм, що включає базові функції забезпечення безпеки вашого ПК. Антивірус

використовує новітні технології захисту, завдяки якому забезпечується безпека й стабільна робота комп'ютера.

Основні функції антивірусу, що розроблений:

– Захист у режимі реального часу.

– Базовий захист при роботі в мережі Інтернет і з електронною поштою.

– Мінімальне завантаження комп'ютера.

– Інтуїтивно зрозумілий інтерфейс.

– Для повноцінного захисту комп'ютера крім антивірусу рекомендується використовувати міжмережний екран.

– Перевірка файлів, веб-сторінок, поштових і ICQ-повідомлень.

– Блокування посилань на заражені веб-сайти й сайти, що перехоплюють інформацію.

– Проактивний захист від невідомих погроз, заснована на аналізі поводження програм.

– Самозахист антивірусу, що розроблений попереджає погрозу вимикання з боку шкідливого ПЗ.

– Система миттєвого виявлення погроз, що моментально блокує нові шкідливі коди.

– Реалізовано модуль «Перевірка посилань», що попереджає про заражені або небезпечних веб-сайти.

– Проактивний захист нового покоління від невідомих погроз.

– Віртуальна клавіатура для безпечного введення логінів, паролів і номерів кредитних карт на веб-сторінках.

– Перевірка операційної системи й установлених програм на наявність уразливостей.

Рисунок 3.2 – Функціональна схема системи

– Налаштування операційної системи й інтернет-браузера для безпечної роботи в мережі Інтернет.

– Відновлення працездатності системи після вірусної атаки.

– Видалення тимчасових файлів інтернет-браузера.

На рисунку 3.2 зображена функціональна схема системи. Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма процесів розробленої системи зображена на рисунку 3.3. При детальному її розгляді можна побачити як саме проходе взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування). Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи. Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

Діаграми потоків даних містять чотири типи елементів:

– Процеси які являють собою трансформацію даних в рамках описуваної системи.

– Сховища даних (репозиторії).

– Зовнішні по відношенню до системи сутності.

– Потоки даних між елементами трьох попередніх типів.

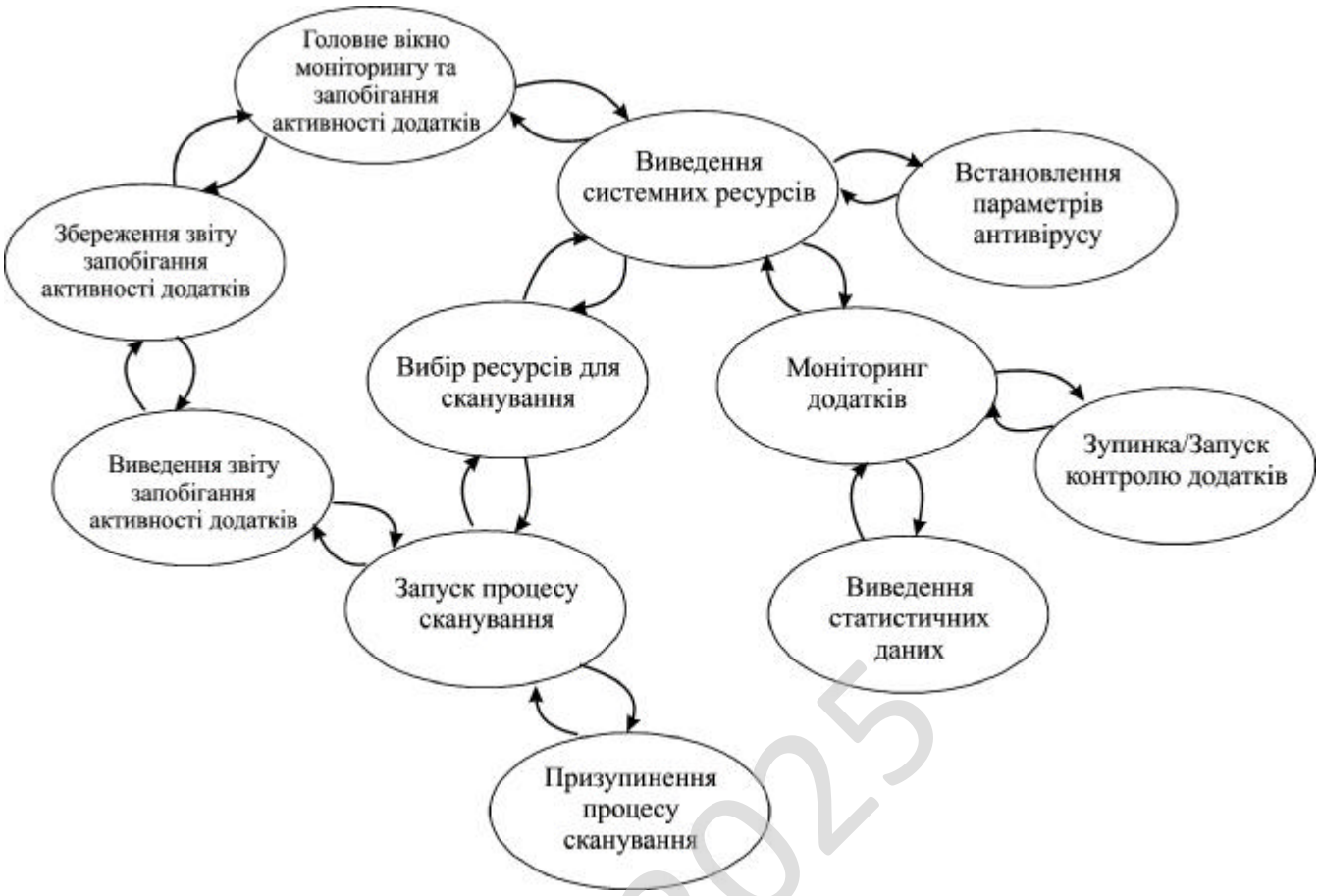

Рисунок 3.3 – Діаграма взаємодії процесів

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Під час роботи над магістерською роботою було створено блок-схеми. Перед їх розглядом необхідно провести роз’яснення який саме тип блок-схем використовується.

Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо. Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму.

У інформаційних технологіях функціональна схема складається з функціональних блоків, які являють собою конструктивно відособлені частини (елементи або пристрої) автоматичних систем, які виконують певні функції. Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються. Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

Функціональні схеми можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними.

У другому варіанті схема відображається більш детально, що полегшує її читання та ілюструє принцип роботи.

Основні елементи схем алгоритму це термінатор, процес, рішення, зумовлений процес (підпрограма), дані та з'єднувач.

Термінатор це елемент відображає вхід із зовнішнього середовища або вихід з неї (найчастіше застосування – початок і кінець програми). Всередині фігури записується відповідна дія.

Процес це виконання однієї або кількох операцій, обробка даних будь-якого виду (зміна значення даних, форми подання, розташування). Всередині фігури записують безпосередньо самі операції.

Рішення це показує рішення або функцію перемикального типу з одним входом і двома або більше альтернативними виходами, з яких тільки один може бути обраний після обчислення умов, визначених всередині цього елементу. Вхід в елемент позначається лінією, що входить зазвичай у верхню вершину елементу. Якщо виходів два чи три то зазвичай кожен вихід позначається лінією, що виходить з решти вершин (бічних і нижній). Якщо виходів більше трьох, то їх слід показувати однією лінією, що виходить з вершини (частіше нижній) елемента, яка потім розгалужується. Відповідні результати обчислень можуть записуватися поруч з лініями, що відображають ці шляхи.

Зумовлений процес (підпрограма) це символ відображає виконання процесу, що складається з однієї або кількох операцій, що визначені в іншому місці програми (у підпрограмі, модулі). Всередині символу записується назва процесу і передані в нього дані.

Дані це перетворення у форму, придатну для обробки (введення) або відображення результатів обробки (виведення). Цей символ не визначає носія даних (для вказівки типу носія даних використовуються специфічні символи).

З'єднувач це символ відображає вихід в частину схеми і вхід з іншої частини цієї схеми. Використовується для обриву лінії та продовження її в іншому місці (приклад: поділ блок-схеми, що не поміщається на листі). Відповідні сполучні символи повинні мати одне (при тому унікальне) позначення.

