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 аналізу ризиків безпеки при використані міжмережевих екранів”

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

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

м. Кропивницький

АНОТАЦІЯ

Оніщук В.М. Дослідження та програмна реалізація системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів. 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 аналізу ризиків безпеки при використані міжмережевих екранів.

Метою розробки є дослідження та програмна реалізація системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Об'єктом дослідження є процес інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Предметом дослідження є методи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Методи дослідження базуються на методах аналізу даних, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

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, аналізу ризиків, міжмережеві екрани

ABSTRACT

Onishchuk V.M. Research and software implementation of the system of intelligent analysis of security risks when using firewalls. 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 analysis of security risks when using firewalls.

The purpose of the development is the research and software implementation of the system of intelligent analysis of security risks when using firewalls.

The object of the research is the process of intelligent analysis of security risks when using firewalls.

The subject of the research is the methods of intelligent analysis of security risks when using firewalls.

The research methods are based on data analysis methods, mathematical statistics methods, and software development methods.

The result of the work is the software implementation of the system of intelligent analysis of security risks when using firewalls.

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 user-friendly user interface has been developed. Instructions for working with the software are provided.

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

The program was developed in the Python environment.

Keywords: computer science, risk analysis, firewalls

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ЛОМ	–	локальна обчислювальна мережа
MME	–	міжмережеві екрани
ATM	–	асинхронний режим передачі
BSD	–	адаптована для Internet реалізація операційної системи UNIX
ICMP	–	міжмережевий протокол управляючих повідомлень
IP	–	Internet Protocol – міжмережевий протокол
NFS	–	мережна файлова система
PPP	–	протокол передачі від точки до точки
RFC	–	опис набору протоколів Internet
RPC	–	віддалений виклик процедури
SLIP	–	міжмережевий протокол для послідовного каналу
SMTP	–	Simple Mail Transfer Protocol – простий протокол передачі пошти
TCP	–	Transmission Control Protocol – протокол управління передачею
UDP	–	User Datagram Protocol – протокол користувальницьких датаграм
UNIX	–	багатозадачна операційна система
UTP	–	незахищена вита пара
URL	–	уніфікований покажчик інформаційного ресурсу

ВСТУП

Актуальність теми. У сучасну цифрову епоху як організації, так і окремі особи сильно залежать від веб-додатків для широкого кола діяльності. Однак ця залежність від Інтернету також відкриває можливості для зловмисників використовувати слабкі місця безпеки, присутні в цих додатках. Брандмауери веб-додатків (WAF), як правило, є першою лінією захисту, захищаючи веб-додатки шляхом фільтрації та моніторингу HTTP-трафіку. Однак, якщо ці брандмауери не налаштовані належним чином, зловмисники можуть обійти їх або скомпрометувати. Зростаюча кількість атак, спрямованих на веб-додатки, підкреслює нагальну потребу в підвищенні їхньої безпеки. Дана робота пропонує поглиблений огляд існуючих досліджень з оцінки вразливостей веб-додатків та тестування на проникнення (VAPT).

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

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

– Огляд існуючих систем інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

– Дослідження системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

– Програмна реалізація системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Об'єктом дослідження є процес інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Предметом дослідження є методи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Методи дослідження базуються на методах аналізу даних, методах математичної статистики, методах розробки програмного забезпечення.

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

– Удосконалено метод інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

– Розроблено вітчизняний продукт інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів, 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 аналізу ризиків безпеки при використані міжмережевих екранів.

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 аналізу ризиків безпеки при використані міжмережевих екранів, 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

В останні десятиліття веб-сайти та веб-додатки все більше інтегруються в наше повсякденне життя. Ці платформи дозволяють нам виконувати широкий спектр діяльності, від онлайн-покупок та споживання новин до соціального спілкування тощо. Дослідження Siteefy показує, що станом на кінець 2025 року в Інтернеті було активних понад 200 мільйонів веб-сайтів [1].

Зі зростанням нашої залежності від цих платформ зловмисники сприймають цю тенденцію як можливість для отримання грошової вигоди та інших зловмисних намірів. Зростаюча залежність від веб-додатків генерує величезні обсяги даних, що є вирішальним для створення чудового користувацького досвіду [2].

Однак, хоча ці дані корисні для різних цілей, вони також становлять значні ризики, якщо їх належним чином не захистити. Брандмауери, що служать першою лінією захисту в більшості цифрових систем, часто стають основними цілями кібератак.

Тому забезпечення їхньої безпеки має вирішальне значення. Недавні дослідження показують, що 73% порушень у корпоративному секторі в першу чергу пов'язані з вразливостями в їхніх веб-додатках [3]. Така статистика підкреслює нагальну необхідність захисту веб-додатків від атак.

1.2 Область застосування

Виявлення вразливостей, які можуть використовувати зловмисники, є першим кроком до захисту брандмауерів та веб-додатків. Тестування на проникнення та оцінка вразливостей – це надійні методи виявлення цих

вразливостей, що дозволяє командам безпеки підвищити безпеку цих платформ. Оцінка вразливостей та тестування на проникнення (VAPT) дозволяє підприємствам оцінити свій стан кібербезпеки, виявити вразливості та вжити необхідних заходів для їх усунення, перш ніж зловмисники зможуть ними скористатися.

Впроваджуючи ці проактивні заходи, підприємства можуть захистити себе від атак та уникнути витрат, пов'язаних з кібератаками. Новизна цього дослідження полягає в його всебічному огляді та синтезі інструментів і методів VAPT, пропонуючи унікальну категоризацію на основі оптимальних випадків використання. На відміну від попередніх досліджень, ця робота не лише розглядає існуючі інструменти VAPT, але й інтегрує передові практики та новітні технології, такі як штучний інтелект та машинне навчання, в структуру VAPT.

Ця інтеграція враховує еволюційний характер кіберзагроз та забезпечує перспективний підхід до кібербезпеки. Крім того, ця робота визначає та аналізує поширені проблеми в процесах VAPT, надаючи практичні рекомендації щодо подолання цих проблем.

У дослідженні також пропонується нова структура для безперервного впровадження VAPT, підкреслюючи важливість ітеративного та адаптивного підходу до кібербезпеки. Висвітлюючи ці унікальні аспекти, ця робота має на меті поглибити сучасне розуміння та застосування VAPT, пропонуючи практичні поради та стратегії для підвищення безпеки вебзастосунків та брандмауерів.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів, 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

Мережева безпека в сучасному світі важливіша, ніж будь-коли. Брандмауери необхідні для захисту людей і компаній від онлайн-небезпек. У 2025 році передові системи брандмауерів пропонують розширені функції, більшу надійність і розумніший захист. У цьому розділі розглядається 10 найкращих брандмауерів 2025 року та досліджується, як вони допомагають захистити ваші цифрові активи.

1. Fortinet FortiGate

Fortinet FortiGate – це потужний інструмент у сфері мережевої безпеки, який пропонує розширений захист від загроз, аналіз безпеки на основі штучного інтелекту та централізоване управління. Він ідеально підходить для підприємств, які бажають керувати кількома локаціями.

Основні характеристики:

– Система запобігання вторгненням (IPS).

– Багатохмарна безпека.

– Перевірка SSL

Чому варто обрати саме його: Fortinet FortiGate відомий своєю масштабованістю та високою швидкістю роботи.

2. Серія PA від Palo Alto Networks

Palo Alto Networks залишається головним претендентом зі своїми брандмауерами серії PA, які забезпечують комплексний захист від відомих та невідомих загроз.

Основні характеристики:

– Розвідка про загрози в режимі реального часу.

– Глибоке навчання для виявлення шкідливих програм.

– Розширена фільтрація URL-адрес.

Чому варто обрати саме його: Серія PA легко інтегрується з хмарними та локальними системами, що робить її ідеальною для гібридних налаштувань.

3. Безпечний брандмауер Cisco

Cisco Secure Firewall пропонує надійне рішення як для малого бізнесу, так і для великих підприємств. Він поєднує в собі аналіз загроз із простими інструментами керування.

Основні характеристики:

– Видимість та контроль застосунків.

– Розширений захист від шкідливих програм.

– Безпечний віддалений доступ.

Чому варто обрати саме його: світова репутація Cisco та постійні оновлення роблять її надійним вибором.

4. Шлюз безпеки Check Point Quantum

Шлюз безпеки Check Point Quantum розроблений для зупинки кібератак до їх початку. Його можливості захисту «нульового дня» відрізняють його від конкурентів.

Основні характеристики:

– Технологія пісочниці.

– Автоматизоване запобігання загрозам.

– Високопродуктивний VPN.

Чому варто обрати саме його: широка підтримка та зручний інтерфейс Check Point роблять його чудовим варіантом.

5. **Серія SonicWall NSa**

Брандмауери серії NSa від SonicWall забезпечують безпеку корпоративного рівня, адаптовану для середнього бізнесу. Вони пропонують високу швидкість роботи без шкоди для безпеки.

Основні характеристики:

– Виявлення зашифрованих загроз.

– Глибока перевірка пакетів.

– Безпечна SD-WAN.

Чому варто обрати саме його: SonicWall – це економічно ефективний вибір для компаній, які надають пріоритет бюджету та продуктивності.

6. Брандмауер Sophos XG

Брандмауер Sophos XG спрощує мережеву безпеку завдяки орієнтованому на користувача підходу. Його інтуїтивно зрозуміла панель інструментів дозволяє адміністраторам легко керувати загрозами.

Основні характеристики:

– Синхронізована безпека.

– Вбудована веб-фільтрація.

– Розширений контроль застосунків.

Чому варто обрати саме його: Sophos ідеально підходить для організацій, які шукають простоту без шкоди для функціональності.

7. Серія Juniper Networks SRX

Брандмауери Juniper Networks серії SRX зосереджені на забезпеченні високої продуктивності та надійності. Це чудовий варіант для компаній з високими потребами у пропускній здатності.

Основні характеристики:

– Уніфіковане управління загрозами (UTM).

– Розширений захист від шкідливих програм.

– Захист від DDoS-атак.

Чому варто обрати саме його: Juniper SRX пропонує надійну підтримку для центрів обробки даних та великих підприємств.

8. Брандмауер Barracuda CloudGen

Брандмауер CloudGen від Barracuda поєднує хмарні функції з традиційними функціями брандмауера, що робить його лідером у гібридній мережевій безпеці.

Основні характеристики:

- Інтелектуальне управління дорожнім рухом.
- Хмарно-готова інфраструктура.
- Розвідка про загрози в режимі реального часу.

Чому варто обрати саме його: він ідеально підходить для компаній, які прагнуть захистити як хмарні, так і локальні середовища.

9. Серія Huawei USG

Брандмауери серії USG від Huawei пропонують доступну ціну без шкоди для якості. Їхнє надійне апаратне та програмне забезпечення забезпечує всебічний захист.

Основні характеристики:

- Виявлення вторгнень.
- Антивірус та антиспам.
- Безпечний веб-шлюз.

Чому варто обрати саме це – Huawei – це надійний вибір для компаній, які шукають економічно ефективні рішення.

10. Firebox WatchGuard

WatchGuard Firebox відомий своєю універсальністю та простотою розгортання. Він популярний серед малих та середніх корпорацій.

Основні характеристики:

- Багатофакторна автентифікація.
- Доступ до мережі з нульовим рівнем довіри.
- Покращена фільтрація контенту.

Чому варто обрати саме його: масштабованість WatchGuard гарантує його зростання разом із потребами вашого бізнесу.

Чому варто обрати саме його: масштабованість WatchGuard гарантує його зростання разом із потребами вашого бізнесу.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – це потужна мова програмування, яка проста у вивченні. Він має ефективні структури даних високого рівня та простий, але ефективний підхід до об'єктно-орієнтованого програмування. Елегантний синтаксис і динамічна типізація Python разом з його інтерпретованим характером роблять його ідеальною мовою для створення сценаріїв і швидкої розробки додатків у багатьох сферах на більшості платформ.