Блок-схеми є першоджерелами стратегії розвитку ПЗ. Тому від точності і детальної блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об’єктно-орієнтована що вимагає оптимізації програми високого рівня, також те, що при розробці програми слід надати особливу увагу модулю інтелектуального моніторингу та запобігання активності додатків.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

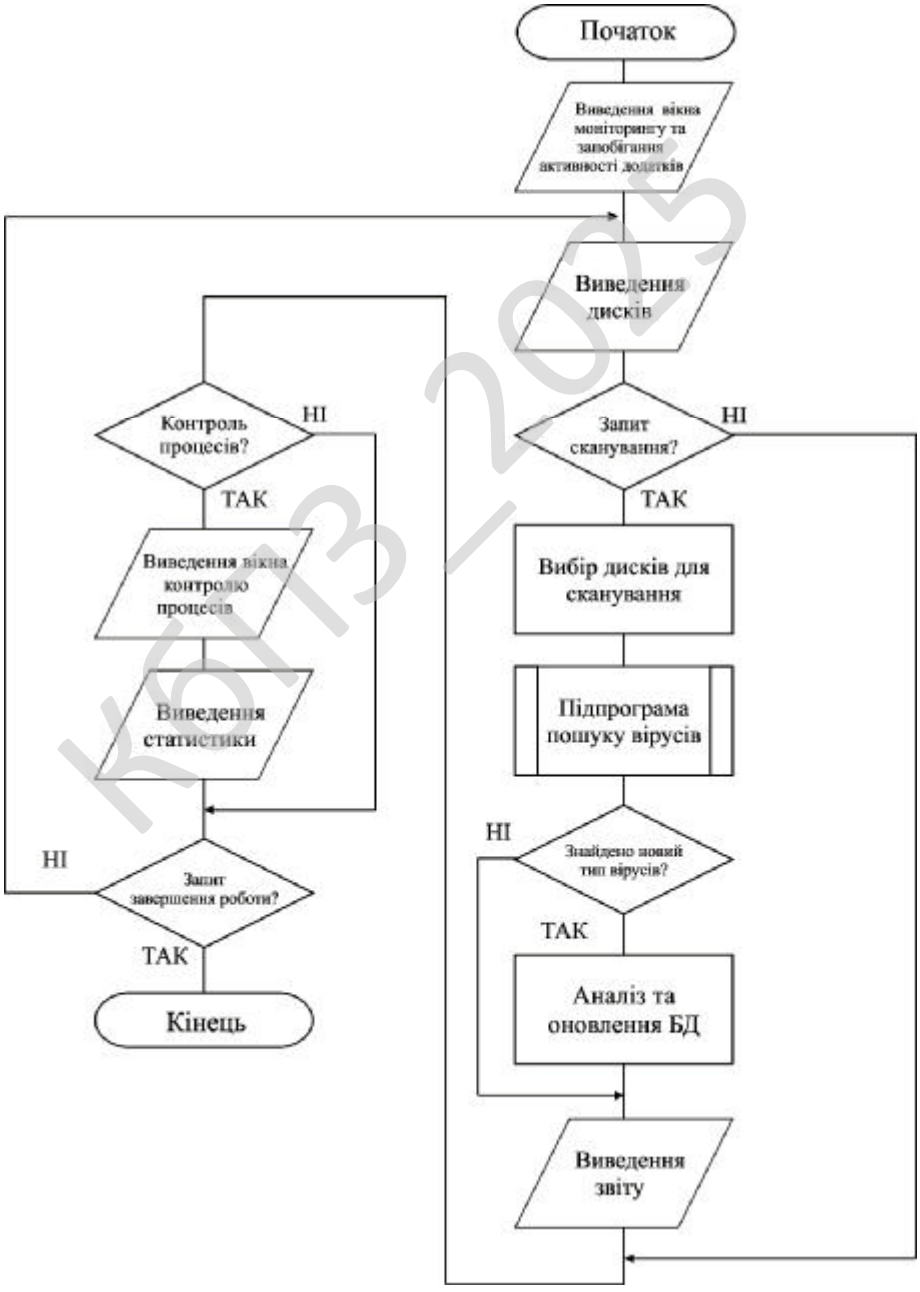

Рисунок 4.1 – Блок-схема основної програми

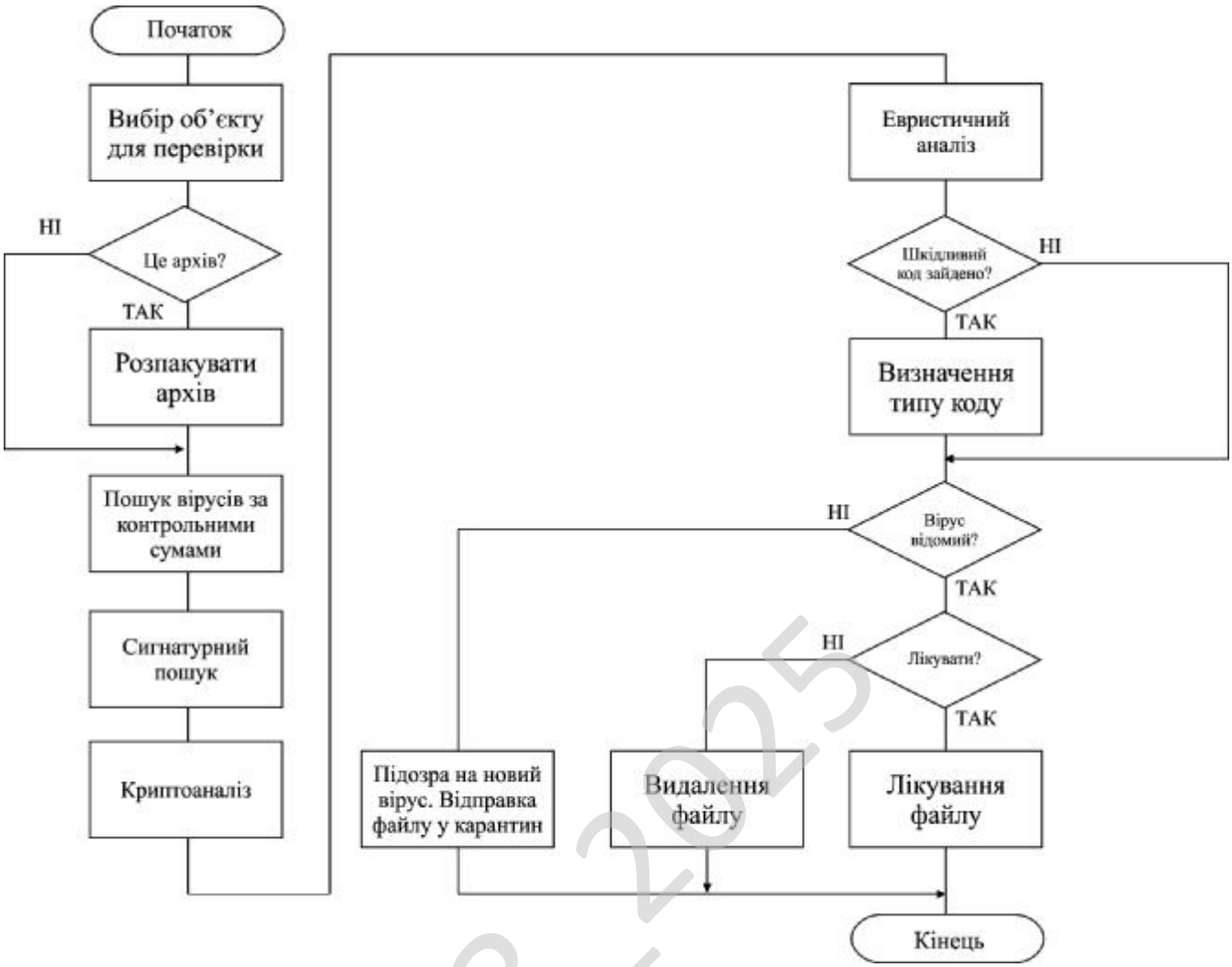

Рисунок 4.2 – Блок-схема роботи підпрограми

З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підпрограми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ. При роботі підпрограми виконується основний функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

Було використано підходи з використанням UML, це уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування. Є невід'ємною частиною уніфікованого процесу розробки

програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, називаної UML-моделлю. UML був створений для визначення, візуалізації, проектування й документування в основному програмних систем. UML не є мовою програмування, але в засобах виконання UML-моделей як інтерпретованого коду можлива кодогенерація.

UML може бути застосовано на всіх етапах життєвого циклу аналізу бізнес-систем і розробки прикладних програм. Різні види діаграм які підтримуються UML, і найбагатший набір можливостей представлення певних аспектів системи робить UML універсальним засобом опису як програмних, так і ділових систем.

Діаграми дають можливість представити систему (як ділову, так і програмну) у такому вигляді, щоб її можна було легко перевести в програмний код. Основною причиною використання мови UML є спілкування розробників між собою.

Крім того, UML спеціально створювалася для оптимізації процесу розробки програмних систем, що дозволяє збільшити ефективність їх реалізації у кілька разів і помітно поліпшити якість кінцевого продукту.

UML прекрасно зарекомендувала себе в багатьох успішних програмних проектах. Засоби автоматичної генерації кодів дозволяють перетворювати моделі мовою UML у вихідний код об’єктно-орієнтованих мов програмування, що ще більш прискорює процес розробки. Практично усі CASE-засоби (програми автоматизації процесу аналізу і проектування) мають підтримку UML. Моделі розроблені в UML, дозволяють значно спростити процес кодування і направити зусилля програмістів безпосередньо на реалізацію системи.

Діаграми підвищують супроводжуваність проекту і полегшують розробку документації.

UML необхідний:

– Керівникам проектів, які керують розподілом завдань і контролем за проектом.

– Проектувальникам інформаційних систем які розробляють технічні завдання для програмістів.

– Бізнес-аналітикам, які досліджують реальну систему і здійснюють інжиніринг і реінжиніринг бізнесу компанії.