Інтерпретатор Python і обширна стандартна бібліотека доступні у вихідному або двійковому вигляді для всіх основних платформ на веб-сайті Python https://www.python.org/ і можуть вільно поширюватися. Цей же сайт також містить дистрибутиви та вказівники на багато безкоштовних сторонніх модулів Python, програм і інструментів, а також додаткову документацію.

Інтерпретатор Python легко розширюється за допомогою нових функцій і типів даних, реалізованих у C або C++ (або інших мовах, які можна викликати з C). Python також підходить як мова розширення для налаштовуваних програм.

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 аналізу ризиків безпеки при використані міжмережевих екранів.

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

Оцінка ризиків – це необхідна діяльність, яку необхідно проводити регулярно, щоб побачити загальний стан навколишнього середовища.

Ця оцінка є важливим процесом, який використовується для виявлення потенційних ризиків або небезпек у певній ситуації чи діяльності, а також для оцінки ймовірності та потенційних наслідків цих ризиків.

Існує кілька причин, чому важливо проводити оцінку ризиків.

Це може включати запобігання нещасним випадкам і травмам, дотримання правил, репутацію тощо.

Що стосується оцінки ризиків брандмауера, то це процес оцінки ефективності заходів безпеки брандмауера організації.

Оцінка ризиків брандмауера включає оцінку конфігурації, правил та політик брандмауера, щоб визначити, чи вони належним чином налаштовані та підтримуються для захисту мережі організації від зовнішніх загроз.

Метою оцінки ризиків брандмауера є виявлення потенційних слабких місць у заходах безпеки брандмауера та рекомендація кроків для покращення безпеки мережі організації.

Opinnate проводить оцінку ризиків брандмауера на основі найкращих галузевих практик, яка охоплює такі теми, як рівень дозволеності правила, використання кластера, використання IPS тощо. Детальні елементи перелічені як категорії тем наступним чином:

- Рівень вседозволеності правила.
- Правила доступу до периметра.
- Стан журналу брандмауера та кожного правила.
- Небезпечний доступ до послуг.

- Конфлікт корпоративної політики.
- Стан кластера брандмауера.
- Використання правила або об'єкта.
- Використання IPS.

Постійне розслідування цих елементів у режимі реального часу надаватиме інформацію про стан ризику в режимі реального часу, яку можна буде постійно переглядати на системних панелях, і відповідно вживати заходів. Крім того, для подальшого аналізу будуть створюватися звіти про оцінку ризиків для кожного брандмауера або системи.

3.2 Розробка структурної схеми

Брандмауер – це пристрій мережевої безпеки, апаратний або програмний, який контролює весь вхідний та вихідний трафік і, на основі визначеного набору правил безпеки, приймає, відхиляє або відкидає цей конкретний трафік. Він діє як охоронець, який допомагає захистити ваш цифровий світ від небажаних відвідувачів та потенційних загроз.

Брандмауер:

– Прийняти: дозволити трафік.

– Відхилити: заблокувати трафік, але відповісти повідомленням «помилка недоступності».

– Відмова: блокування трафіку без відповіді.

Потреба в брандмауері

Брандмауер є важливим, оскільки мережі постійно піддаються впливу як безпечного, так і шкідливого трафіку з Інтернету чи інших мереж. Без брандмауера ваші системи не матимуть захисту від небажаного доступу, шкідливої діяльності чи випадкових витоків даних.

1. Запобігання несанкціонованому доступу

Уявіть, що двері вашого будинку завжди відчинені. Будь-хто, хто переходить повз, може зайти та забрати ваші речі. Брандмауер – це як замкнені двері з охоронцем, які впускають лише перевірених людей і не пускають незнайомців.

2. Блокування шкідливого трафіку

Подумайте про свою поштову скриньку електронної пошти. Без спам-фільтра ви б отримували шахрайські та спам-повідомлення. Брандмауер працює як цей спам-фільтр, він блокує шкідливі дані, перш ніж вони досягнуть вас.

3. Захист конфіденційної інформації

Це як зберігати свій банківський PIN-код у сейфі, а не залишати його на столі, де будь-хто може його побачити. Брандмауер гарантує, що ваші особисті та бізнес-дані залишаться прихованими від кіберзлочинців.

4. Запобігання кібератак

Якщо ви залишите свій автомобіль незамкненим на парковці, злодії можуть його вкрасти. Брандмауер блокує вашу мережу, щоб зловмисники не могли її викрасти.

5. Контроль використання мережі

Так само, як батьки встановлюють батьківський контроль, щоб діти не могли відвідувати небезпечні веб-сайти, брандмауери контролюють, де вашим комп’ютерам дозволено підключатися.

Робота брандмауера

Брандмауер працює як охоронець вашої мережі, стоячи між вашими внутрішніми системами, такими як комп’ютери, сервери та пристрої, та зовнішнім світом, таким як Інтернет чи інші мережі. Він ретельно перевіряє всі дані, що входять або виходять, щоб забезпечити проходження лише безпечного трафіку.

– Коли дані намагаються увійти до вашої мережі або вийти з неї, вони спочатку проходять через брандмауер.

– Брандмауер перевіряє пакети даних (невеликі фрагменти інформації) за допомогою заздалегідь визначених правил.

– Правила можна визначити на брандмауері на основі необхідності та політик безпеки організації.

– Брандмауер дозволяє приймати рішення, такі як: Дозволити → Якщо пакет відповідає правилам безпеки, або Блокувати → Якщо пакет підозрілий, походить з чорного списку джерела або містить шкідливий код.

– Брандмауер реєструє заблокований або незвичний трафік для перевірки командами безпеки.

– Сповіщення можуть надсилатися в режимі реального часу, якщо виявлено серйозну загрозу.

Політика за замовчуванням: Дуже важко чітко охопити всі можливі правила на брандмауері. З цієї причини брандмауер завжди повинен мати політику за замовчуванням. Політика за замовчуванням складається лише з дій (прийняти, відхилити або видалити). Припустимо, що на брандмауері не визначено жодного правила щодо SSH-підключення до сервера. Отже, він дотримуватиметься політики за замовчуванням. Якщо політика за замовчуванням на брандмауері встановлена на прийняття, то будь-який комп'ютер за межами вашого офісу може встановити SSH-підключення до сервера. Тому встановлення політики за замовчуванням як "відхилити" (або "скинути") завжди є гарною практикою.

Типи брандмауерів

1) Розміщення в мережі:

– Брандмауер фільтрації пакетів.

– Брандмауер з перевіркою стану.

– Проксі-брандмауер (рівень програми).

– Шлюз на рівні схеми.

– Брандмауер веб-застосунків (WAF).

– Брандмауер наступного покоління (NGFW).

2) Захищені системи:

– Мережевий брандмауер.

– Брандмауер на базі хоста.

3) Метод фільтрації даних:

– Периметральний брандмауер.

– Внутрішній брандмауер.

– Розподілений брандмауер.

4) Форм-фактори:

– Апаратний брандмауер.

– Програмний брандмауер.

Важливість брандмауерів

Мережевий брандмауер – це ваша перша лінія захисту в кібербезпеці. Він відстежує, фільтрує та контролює дані, що передаються в вашу мережу та з неї.

– Мережі вразливі до будь-якого трафіку, який намагається отримати доступ до ваших систем, незалежно від того, чи є він шкідливим, чи ні. Саме тому вкрай важливо перевіряти весь мережевий трафік.

– Коли ви підключаєте персональні комп’ютери до інших IT-систем або Інтернету, це відкриває багато переваг, таких як співпраця, спільний доступ до ресурсів та творчість. Але це також наражає вашу мережу та пристрої на ризики, такі як злом, крадіжка особистих даних, шкідливе програмне забезпечення та онлайн-шахрайство.

– Як тільки зловмисник знайде вашу мережу, він зможе легко отримати до неї доступ та створювати до неї загрозу, особливо за умови постійного підключення до Інтернету.

– Використання брандмауера є важливим для проактивного захисту від цих ризиків. Він допомагає користувачам захистити свої мережі від найгірших небезпек.

Брандмауер служить бар'єром безпеки для мережі, звужуючи поверхню атаки до однієї точки контакту. Замість того, щоб кожен пристрій у мережі був

підданий доступу до Інтернету, весь трафік спочатку має пройти через брандмауер. Таким чином, брандмауер може фільтрувати та блокувати недозволений трафік, незалежно від того, чи він входить, чи виходить. Крім того, брандмауери допомагають створювати облік спроб підключень, підвищуючи обізнаність про безпеку.

Брандмауери регулюють як вхідний, так і вихідний трафік, захищаючи мережу від:

– Зовнішні загрози , такі як віруси, фішингові електронні листи, атаки типу «відмова в обслуговуванні» (DoS) та бекдори. Брандмауери фільтрують вхідний трафік, запобігаючи несанкціонованому доступу до конфіденційних даних та запобігаючи потенційним зараженням шкідливим програмним забезпеченням.

– Внутрішні загрози, такі як відомі зловмисники або ризиковані програми. Брандмауер може застосовувати правила та політики для обмеження певних типів вихідного трафіку, що допомагає виявляти підозрілу активність та запобігати витоку даних.

Брандмауери можуть захищати від різноманітних загроз, моніторячи та контролюючи вхідний і вихідний мережевий трафік. Ось основні загрози, від яких вони допомагають захиститися:

– Проникнення зловмисників: Брандмауери можуть блокувати підозрілі з’єднання, запобігаючи прослуховуванню та розширеним постійним загрозам (APT).

– Батьківський контроль: Батьки можуть використовувати брандмауери, щоб заборонити своїм дітям доступ до відвертого веб-контенту.

– Обмеження перегляду веб-сторінок на робочому місці: Роботодавці можуть обмежити працівників у використанні мережі компанії для доступу до певних сервісів та веб-сайтів, таких як соціальні мережі.

– Національно контрольована інтрамережа: Уряди можуть блокувати доступ до певного веб-контенту та послуг, які суперечать національній політиці чи цінностям.

Дозволяючи власникам мережі встановлювати певні правила, брандмауери пропонують настроюваний захист для різних сценаріїв, підвищуючи загальну безпеку мережі.

Практики безпеки брандмауера

Тримайте брандмауер увімкненим

Ніколи не вимикайте брандмауер лише для підключення до пристрою чи мережі. Натомість налаштуйте правила брандмауера та додайте довірені пристрої до списку дозволених пристроїв.

Будьте в курсі подій

Регулярно оновлюйте програмне забезпечення брандмауера або операційну систему, щоб виправляти вразливості та запобігати новим загрозам безпеці.

Підключення до VPN

VPN шифрує ваш інтернет-трафік, додаючи ще один рівень захисту поряд із вашим брандмауером. Тільки обов’язково налаштуйте правила брандмауера, якщо виникне конфлікт.

Відхилити невідомі запити

Якщо ви отримаєте підозрілий запит на доступ, негайно його заблокуйте. Розслідуйте це пізніше, перш ніж вносити будь-які постійні зміни.

Додайте додаткові інструменти безпеки

Брандмауери не блокують усі загрози, особливо шкідливі програми, які ви встановлюєте самостійно. Використовуйте перевірене антивірусне або антивірусне програмне забезпечення для повного захисту.