– Програмістам які реалізовують модулі інформаційної системи.

При модифікації системи об'єктний підхід дозволяє легко включати в систему нові об'єкти і виключати застарілі без істотної зміни її життєздатності. Використання побудованої моделі при модифікаціях системи дає можливість усунути небажані наслідки змін, оскільки вони не ламають структури системи, а тільки змінюють поведінку об'єктів.

Також при розробці бакалаврської дипломної роботи було використано наступні підходи UML: діаграма прецедентів (діаграми поведінки типу); Діаграма класів; Діаграма компонент; Діаграма об'єктів; Діаграма розгортання.

Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в наступному: проектована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання (use case) використовують для описання послуг, які система надає актору. Іншими

словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором.

При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

– асоціації (association relationship);

– включення (include relationship);

– розширення (extend relationship);

– узагальнення (generalization relationship).

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме – за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення (include) у мові UML – це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності (dependency) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення розширення (extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

Діаграма класів це статичне представлення структури моделі. Відображає статичні (декларативні) елементи, такі як: класи, типи даних, їх зміст та відношення.

Діаграма класів, також, може містити позначення для пакетів та може містити позначення для вкладених пакетів. Також, діаграма класів може містити

позначення деяких елементів поведінки, однак їх динаміка розкривається в інших типах діаграм.

Діаграма класів (class diagram) служить для представлення статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування. На цій діаграмі показують класи, інтерфейси, об'єкти й кооперації, а також їхні відносини.

В UML існують наступні типи зв'язків які використовуються у діаграмі класів: Асоціації; Агрегація; Композиція.

Асоціації це якщо між двома класами визначена асоціація, то можна переміщатися від об'єктів одного класу до об'єктів іншого. Цілком припустимі випадки, коли обидва кінці асоціації відносяться до одного і того ж класу. Це означає, що з об'єктом деякого класу дозволено зв'язати інші об'єкти з того ж класу. Асоціація, що зв'язує два класи, називається бінарної. Можна, хоча це рідко буває необхідним, створювати асоціації, що зв'язують відразу кілька класів. Графічно асоціація зображується у вигляді лінії, що з'єднує клас сам з собою або з іншими класами.

Асоціації може бути присвоєно ім'я, яке описує природу відносини. Зазвичай ім'я асоціації не вказується, якщо тільки ви не хочете явно задати для неї рольові імена або у вашій моделі настільки багато асоціацій, що виникає необхідність посилатися на них і відрізняти один від одного. Ім'я буде особливо корисним, якщо між одними і тими ж класами існує кілька різних асоціацій.

Клас, що бере участь в асоціації, грає в ній деяку роль. По суті, це "обличчя", яким клас, що знаходиться на одній стороні асоціації, звернений до класу з іншого її боку. Можна явно позначити роль, яку клас грає в асоціації.

Часто при моделюванні буває важливо вказати, скільки об'єктів може бути пов'язано допомогою одного примірника асоціації. Це число називається кратністю (Multiplicity) ролі асоціації та записується або як вираз, значенням якого є діапазон значень, або в явному вигляді.

Вказуючи кратність на одному кінці асоціації, ви тим самим говорите, що на цьому кінці саме стільки об'єктів повинно відповідати кожному об'єкту на протилежному кінці. Кратність можна задати рівною одиниці (1), можна вказати діапазон: "нуль або одиниця" (0..1), "багато" (0 .. *), "одиниця або більше" (1 .. *). Дозволяється також вказувати певне число (наприклад, 3). За допомогою списку можна задати і більш складні кратності, наприклад 0. . 1, 3..4, 6 .. *, що означає "будь-яке число об'єктів, крім 2 і 5".

Агрегація це проста асоціація між двома класами відображає структурний відношення між рівноправними сутностями, коли обидва класу знаходяться на одному концептуальному рівні і ні один не є більш важливим, ніж інший. Але іноді доводиться моделювати відношення типу «частина/ціле», в якому один з класів має більш високий ранг (ціле) і складається з декількох менших за рангом (частин).

Ставлення такого типу називають агрегацією; воно зараховане до відносин типу «має» (з урахуванням того, що об'єкт-ціле має кілька об'єктів-частин). Агрегація є окремим випадком асоціації і зображується у вигляді простої асоціації з незафарбованим ромбом з боку «цілого». Графічно агрегація представляється порожнім ромбом на блоці класу, і лінією, яка від цього ромба до міститься класу.

Композиція це більш суворий варіант агрегації. Відома також як агрегація за значенням.

Композиція має жорстку залежність часу існування екземплярів класу контейнера та примірників містяться класів. Якщо контейнер буде знищений, то весь його вміст буде також знищено. Графічно представляється як і агрегація, але з зафарбовані ромбиком.

Діаграма компонент в UML це діаграма, на якій відображаються компоненти, залежності та зв'язки між ними.

Діаграма компонент відображає залежності між компонентами програмного забезпечення, включаючи компоненти вихідних кодів, бінарні компоненти, та компоненти, що можуть виконуватись.

Модуль програмного забезпечення може бути представлено в якості компоненти. Деякі компоненти існують під час компіляції, деякі – під час компонування, а деякі під час роботи програми.

Діаграма компонент відображає лише структурні характеристики, для відображення окремих екземплярів компонент слід використовувати діаграму розгортування.

Компоненти об'єднуються разом використовуючи структурні зв'язки (assembly connector) щоб об'єднати інтерфейси двох компонент. Це ілюструє зв'язок типу «клієнт-сервер».

Структурна взаємодія – «зв'язок двох компонент, який передбачає, що один з них надає послуги, потрібні іншому компоненту».

При використанні діаграми компонент щоб показати внутрішню структуру компонента, клієнтські та серверні інтерфейси можуть утворювати пряме з'єднання з внутрішніми. Таке з'єднання називається з'єднанням делегації.

Діаграма об'єктів в UML це діаграма, що відображає об'єкти та їх зв'язки в певний момент часу. Діаграма об'єктів може розглядатись як окремий випадок діаграми класів, на якій можуть бути представлені як класи, так і екземпляри (об'єкти) класів. Схожою за змістом є діаграма взаємодії (collaboration diagram).

Діаграми об'єктів не мають власної нотації. Оскільки діаграми класів можуть відображати об'єкти, то діаграма класів, на якій відображено лише об'єкти, та не відображено класи, може вважатись діаграмою об'єктів.

Діаграма об'єктів відображає об'єкти та зв'язки в певний момент роботи програми. Об'єкти можуть містити інформацію про власні значення а не про описання. Для відображення загальних шаблонів об'єктів та зв'язків, що можуть багаторазово створюватись під час роботи програми, слід використовувати діаграму взаємодії, яка може відображати характеристики об'єктів та зв'язків. Екземпляр діаграми взаємодії створює діаграму об'єктів.

Діаграма об'єктів не відображає еволюцію системи під час роботи. Натомість, слід використовувати діаграми взаємодії з повідомленнями, або діаграми послідовності.

Діаграма розгортання (deployment diagram) це діаграма в UML, на якій відображаються обчислювальні вузли під час роботи програми, компоненти, та об'єкти, що виконуються на цих вузлах. Компоненти відповідають представленню робочих екземплярів одиниць коду. Компоненти, що не мають представлення під час роботи програми на таких діаграмах не відображаються; натомість, їх можна відобразити на діаграмах компонент. Діаграма розгортання відображає робочі екземпляри компонент, а діаграма компонент, натомість, відображає зв'язки між типами компонент.

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм MMB, в основі якого лежить змішування операцій різних алгебраїчних груп. MMB – ітеративний алгоритм, що складається з лінійних дій (XOR і використання ключа) і паралельного застосування чотирьох великих оборотних нелінійних підстановок. Ці підстановки визначаються за допомогою множення по модулю 2^{32}-1 з постійними множниками. У підсумку з'являється алгоритм, що використовує 128-бітовий ключ і 128-бітовий блок.

Алгоритм MMB оперує 32-бітовими підблоками тексту (x_0, x_1, x_2, x_3) і 32-бітовими підблоками ключу (k_0, k_1, k_2, k_3). Це спрощує реалізацію алгоритму на сучасних 64-бітових процесорах. Чергуючись із операцією XOR, шість разів використовується нелінійна функція *f*. Запишемо операції алгоритму (всі операції з індексами виконуються по модулю 4):

$$x_i = x_i \oplus k_i \text{ для } i = 0..3$$

$$f(x_0, x_1, x_2, x_3)$$

$$x_i = x_i \oplus k_{i+1} \text{ для } i = 0..3$$

$$f(x_0, x_1, x_2, x_3)$$

$$x_i = x_i \oplus k_{i+2} \text{ для } i = 0..3$$

$$f(x_0, x_1, x_2, x_3)$$

$$x_i = x_i \oplus k_i \text{ для } i = 0..3$$

$$f(x_0, x_1, x_2, x_3)$$

$$x_i = x_i \oplus k_{i+1} \text{ для } i = 0..3$$

$$f(x_0, x_1, x_2, x_3)$$

$$x_i = x_i \oplus k_{i+2} \text{ для } i = 0..3$$

$$f(x_0, x_1, x_2, x_3)$$

Функція f виконується в три кроки:

1. $x_i = c_i * x_i$ для $i = 0..3$ (Якщо на вході множення одні одиниці, то на виході – теж одні одиниці).