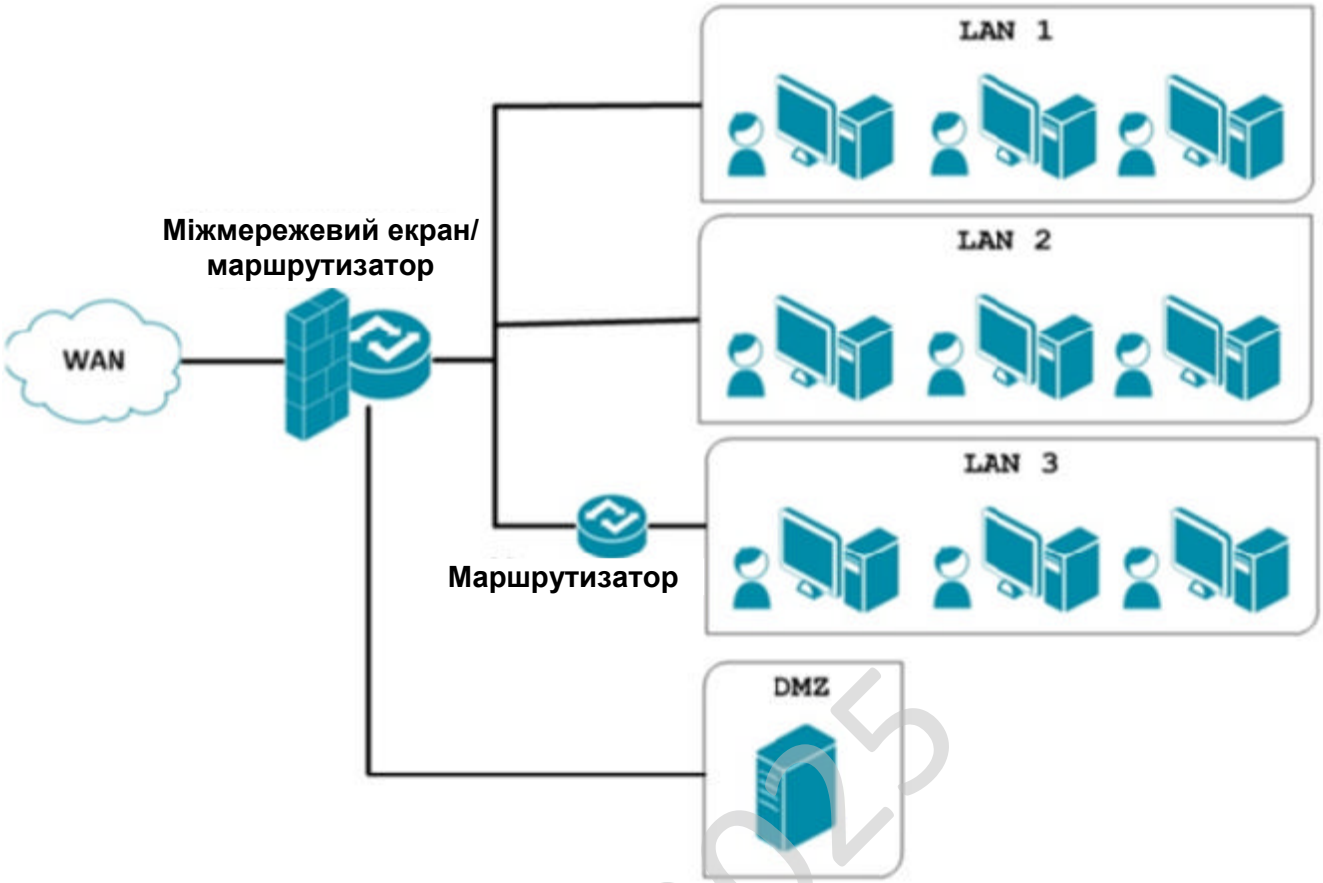

Рисунок 3.1 – Структурна схема системи

Брандмауер – це механізм безпеки, який може бути апаратним або програмним, здебільшого призначений для забезпечення безпеки. Його основна мета – забезпечити доступ до комп’ютера або мережі лише уповноваженим особам. Він виступає посередником між приватною мережею та публічним Інтернетом для регулювання доступу до трафіку відповідно до заданої політики безпеки.

Апаратні брандмауери діють як фізичні бар'єри між мережею та Інтернетом, тоді як програмні брандмауери – це додаткові програми для ПК або мережевих комп'ютерів. Брандмауер можна уявити як охоронця, який стоїть між приватною мережею та публічним Інтернетом і дозволяє перетинати цей бар'єр лише такому трафіку.

Брандмауери функціонують, аналізуючи потоки мережевих пакетів або пакетів даних, що циркулюють у мережі. Вони визначають, чи пропускати пакет чи ні, залежно від встановлених параметрів безпеки. Це може включати аналіз джерела та пункту призначення пакета, використовуваного протоколу зв'язку та вмісту повідомлення.

Існують різні типи брандмауерів, зокрема:

– Фільтрація пакетів Брандмауер.
– Брандмауер проксі-сервісу.
– Брандмауер з перевіркою стану.
– Шлюзи рівня каналу Брандмауер.
– Брандмауер наступного покоління (NGFW).
– Програмні брандмауери.
– Апаратні брандмауери.
– Хмарні брандмауери.

Тепер перейдемо до функцій брандмауера, щоб зрозуміти, як вони допомагають компаніям та окремим особам створювати безперебійні та безпечні мережі.

Головна функція брандмауера – забезпечення безпеки. Ці системи пропонують численні цілі, які безпосередньо покращують безпеку, керованість та продуктивність мережі. Нижче ми пояснили деякі інші функції брандмауера.

Моніторинг мережевого трафіку

Важливою функцією брандмауера є регулювання руху трафіку між приватною мережею та публічним Інтернетом. Брандмауери аналізують кожен пакет інформації, який хоче ввійти або вийти через брандмауер, перевіряючи адресу джерела та призначення, а також використовуваний протокол. Цей пакет порівнюється з попередньо встановленими правилами безпеки; якщо брандмауер схвалює, пакет пропускається; в іншому випадку він блокується.

Контроль доступу

Брандмауери допомагають регулювати, хто може отримувати доступ до вашої мережі та який рівень свободи в ній є. Використовуючи правила доступу, ви можете вирішувати, хто або що може підключатися до вашої мережі. Наприклад, ви можете дозволити переглядати деякі дані лише певним користувачам, а іншим – заборонити.

Контроль доступу описується як список дозволів для вашої мережі, відомий як Список контролю доступу . Так само, як двері будинку зачинені для людей, яких ви не знаєте або не є вашими родичами, брандмауер дозволяє перебувати всередині мережі лише тим, кому дозволено.

Фільтрація пакетів

Фільтрація пакетів використовується в брандмауерах для перевірки мережевого трафіку на рівні пакетів. Брандмауери аналізують пакети, спочатку перевіряючи їх на рівні заголовка, який складається з інформації, що передається з кожним пакетом; він містить IP-адреси джерела та призначення . Потім брандмауери аналізують протокол зв'язку та номери портів, що використовуються.

Представлені тут пакетні дані можуть використовуватися брандмауерами та пов'язаними з ними системами для фільтрації та подальшого вжиття заходів, тобто «дозволу» або «заборони» різних типів трафіку відповідно до політик безпеки. Наприклад, брандмауер можна налаштувати так, щоб він забороняв підключення всього трафіку на стандартному веб-порті 80, щоб запобігти доступу хакерів до хостів на веб-серверах.

Фільтрація на рівні програми

Окрім фільтрації на рівні пакетів, сучасні брандмауери розширюють свою функціональність, фільтруючи на рівні програм. Брандмауери роблять це, розглядаючи вміст повідомлення та контекст мережевого трафіку, а не лише заголовок. Це також дозволяє їм контролювати та забороняти або дозволяти певні

програми чи служби, такі як перегляд веб-сторінок, електронна пошта або протокол передачі файлів .

Фільтрація на рівні програм через брандмауери забезпечує дотримання політик безпеки, таких як запобігання доступу до певних веб-сайтів або використання певних програм.

Фільтрація контенту

Ще однією важливою функцією брандмауерів є те, що їх також можна запрограмувати на фільтрацію контенту на основі різних критеріїв. Ця функція дозволяє організації або окремій особі заборонити будь-якій IP-адресі доступ до певних сайтів або контенту, доступного в Інтернеті та визнаного забороненим або небезпечним.

Фільтрація контенту схожа на батьківський контроль. Так само, як ви б налаштували батьківський контроль на телевізорі чи Інтернеті для своїх дітей, фільтрація контенту робить те саме для всієї мережі.

Система виявлення вторгнень та система запобігання вторгненням

Брандмауери також можуть виконувати роль IDPS, скорочення від intrusion detection and prevention systems (систем виявлення та запобігання вторгненням) . Таким чином, вони можуть шукати ознаки, що вказують на зловмисну активність, таку як мережева атака, спроба несанкціонованого доступу або будь-яка дивна активність у мережі. Коли йдеться про загрози безпеці, брандмауери можуть діяти ефективно та запобігати атаці або пом'якшувати її наслідки, якщо така є.

Ця функція виявлення та запобігання вторгненням допомагає захистити мережу та підключені до неї пристрої від багатьох видів кіберзагроз, таких як DDoS, мережеве шкідливе програмне забезпечення та незаконний доступ.

Підтримка віртуальної приватної мережі (VPN)

Сучасні брандмауери пропонують VPN- рішення як у своєму списку функцій, так і в основному наборі функцій. У випадку VPN-з'єднання, формування безпечного та зашифрованого тунелю між користувачем та мережею

здійснюється за допомогою VPN, щоб користувач міг безпечно підключатися до мережі віддалено.

Брандмауери, що підтримують VPN, можуть допомогти організації розширити периметр своєї мережі та задовольнити потреби користувачів, яким доводиться або які бажають працювати віддалено в сучасному бізнес-середовищі.

Трансляція мережевих адрес (NAT)

Ще однією важливою функцією брандмауерів є те, що вони містять функції NAT, які дозволяють їм перетворювати одну схему мережевих адрес в іншу. Це особливо актуально, коли приватна мережа використовує інший набір IP-адрес, ніж публічна мережа.

Коли брандмауери виконують NAT, внутрішня фактична топологія мережі залишається прихованою від глобальної мережі, що робить її безпечнішою. Це може допомогти мінімізувати прямий доступ до внутрішніх пристроїв з Інтернету, тим самим зменшуючи ймовірність атаки.

Ведення журналів та звітність

Брандмауери зазвичай мають функції реєстрації та звітності, що дозволяють їм записувати та аналізувати мережевий трафік і події безпеки. Цю інформацію можна використовувати для різних цілей, таких як:

– Моніторинг та аудит мережевої активності.

– Виявлення та розслідування інцидентів безпеки.

– Створення звітів для цілей дотримання вимог та регулювання.

– Оптимізація конфігурацій брандмауера та політик безпеки.

Ось деякі з основних функцій брандмауера, які допомагають захистити вашу приватну мережу та пристрої.

3.3 Розробка функціональної схеми

На рисунку 3.2 зображена функціональна схема системи. У багатьох пристроях для домашніх мереж, наприклад інтегрованих маршрутизаторах, часто

є багатофункціональні програмні міжмережеві екрани. Такі міжмережеві екрани звичайно реалізують трансляцію мережних адрес (NAT), динамічний аналіз пакетів (SPI), а також фільтрацію по IP-адресах, додатках і веб-сайтах. Додатково вони підтримують функції DMZ.