2. Якщо молодший значущий біт $x_0 = 1$, то $x_0 = x_0 \oplus C$. Якщо молодший значущий байт $x_3 = 0$, то $x_3 = x_3 \oplus C$.

3. $x_i = x_{i-1} \oplus x_i \oplus x_{i+1}$ для $i = 0..3$.

Всі операції з індексами виконуються по модулю 4. Операція множення на кроці 1 виконується по модулі 2^{32}-1. Спеціальний випадок для даного алгоритму: якщо другий операнд дорівнює 2^{32}-1, результат теж дорівнює 2^{32}-1. В алгоритмі використовуються наступні константи:

C = 2aaaaaaa, c_0 = 025f1cdb, $c_1 = 2*c_0$, $c_2 = 2^3 * c_0$, $c_3 = 2^7 * c_0$.

Константа C – «найпростіша» константа без кругової симетрії, високою трійковою вагою й нульовим молодшим значущим бітом. У константи c_0 є інші особливі характеристики. Константи c_1, c_2 і c_3 – зрушені версії c_0, і служать для запобігання атак, заснованих на симетрії.

Розшифрування виконується у зворотному порядку, Етапи 2 і 3 інверсні їм самим. На етапі 1 замість c_i використовується c_i^{-1}. Значення c_0^{-1} = 0dad4694.

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Розглянемо розроблене ПЗ системи інтелектуального моніторингу та запобігання активності додатків яке зображено на рисунку 5.1. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

– Функції представлені у графічному вигляді (іконки).

– Розділів: Загальне; Контроль процесів; Додатково; Фільтр сканування; Шляхи сканування.

– Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.

– Функціональних кнопок ПЗ.

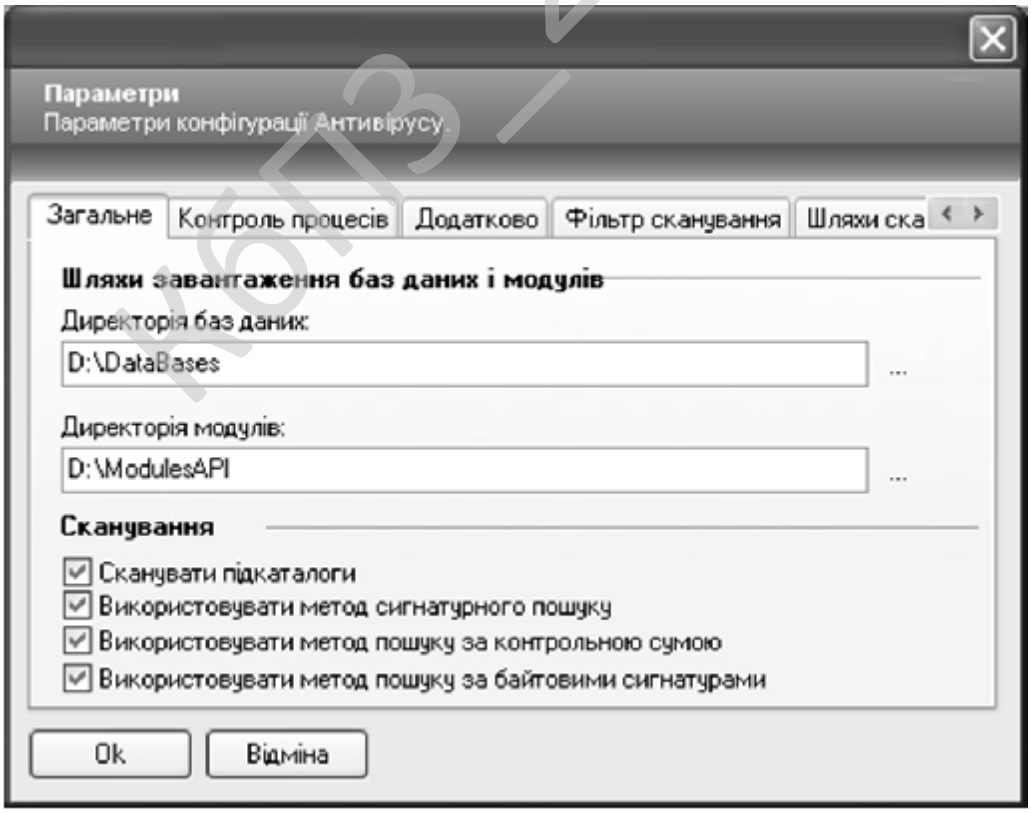

Рисунок 5.1 – Головне вікно ПЗ

Розроблена програма має дуже простий і інтуїтивно зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий.

Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

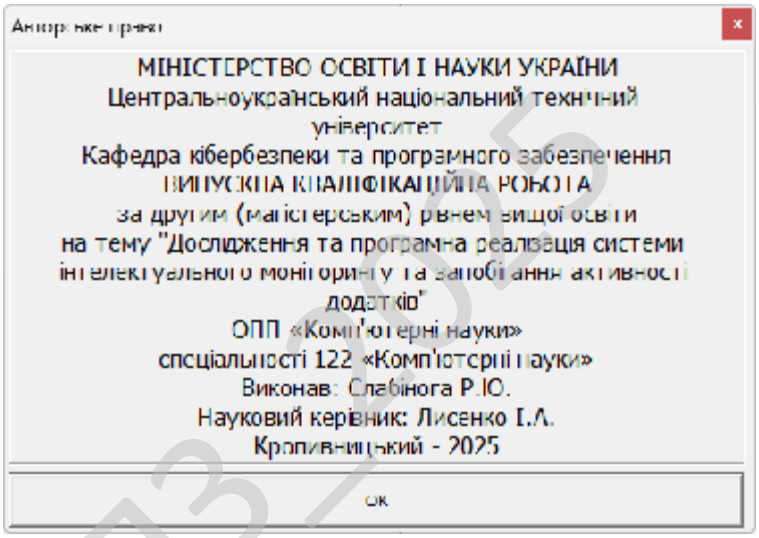

Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку виробника так і з боку споживача. Оскільки кожна програмна система є

унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

– Випуск.

– Встановлення та активація.

– Деактивація.

– Адаптація.

– Обновлення.

– Вмонтування.

– Відстежування версій.

– Видалення.

– Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати IT рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене IT рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження;

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються

в ІТ рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

– відповідності поставленим вимогам;

– правильна відповідь для усіх можливих вхідних даних;

– виконання функцій за прийнятний час;

– практичність;

– сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом білої скриньки засноване на аналізі керуючої структури програми. Програма вважається повністю перевіреною, якщо проведено вичерпне тестування маршрутів (шляхів) її графа управління.

У цьому випадку формуються тестові варіанти, в яких:

– Гарантується перевірка всіх незалежних маршрутів програми.

– Знаходяться гілки True, False для всіх логічних рішень.

– Виконуються всі цикли (у межах їхніх кордонів та діапазонів).

– Аналізується правильність внутрішніх структур даних.

Недоліки тестування "білої скриньки":

– Кількість незалежних маршрутів може бути дуже велика.

– Повне тестування маршрутів не гарантує відповідності програми вихідним вимогам до неї.

– У програмі можуть бути пропущені деякі маршрути.

– Не можна виявити помилки, поява яких залежить від даних.

Переваги тестування "білої скриньки" пов'язані з тим, що принцип «білої скриньки» дозволяє врахувати особливості програмних помилок:

– Кількість помилок мінімально в «центрі» і максимально на «периферії» програми.

– Попередні припущення про ймовірність потоку керування або даних у програмі часто бувають некоректними. У результаті типовим може стати маршрут, модель обчислень за яким опрацьована слабо.

– При записі алгоритму програмного забезпечення у вигляді тексту на мові програмування можливе внесення типових помилок трансляції (синтаксичних та семантичних).

– Деякі результати в програмі залежать не від вихідних даних, а від внутрішніх станів програми.

Проводилось тестування чорної скриньки.

Основне місце програми тестів «чорної скриньки» – інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

– Як виконуються функції програми.

– Як приймаються вихідні дані.

– Як виробляються результати.

– Як зберігається цілісність зовнішньої інформації.

При тестуванні «чорної скриньки» розглядаються системні характеристики програм, ігнорується їхня внутрішня логічна структура. Вичерпне тестування, як правило, неможливе.

Наприклад, якщо в програмі 10 вхідних величин і кожна приймає по 10 значень, то кількість тестових варіантів становитиме 10^10. Тестування «чорної скриньки» не реагує на багато особливостей програмних помилок.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, які забезпечують повну перевірку всіх функціональних вимог до програми.

Програмний виріб тут розглядається як «чорна скринька», чию поведінку можна визначити тільки дослідженням його входів та відповідних виходів. При такому підході бажано мати:

– Набір, утворений такими вхідними даними, які призводять до аномалій у поведінці програми (назвемо його ITc).

– Набір, утворений такими вхідними даними, які демонструють дефекти програми (назвемо його OT).