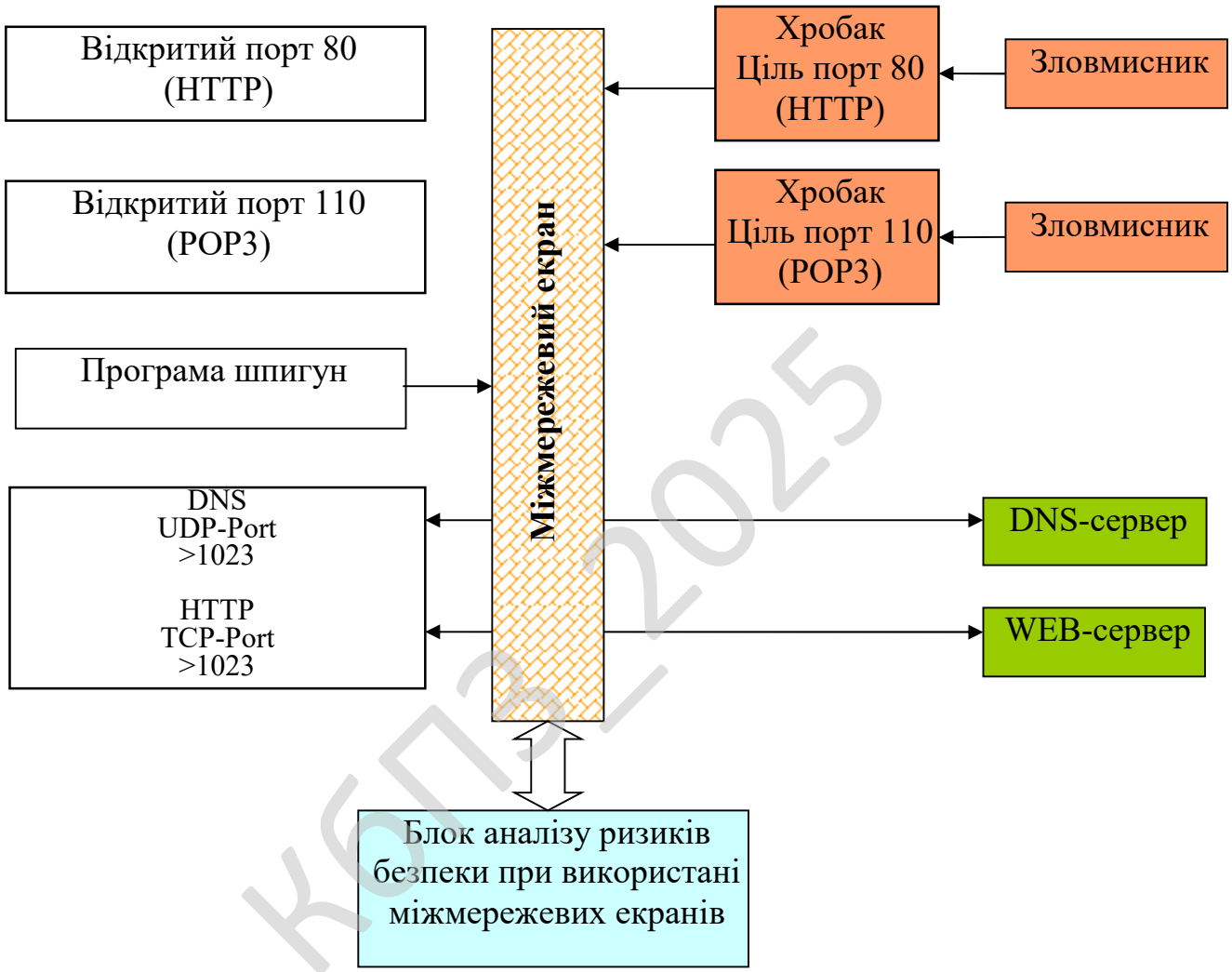

Рисунок 3.2 – Функціональна схема системи

Інтегрований маршрутизатор дозволяє настроїти примітивну DMZ для доступу до внутрішнього сервера з вузлів за межами мережі. Для цього сервер повинен мати статичну IP-адресу, що вказується в конфігурації DMZ. Інтегрований маршрутизатор ізолює трафік, що пересилається на зазначений IP-адрес. Цей трафік пропускається тільки на той порт комутатора, до якого

підключений сервер. На всі інші вузли як і раніше поширюється захист міжмережевого екрану.

При активації DMZ у найпростішому виді зовнішні вузли одержують доступ до всіх портів сервера, наприклад 80 (HTTP), 21 (FTP) і 110 (POP3 для електронної пошти).

Функція переадресації портів дозволяє настроїти більш строгу конфігурацію DMZ. У цьому випадку вказуються порти, які повинні бути доступні на сервері. Пропускається тільки трафік, спрямований на ці порти. Весь інший трафік виключається.

Бездротова точка доступу в складі інтегрованого маршрутизатора часто вважається частиною внутрішньої мережі. Необхідно усвідомлювати, що при роботі точки доступу в незахищеному режимі всі користувачі, що підключилися до неї, одержують доступ до внутрішньої захищеної мережі без проходження міжмережевого екрану. Зловмисники можуть у такий спосіб одержати доступ до внутрішньої мережі, минаючи всі засоби захисту.

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.