Будь-який спосіб тестування «чорної скриньки» повинен:

– Виявити такі вхідні дані, які з високою ймовірністю належать набору ITc;

– Сформулювати такі очікувані результати, які з високою імовірністю є елементами набору OT.

Принцип «чорної скриньки» не альтернативний принципу «білої скриньки». Скоріше це доповнює підхід, який виявляє інший клас помилок.

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

– Некоректних чи відсутніх функцій;

– Помилок інтерфейсу;

– Помилок у зовнішніх структурах даних або в доступі до зовнішньої бази даних;

– Помилок характеристик (необхідна ємність пам'яті і т.д.);

– Помилок ініціалізації та завершення.

Обрано умови розповсюдження – Freeware.

Це власницьке програмне забезпечення, котре можна Безоплатно використовувати протягом необмеженого терміну без обмежень у функціональності, і поширюване без сирцевих кодів.

Автори такого програмного забезпечення, як правило, хочуть «дати щось спільноті», але хочуть також контролювати його подальшу розробку. Іноді, коли програмісти вирішують припинити розробку, вони передають сирцевий код іншим програмістам, або ж спільності як вільне програмне забезпечення.

Дуже часто плутають поняття «безплатне програмне забезпечення» та «вільне програмне забезпечення», хоча вони суттєво відрізняються.

Безплатне програмне забезпечення можна безоплатно встановлювати та використовувати (іноді з певними обмеженнями, як, наприклад, «безплатне для домашнього або некомерційного вжитку»), в той час як вільне програмне забезпечення можна продавати за будь-яку суму, але при тому, у користувача, котрий його отримує, повинні бути права на вивчення, модифікацію та поширення сирцевих кодів одержаної програми.

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи інтелектуального моніторингу та запобігання активності додатків.

Метою розробки є дослідження та програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків.

Об'єктом дослідження є процес інтелектуального моніторингу та запобігання активності додатків.

Предметом дослідження є методи інтелектуального моніторингу та запобігання активності додатків.

Методи дослідження базуються на методах інтелектуального моніторингу, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод інтелектуального моніторингу та запобігання активності додатків.

– Розроблено вітчизняний продукт інтелектуального моніторингу та запобігання активності додатків, який має більш широкі можливості, на відміну від існуючих аналогів.

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати дослідження та розробки системи інтелектуального моніторингу та запобігання активності додатків можуть бути насамперед корисними для підприємств, які мають розгалужену ІТ-інфраструктуру й використовують сервери, мережеве обладнання та корпоративні сервіси для підтримки своєї діяльності. Для таких компаній стабільність і безперебійність роботи інформаційних систем є критично важливими, тому можливість своєчасного виявлення несправностей або перевантажень стає суттєвою конкурентною перевагою. Саме система моніторингу допомагає контролювати роботу мережевих пристроїв у режимі реального часу, виявляючи проблеми ще до того, як вони вплинуть на користувачів.

Особливий інтерес до таких систем можуть проявити ІТ-компанії, які займаються наданням послуг хостингу, розробкою програмного забезпечення або підтримкою клієнтів. Для них швидкість реагування на інциденти та якість технічного обслуговування є показниками репутації, а отже, від роботи системи моніторингу залежить рівень довіри клієнтів і лояльність користувачів. Такі підприємства часто працюють у середовищі, де навіть хвилинна затримка чи зупинка сервера призводить до фінансових збитків, тому автоматизація контролю за станом мережі – це не розкіш, а необхідність.

Крім комерційних компаній, результати дослідження будуть актуальними для державних структур, освітніх установ і організацій, які мають внутрішні мережі та зберігають великі обсяги інформації. У таких установах впровадження системи моніторингу підвищує ефективність роботи ІТ-відділів, зменшує ризик втрати даних і допомагає раціонально використовувати наявні ресурси.

Не менш важливим є значення цієї розробки для навчальних і наукових закладів. Вони можуть використовувати систему як навчальну платформу для підготовки фахівців у сфері інформаційних технологій. Студенти отримають можливість не лише спостерігати за реальною роботою системи моніторингу, а й аналізувати дані, моделювати різні ситуації та вчитися реагувати на інциденти. Таким чином, результати дослідження мають універсальний характер і можуть бути впроваджені як у бізнесі, так і в освіті.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Для оцінки привабливості програмного продукту було проведено експертне опитування серед фахівців у галузі IT-інфраструктури, адміністраторів систем і представників компаній, що мають досвід використання схожих рішень. Експертам було запропоновано оцінити систему за основними критеріями – функціональні можливості, надійність, простота впровадження, масштабованість, вартість експлуатації та потенційна економічна ефективність.

Більшість експертів високо оцінили саме інтелектуальну частину системи – можливість автоматичного сповіщення про інциденти, генерацію аналітичних звітів і прогнозування потенційних відмов обладнання. Особливо було відзначено, що система працює стабільно навіть при великому навантаженні й може адаптуватися до різних типів мережевої інфраструктури, що робить її універсальною.

За результатами оцінки середній рівень привабливості продукту склав 8,7 бала з 10 можливих. Експерти зазначили, що така система може мати великий попит серед середніх і великих підприємств, особливо якщо її вартість залишатиметься конкурентною. Також було підкреслено, що простота інтерфейсу та можливість кастомізації під конкретного користувача є суттєвими перевагами, які підвищують комерційний потенціал рішення.

Таким чином, метод експертних оцінок показав, що система має високу ринкову привабливість, відповідає актуальним потребам бізнесу та може стати успішним продуктом за умови належного маркетингового просування та підтримки користувачів.

7.3 Вибір методу оцінки вартості ПЗ

Для оцінки вартості розробки системи інтелектуального моніторингу та запобігання активності додатків доцільно використовувати витратний метод. Він передбачає визначення всіх фактичних витрат, які були понесені під час створення програмного продукту, включаючи оплату праці розробників, витрати на апаратне забезпечення, ліцензії, тестування та впровадження. Такий підхід дозволяє точно визначити базову собівартість проєкту, що є особливо важливим для невеликих команд і стартапів.

Однак, у випадку комерційного впровадження, доцільно поєднати цей підхід із дохідним методом. Дохідний метод дає змогу оцінити майбутні вигоди, які підприємство отримає після впровадження системи. Наприклад, скорочення простоїв серверів, підвищення ефективності роботи персоналу та зменшення витрат на ручну діагностику мережі є прямими джерелами економічної вигоди.

Такий комбінований підхід дозволяє не лише визначити початкову вартість розробки, а й обґрунтувати економічну доцільність проєкту. Він допомагає потенційним інвесторам побачити не просто витрати, а реальні фінансові перспективи, які відкриває впровадження системи.

У результаті використання комбінованої моделі оцінки можна отримати повну картину вартості та окупності проєкту, що стане основою для прийняття управлінських рішень щодо його реалізації чи масштабування.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Компанія має розгалужену IT-інфраструктуру, яка включає сервери, мережеве обладнання, робочі станції, системи зберігання даних і корпоративні сервіси. До впровадження системи моніторингу контроль за станом мережі здійснювався вручну: адміністратори виявляли проблеми лише після звернень користувачів або повного виходу сервісів із ладу. Це призводило до простоїв, затримок у роботі та фінансових втрат. Основна мета впровадження системи мережевого моніторингу – забезпечити цілодобове автоматичне відстеження стану обладнання, серверів і додатків, оперативне реагування на інциденти, зниження кількості простоїв і запобігання критичним збоям у роботі IT-інфраструктури. Вхідні дані зафіксовано в таблиці 7.1.

Таблиця 7.1 – Вихідні дані для розрахунку

Показник	До впровадження	Після впровадження	Економічний ефект
Кількість простоїв серверів на рік	20 випадків	5 випадків	–15
Середня тривалість простою одного сервера	4 години	1 година	–3 години
Середні втрати підприємства за 1 годину простою	25 000 грн	5 000 грн	–20 000 грн
Витрати на ручну діагностику й усунення збоїв	300 000 грн/рік	150 000 грн/рік	–150 000 грн
Вартість впровадження системи моніторингу	—	—	450 000 грн
Річні витрати на підтримку системи	—	—	100 000 грн

Розрахунок економічного ефекту демонструє наступне: зменшення збитків від простоїв – 1 975 000 грн/рік, економія на технічному обслуговуванні – 150 000 грн/рік, сукупний річний ефект – 2 125 000 грн/рік, чистий ефект – 2 025 000 грн/рік, термін окупності (Payback Period) – 0,22 року (~2,5 місяці), коефіцієнт ефективності (ROI) – 450%.

Додаткові (немонетарні) переваги: підвищення стабільності IT-інфраструктури завдяки ранньому виявленню збоїв, зменшення навантаження на IT-персонал через автоматизацію моніторингу, покращення SLA (Service Level Agreement) і задоволеності користувачів, прогнозування потенційних проблем через аналітику та звітність у реальному часі, зростання репутації підприємства, адже мінімізуються ризики затримок у наданні послуг або збою критичних бізнес-процесів.

Таким чином, моніторинг стає не лише технічним інструментом, а й важливою складовою операційної надійності та конкурентоспроможності підприємства.

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Просування системи моніторингу має будуватися на поетапному підході, що включає як технічну демонстрацію, так і інформаційне просування. На першому етапі варто створити пілотний проєкт і запропонувати його впровадження у невеликій кількості підприємств для збору відгуків і реальних кейсів. Це дозволить перевірити ефективність системи в реальних умовах і створити довіру до продукту.

Далі важливо забезпечити інформаційну присутність продукту – через участь у галузевих конференціях, IT-форумах, онлайн-презентаціях і спеціалізованих публікаціях. Саме через публічну експертну комунікацію формується репутація розробника та усвідомлення цінності рішення на ринку.

Наступним етапом є розширення партнерських зв’язків. Доцільно співпрацювати з ІТ-компаніями, які займаються інтеграцією корпоративних систем, адже вони можуть пропонувати продукт своїм клієнтам як частину комплексного рішення. Водночас слід розробити гнучку цінову політику – наприклад, ліцензування за кількістю пристроїв або модель передплати, що зробить продукт доступнішим для малого та середнього бізнесу.

Просування має супроводжуватися технічною підтримкою користувачів, оновленнями та навчанням персоналу. Це створює позитивний досвід використання продукту та сприяє формуванню довгострокових відносин із клієнтами. У підсумку правильна стратегія просування допоможе не лише збільшити продажі, а й побудувати впізнаваний бренд на ринку ІТ-рішень.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для оптимізації каналів збуту варто поєднати прямі продажі з цифровими платформами розповсюдження програмного забезпечення. Власний сайт компанії може стати не лише вітриною продукту, а й каналом комунікації з клієнтами, де вони зможуть отримати демо-версію, консультацію або підтримку. Це сприятиме зниженню витрат на маркетинг і збільшенню довіри.

Додатково ефективним буде впровадження партнерської програми для системних інтеграторів і реселерів, які вже мають доступ до корпоративних клієнтів. Така модель дозволяє розширити охоплення ринку без суттєвих додаткових інвестицій. Також можна запропонувати гібридну форму реалізації: ліцензування для великих компаній і модель SaaS (Software as a Service) для малого бізнесу. Це підвищить доступність системи та дозволить гнучко реагувати на потреби різних сегментів ринку. Ключовим напрямом оптимізації збуту є створення якісного сервісу після продажу – технічна підтримка, регулярні оновлення, аналітичні звіти. Усе це забезпечує стабільність роботи клієнта й стимулює його до подальшої співпраці.

7.7 Визначення ключових факторів успіху конкретного проєкту

Основним фактором успіху є стабільність і надійність системи. Якщо система моніторингу працює без збоїв і забезпечує реальну користь, вона швидко здобуває довіру користувачів. Технологічна якість продукту, його здатність масштабуватися й інтегруватися з іншими IT-рішеннями відіграють ключову роль у його життєздатності.

Другим важливим чинником є професійна команда розробників і технічної підтримки. Клієнти цінують не лише продукт, а й можливість отримати швидку допомогу у випадку проблем або питань. Від рівня компетенції фахівців залежить не лише якість обслуговування, а й довгострокові відносини з партнерами.

Не менш значущим є гнучкість системи – можливість адаптувати її під специфіку кожного клієнта. Різні компанії мають різну інфраструктуру, тому універсальне, але налаштовуване рішення стає перевагою.

І, нарешті, успіх будь-якого IT-проєкту визначається здатністю постійно вдосконалюватися. Регулярні оновлення, впровадження нових технологій і зворотний зв’язок із користувачами формують довіру й підтримують актуальність продукту на ринку. Саме ці чинники разом створюють основу для стабільного розвитку та комерційного успіху системи моніторингу.

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Електронно-обчислювальна машина (ЕОМ) відіграє важливу роль у житті сучасної людини. Кожного дня мільйони людей використовують ЕОМ для пошуку необхідної інформації, спілкуванні у соціальних мережах, перегляду новин, роботи тощо. Багато людей користуються ЕОМ у професійних цілях, оскільки завдяки ЕОМ з'явилося багато нових професій. Тому для розробника хмарних сервісів так важливо розробити зручний інтерфейс для зручного сприйняття інформації, та необхідний функціонал, який буде відповідати необхідним вимогам та навантаженням. Все це вимагає багато часу та великого навантаження з боку розробників. Тому так важливо слідкувати за умовами праці, в яких відбувається робочий процес. Оскільки захворювання можуть бути спричинені надмірним фізичним або розумовим навантаженням, через велику нервово-емоційну напругу, або через виробниче середовище. В даному розділі магістерської роботи проведемо аналіз основних чинників при роботі програміста.

Законом України "Про охорону праці" регламентуються загальні положення державної політики в галузі охорони праці, а конкретизуються ці положення нормативно-правовими актами про охорону праці, зокрема Наказом Міністерства соціальної політики України 14.02.2018 № 207, який зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за №508/31960 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями», яким затверджено нормативно-правовий акт з охорони праці НПАОП 0.00-7.15-18, «Правила охорони праці під час експлуатації електронно-обчислювальних машин», та «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98.

8.2 Шкідливі і небезпечні фактори при роботі з комп'ютером

Електронно-обчислювальна машина (ЕОМ) та інше обладнання є джерелами небезпеки ураження електричним струмом. Так як робота програміста характеризується істотним зоровим навантаженням, то вимагає належного освітлення. У приміщенні, в якому працюють люди (у т.ч. програмісти) необхідно створити належний мікроклімат, параметри якого регламентуються, Державними санітарними правилами і нормами, зокрема ДСанПіН 3.3.2.007-98.

На роботу програміста впливають наступні фактори: невідповідний мікроклімат приміщення (температура, вологість), недостатня освітленість робочої зони, підвищений рівень шуму та електромагнітного випромінювання, порушення іонного складу повітря, неправильна ергономічна організація робочого місця, ризики, пов'язані із погіршенням зору, порушенням фізичного стану, стресом тощо.

Шкідливими факторами при роботі з персональним комп'ютером є неіонізуюче випромінювання промислової частоти, збільшене нервово- емоційне навантаження на оператора, збільшення навантаження на органи зору та дрібні стереостатичні рухи кінцівок. Ці фактори можуть викликати у працівника певні розлади здоров'я, зокрема підвищення артеріального тиску, кон'юктивіти, тендовагініти та інші захворювання.

Комп'ютер, як і будь-який електричний прилад, особливо при його неправильному підключенні, може бути джерелом ураження оператора електричним струмом. Саме тому всі працівники, які працюють з персональним комп'ютером, повинні мати першу (або другу) групу допуску з електробезпеки.

Через наявність зазначених факторів працівники, які працюють з персональними комп'ютерами, підлягають попередньому та періодичному медичному огляду згідно з пунктом 6.2.3 додатку 4 до наказу Міністерства охорони здоров'я України «Про затвердження Порядку проведення медичних оглядів працівників певних категорій» від 21 травня 2007 року № 246.

8.3 Аналіз санітарно-гігієнічних умов праці на робочому місці програміста

Оптимальна температура в приміщенні для праці має становити 20-24°С, відносна вологість – 40-60 %, атмосферний тиск – 750 мм. рт. ст., запиленість не повинна перевищувати 10 мг/м^3, швидкість руху повітря – 0,1 м/с.

Через те, що обчислювальна техніка є джерелом тепловиділення, організація мікроклімату потребує додаткових зусиль: кондиціонування, провітрювання, використання систем опалення тощо. Об’єм приміщень повинен передбачатися з урахуванням як мінімум 20 м^3 /на особу [4].

Монітори комп’ютерів є джерелом випромінювання, яке може зашкодити здоров’ю людини. Для убезпечення роботи з комп’ютером відстань від монітора повинна становити не менше 50 см, бажано використовувати монітори зі зниженим рівнем, скорочувати час безперервної роботи за комп'ютером (робити п’ятнадцяти хвилинні перерви після кожних півтори години праці). Також в приміщенні необхідно встановлювати іонізатори повітря, використовувати нейтралізатори та зволожувачі.

Комп’ютери та периферійні пристрої є джерелами шуму, висока інтенсивність якого може призвести до проблем з органами слуху та негативно впливати на психологічний стан. Рівень шуму на робочому місці не повинен перевищувати 50 дБА [5]. Для зменшення рівня шуму можна використовувати звукопоглинальні пристрої, а стіни приміщень з комп’ютерами можуть бути покриті звукопоглинальними матеріалами. Поряд із шумом часто виникає вібрація. Для зменшення рівня вібрації в приміщенні на поверхні необхідно встановлювати віброізолятори.

Ергономічні показники робочого місця програміста мають бути наступними: висота робочої поверхні повинна складати 720 мм, розмір поверхні має становити 1600 х 1000 мм; під столом повинен бути простір з розмірами по глибині 650 мм; стіл повинен мати підставку для ніг, розташовану під кутом

15°до поверхні; відстань клавіатури від краю столу має бути не більше 300 мм; відстань між очима й екраном повинна складати 40 – 80 см; стілець повинен мати підйомно-поворотний механізм; висота сидіння має регулюватися в межах 400 – 500 мм, глибина – не менше 380 мм, а ширина – не менше 400 мм, висота опорної поверхні спинки має бути не менше 300 мм, ширина – не менше 380 мм. Кут нахилу спинки стільця до площини сидіння повинен змінюватися в межах 90 – 110° [6].

Проведений аналіз показує, що показники мікроклімату в приміщенні відповідають установленим нормам. Штучне опалення застосовується у холодний період року.

В літню пору застосовується кондиціонер.

Для боротьби з пилом робляться регулярні провітрювання та вологі прибирання приміщенні.

У приміщенні знаходяться наступні джерела шуму: принтер Prinics PicKit M1 Smartphone Photo Printer White, електродвигуни вентиляторів ЕОМ.

Робота програміста передбачає постійний візуальній контакт з моніторами комп'ютерів, та, як наслідок, значне навантаження на зір. Традиційно, це зорова робота високої або середньої точності. Для зорової роботи високої точності загальне освітлення (розподіл світла у всьому об'ємі приміщення) має становити 300 лк, комбіноване освітлення (поєднання загального і місцевого освітлення) – 750 лк. Штучне освітлення повинно бути рівномірним та використовуватися в світлий і темний час доби. Джерелами штучного освітлення можуть слугувати люмінесцентні лампи. Правильне освітлення передбачає уникнення відблисків на екранах.

З 2019 року діють Державні будівельні норми України "Природне і штучне освітлення" – ДБН В.2.5-28:2018 [4], у яких прописані вимоги до використання всіх освітлювальних приладів, у т.ч. світлодіодних.

Працю працівника, який постійно працює за комп'ютером, згідно ДБН В.2.5-28:2018 [4], можна віднести до роботи з малою точністю (найменший

розмір об'єкта розрізнення від 1 до 5 мм) V-го розряду зорової роботи, з великою контрастністю об'єкта розрізнення (символів на екрані дисплея), з темним тлом (під розряд зорової роботи В). Приміщення можна віднести до 1-ої групи приміщень, у яких проводиться розрізнення об'єктів зорової роботи при фіксованому напрямку лінії зору того, що працює на робочу поверхню. Для такого типу приміщень і розряду зорової роботи нормоване значення коефіцієнта природної освітленості (КПО) робочої поверхні (при поєднаному, спільному освітленні), повинен становити не більше 1,5%, освітленість при штучному висвітленні повинна становити 300 Лк. [4], Крім того все поле зору повинно бути освітлено достатньо рівномірно – це основна гігієнічна вимога. Оскільки яскраве світло на ділянці периферійного зору значно збільшує напруженість очей і, як наслідок, призводить до їх швидкої стомлюваності, ступінь освітлення приміщення і яскравість екрану комп'ютера повинні бути приблизно однаковими.

8.4 Розробка заходів з умов поліпшення охорони праці

Згідно аналізу умов праці в розглянутому приміщенні, ми одержали наступні результати:

– розмірі приміщення, у розрахунку на одному працюючого, відповідають нормативам;

– мікроклімат відповідає нормативному значенню;

– акустичні умови роботи не перевищують нормативних значень;

Таким чином можна припустити, що основною причиною можливого зниження працездатності програміста є психофізіологічний фактор, тому основна пропозиція буде така: дотримання позитивної психологічної атмосфери в колективі та регламентованого режиму праці та відпочинку, організація робочого місця з урахуванням ергономічних вимог.

Рекомендовані заходи: регулярні періодичні наочні огляди персоналом шляхів для евакуації людей із приміщення, відповідно до плану евакуації (який

повинен розташовуватись на видному місці у приміщенні), включення до колективного договору мінімально можливого вмісту аптечок з обов'язковою наявністю масок-клапанів, або іншого спорядження для штучного дихання. Регулярна періодична перевірка параметрів заземлення та занулення (вимірювання опору ланцюга).

Регулярна наочне знайомство персоналу із шляхами для евакуації людей із приміщення відповідно до плану евакуації, забезпечення розподільних щитів спеціальними розетками з заземлюючими контактами; організація заземлення всіх приладів і пристроїв, які працюють при напрузі вище 36 В.

Так як при ураженні електричним струмом у людини може статися фібриляція шлуночків серця, в організації бажано мати дефібрилятор і підготовлений персонал для роботи з ним.

8.5 Розрахункова частина

Проведемо розрахунок штучного освітлення за методом коефіцієнта використання світлового потоку для приміщення ширина якого складає 6 м, довжина – 7 м, висота – 2,9 м.

У зазначеному приміщенні працює 4 людей.

Для того, щоб визначити потрібну кількість світильників, які повинні забезпечити нормований рівень освітленості, визначимо світловий потік, що падає на робочу поверхню за формулою [1]:

$$F = E \cdot S \cdot K \cdot Z / n,$$

де:

F – світловий потік, що розраховується, Лм;

E – нормована мінімальна освітленість, Лк; E = 300 Лк;

S – площа освітлюваного приміщення (у нашому випадку S=6 х 7 = 42 м^2);

K – коефіцієнт запасу, що враховує зменшення світлового потоку лампи в результаті забруднення світильників у процесі експлуатації (його значення

залежить від типу приміщення і характеру робіт, що проводяться в ньому, в нашому випадку $K = 1,5$);

Z – відношення середньої освітленості до мінімальної (зазвичай приймається рівним 1.1... 1.2, у нашому випадку $Z = 1,1$);

n – коефіцієнт використання світлового потоку, (відношення світлового потоку, що падає на розрахункову поверхню, до сумарного потоку від усіх ламп і обчислюється в долях одиниці [8]; залежить від характеристик світильника, розмірів приміщення, забарвлення стін і стелі, що характеризуються коефіцієнтами відбиття від стін ($\rho_{стін.}$) і стелі ($\rho_{стелі}$), значення коефіцієнтів дорівнюють $\rho_{стін} = 50\%$ і $\rho_{стелі} = 50\%$.

Обчислимо індекс приміщення за формулою:

$$i = S/(h\cdot (A+B)),$$

де:

S – площа приміщення, $S = 42$ м2;

h – розрахункова висота підвісу, $h = 2,9$ м (співпадає з висотою стелі, оскільки лампи освітлення закріплюються на стелі);

A – ширина приміщення, $A = 6$ м;

B – довжина приміщення, $B = 7$ м.

Підставимо всі значення у формулу та визначимо індекс приміщення:

$$i=1,4.$$

Знаючи індекс приміщення, за знаходимо $n = 0,29$ (з табличних даних коефіцієнтів використання світлового потоку (n) світильників з відповідним типом лампам) [8]. Підставимо всі значення у формулу, визначимо світловий потік: F=71689 Лм.

Для розрахунку будемо використовувати світлодіодні стельові панелі Delux LED Panel 41 44 Вт, світловий потік яких $F_л = 3600$ Лм.

Число ламп визначається за формулою:

$$N=F/F_л$$

де:

F – світловий потік,

$F_л$ – світловий потік однієї лампи.

Підставимо всі значення у формулу та визначимо індекс приміщення:

$$N= 71689 / 3600 =19,9 \text{ шт.}$$

Приймаємо необхідну кількість світлодіодних світильників 20 шт.

Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз умов праці, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок штучного освітлення, як одного з ключових факторів впливу на працездатність та здоров'я програміста. Розроблено заходи з умов поліпшення охорони праці.

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи інтелектуального моніторингу та запобігання активності додатків.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів інтелектуального моніторингу та запобігання активності додатків.

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем інтелектуального моніторингу та запобігання активності додатків.

– Досліджена система інтелектуального моніторингу та запобігання активності додатків.

– На основі отриманих результатів досліджень створена програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання інтелектуального моніторингу та запобігання активності додатків.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створені програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм MMB.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Проведено маркетингове та економічне обґрунтування ІТ-проєкту, що дозволило визначити ключові фактори успіху даного проекту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Слабінога Р.Ю. Дослідження та програмна реалізація системи інтелектуального моніторингу та запобігання активності додатків // Збірник праць молодих науковців ЦНТУ. – Вип. 15. – Кропивницький: ЦНТУ, 2025.

2. Loren Kohnfelder. Designing Secure Software. No Starch Press. 2022. 332 p.

3. Samir Kumar Rakshit. Ethical Hacker's Penetration Testing Guide. BPB Online. 2022. 509 p.

4. Corey J. Ball. Hacking APIs. No Starch Press. 2022. 353 p.

5. Kevin Beaver. Hacking for Dummies. John Wiley & Sons. 2022. 419 p.

6. Mark S. Merkow. Practical Security for Agile and DevOps. CRC Press. 2022. 236 p

7. Derek Fisher. Application Security Program Handbook. Manning Publications. 2021. 155 p.

8. Cameron Wyatt PH.D. Kali Linux Tutorial. Independently published. 2021. 60 p.

9. Alex Matrosov, Eugene Rodionov, Sergey Bratus. Rootkits and Bootkits. No Starch Press. 2019. 450 p.

10. Петрик В.М., Присяжнюк М.М., Аль-Файюмі Халед та ін. «Системи інформаційної зброї та технології інформаційної війни»: підручник / Петрик В.М., Присяжнюк М.М., Аль-Файюмі Халед, Жарков Я.М., Смірнов О.А, Буравченко К.О., Давидюк А.В., Кононович В.Г., Корчинский В.В., Кудирко В.М., Фесенко А.О.; за заг. ред. В.М. Петрика, М.М. Присяжнюка.– К.: Видавничий центр “Кафедра”, 2025.– 320 с.

11. Усік, П.С., Смірнова, Т.В., Буравченко, К.О., Смірнов, О.А., Улічев, О.С., Смірнов, С.А. «Дослідження технологій забезпечення кібербезпеки банківських систем з використанням штучного інтелекту». *Кібербезпека: освіта, наука, техніка.* 2025. Том 1 № 29. С.704–716, 2025

12. Kuznetsov, O., Frontoni, E., Kuznetsova, K., Arnesano, M., Smirnov, O. «A secure biometric authentication architecture for blockchain-driven cyber-physical systems». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 193–224.

13. Kuznetsov, O., Atzeni, G., Arnesano, M., Randieri, C., Smirnov, O. «Secure IoT-based smart wheelchair system: From implementation to security enhancement strategy». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 225–257.

14. Kuznetsov, O., Smirnov, O., Kuznetsova, T., Shaikhanova, A., Svatowsky, I. «Privacy-utility trade-offs in IoT networks: A comparative analysis of differential privacy mechanisms for sensor data aggregation». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 589–622.

15. Lakhno, V., Malyukov, V., Smirnov, O., Bebeshko, B., Chubaievskiy, V., Zhumadilova, M., Malyukova, I., Smirnov, S. «Multifactorial Model for Targeted Attacks Counteracting Within the Framework of a Multi-Step Quality Game with Fuzzy Information». *8th International Symposium on Intelligent Informatics, ISI 2023*, 2025. vol 389. pp 377-389. Springer, Singapore.

16. Kuznetsov O., Frontoni E., Kuznetsova Y., Smirnov O., Moskovchenko I. «Trust-Based Security Architecture for Edge Computing: A Simulation Study of Dynamic Trust Evolution and Attack Detection». *CEUR Workshop Proceedings,* 2024, 3909, pp. 227–241.

17. Kuznetsov, O., Frontoni, E., Kryvinska, N., Chevardin, V., Smirnov, O. «Wireless Network Encryption Stream Ciphers, Computational Modeling, and Security Analysis». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 379–402.

18. Kuznetsov, O., Frontoni, E., Kryvinska, N., Smirnov, O., Imoize, G.L. «Computational Modeling of Enhanced Spread Spectrum Codes for Asynchronous

Wireless Communication». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 403–447.

19. Ткаченко, О., Ільєнко, А., Улічев, О., Мелешко, Є., Смірнов, О. «Правові засади поширення інформаційних впливів в соціальних мережах». *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»,* 2024. № 2(26), С. 170–188.

20. Смірнова Т.В., Конопліцька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка.* 2024. №4(24), С. 6-27.

21. Вінтенко, Б., Миронець, І., Смірнов, О., Кравчук, О., Козірова, Н., Савеленко, Г., Коваленко, А. «Дослідження вимог та аналіз кібербезпеки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Кібербезпека: освіта, наука, техніка.* 2024. №3(23), С. 111-131.

22. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». *Підводні технології,* 2024, № 13, с. 28-35.

23. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.

24. Akhalaia, G., Iavich, M., Iashvili, G., Prysiazhnyy, D., Smirnova, T. «Secure Encrypted Connection on Georgian Website». *CEUR Workshop Proceedings,* 2023, 3550, pp. 313-320.

25. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56

26. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings,* Volume 3530, 2023, pp. 256-265.

27. Kuznetsov, O., Kandiy, S., Frontoni, E., Smirnov, O. «Trade-offs in Post-Quantum Cryptography: A Comparative Assessment of BIKE, HQC, and Classic McEliece». *CEUR Workshop Proceedings,* Volume 3504, 2023, pp. 1-11.

28. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskyi, V., Khorolska, K., Bebeshko, B. «Selection of a Rational Composition of İnformation Protection Means Using a Genetic Algorithm». *In: Rajakumar, G., Du, KL., Vuppalapati, C., Beligiannis, G.N. (eds) Intelligent Communication Technologies and Virtual Mobile Networks. Lecture Notes on Data Engineering and Communications Technologies*, vol 131. 2023. Springer, Singapore. pp. 21-34.

29. Смірнова Т.В., Гнатюк С.О., Бердибаєв Р.Ш., Сидоренко В.М., Жигаревич О.К., «Система корелювання подій та управління інцидентами кібербезпеки на об'єктах критичної інфраструктури». *Кібербезпека: освіта, наука, техніка*, №*3*(19), 2023, С. 176-196.

30. Смірнов О.А. Козлов Я.О., Смірнова Т.В. «Дослідження застосування SIEM-систем для забезпечення кібербезпеки та захисту інформації». *ІІ Міжнародна науково-практична Інтернет-конференція «Інновації та перспективні шляхи нозвитку інформаційних технологій (ІПШРІТ-2023)»* м.Черкаси 6 грудня 2023 року – Черкаси: ЧДТУ.– 2023. – С.251-252.

31. Козлов Я.О., Смірнова Т.В., Смірнов О.А. «Дослідження SIEM-систем для забезпечення кібербезпеки». *VII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології" до 30-ти річчя кафедри кібербезпеки та програмного забезпечення,* м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 26.

32. Козлов Я.О., Козірова Н.Л., Смірнов О.А. «Дослідження структури та принципу роботи SIEM-системи». *VII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології" до 30-ти річчя*

кафедри кібербезпеки та програмного забезпечення, м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 59.

33. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А., Коваленко А.С. «Дослідження нормативних документів та галузевих стандартів розробки програмного забезпечення комп'ютерних систем управління АЕС, важливих для безпеки». *Системи управління, навігації та зв'язку*, 2023, вип. 2(72), С. 170-178.

34. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» *CEUR Workshop Proceedings,* Volume 3187, 2022,

35. Smirnov O.A., Al-Oraiqat A.M., Ulichev O.S., Meleshko Ye.V., Al-Rawashdeh H.S., Polishchuk L.I. «Modeling strategies for information influence dissemination in social networks». *Journal of Ambient Intelligence and Humanized Computing* Volume 13, Issue 5. Springer, Cham. 2022, pp. 2463-2477.

36. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління,* № 2(70). 2022. С. 28-37.

37. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку,* 2022, № 3(69). С. 93-98.

38. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»,* № 2 (307). С. 46-52. 2022.

39. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 1(67). С. 84-89.

40. Smirnov O., Kuznetsov A., Zhora V., Onikiychuk A., Pieshkova O. «Hiding Messages in Audio Files Using Direct Spread Spectrum». *11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021,* Cracow, Poland, 22-25 September 2021. P. 414-418

41. Smirnov O., Kuznetsov A., Lokotkova I., Kuznetsova T., Florov S., Lebid O. «Using Orthogonal Signals to Hide Information in Images». *4 IEEE International Conference on Advanced Information and Communication Technologies (AICT) – 2021*, Lviv, Ukraine, September 21-25, 2021. P. 255-260.

42. Smirnov O., Kuznetsov A., Girzheva O., Kiian A., Nakisko O., Kuznetsova T. «Advanced Code-Based Electronic Digital Signature Scheme». *2020 IEEE International Conference on Problems of Infocommunications Science and Technology, PIC S and T 2020,* Kharkiv, 6 October 2020-9 October 2020, P. 358-362.

43. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova K. «Data hiding scheme based on spread sequence addressing». *CEUR Workshop Proceedings* Volume 2805, 2020, Pages *44-58.*

44. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». *International Journal of Computing*; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

45. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings*. Volume 2740, 2020, Pages 102-114.

46. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.

47. Smirnov O., Kuznetsov A., Arischenko A., Chepurko I., Onikiychuk A., Kuznetsova T. «Pseudorandom sequences for spread spectrum image steganography». *CEUR Workshop Proceedings* Volume 2654, 2020, Pages 122-131.

48. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New technique for data hiding in cover images using adaptively generated pseudorandom sequences». *CEUR Workshop Proceedings* Volume 2654, 2020, Pages 1-14.

49. Smirnov O., Lutsenko M., Kuznetsov A., Kiian A., Kuznetsova T., «Biometric cryptosystems: overview, state-of-the-art and perspective directions». *Lecture Notes in Networks and Systems*, vol 152. Springer, Cham. 2021, pp 66-84.

50. Smirnov O., Kuznetsov A., Onikiychuk A., Makushenko T., Anisimova O., Arischenko A. «Adaptive pseudo-random sequence generation for spread spectrum image steganography». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Ukraine, Kyiv, May 14-18. 2020. P. 161-165.

51. Smirnov O., Kuznetsov A., Kiian A., Babenko V., Perevozova I.,Chepurko I. «New Approach to the Implementation of Post-Quantum Digital Signature Scheme». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT),* Ukraine, Kyiv, May 14-18. 2020. P. 166-171.

52. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT),* Ukraine, Kyiv, May 14-18. 2020. P. 172-177.