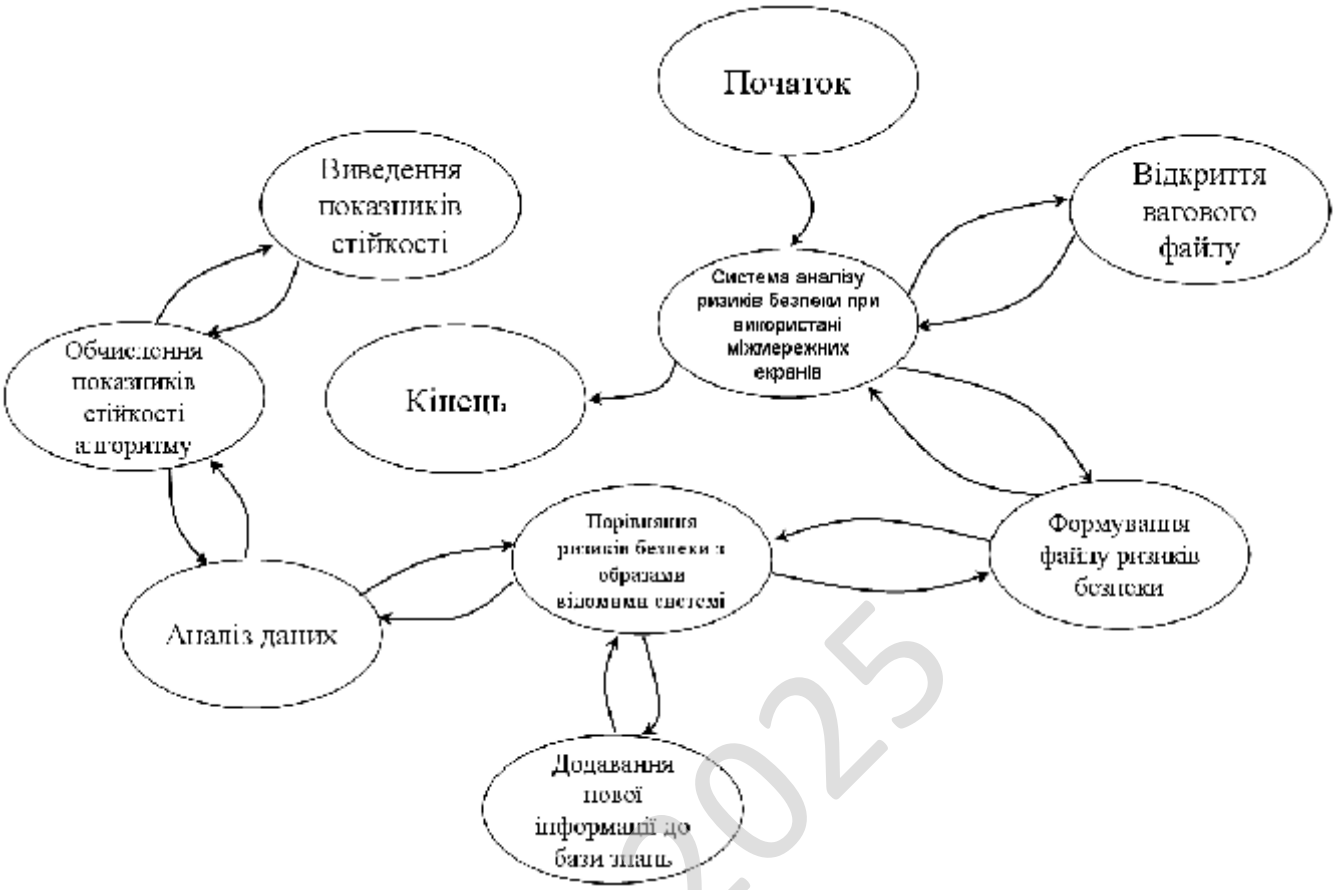

Рисунок 3.3 – Діаграма взаємодії процесів

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.

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 РОБОТИ. 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 системи аналізу ризиків безпеки при використані міжмережевих екранів.

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

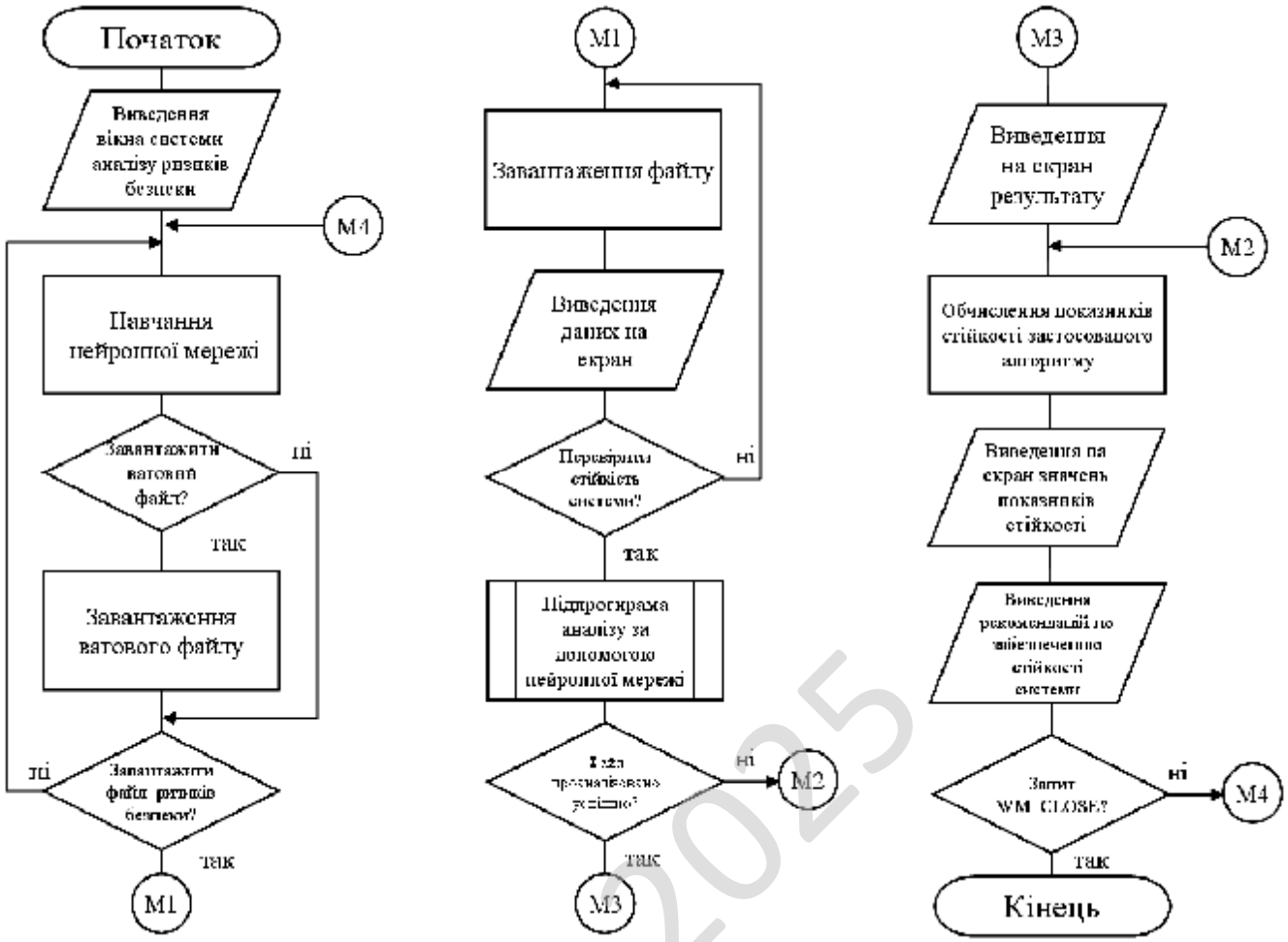

Рисунок 4.1 – Блок-схема основної програми

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 магістерської роботи було використано наступні підходи UML: діаграма діяльності (діаграми поведінки типу); діаграма прецедентів (діаграми поведінки типу); Діаграма класів; Діаграма компонент; Діаграма об'єктів; Діаграма розгортання.

Діаграма діяльності. Це візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій. Дія є фундаментальною одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних сигналів.

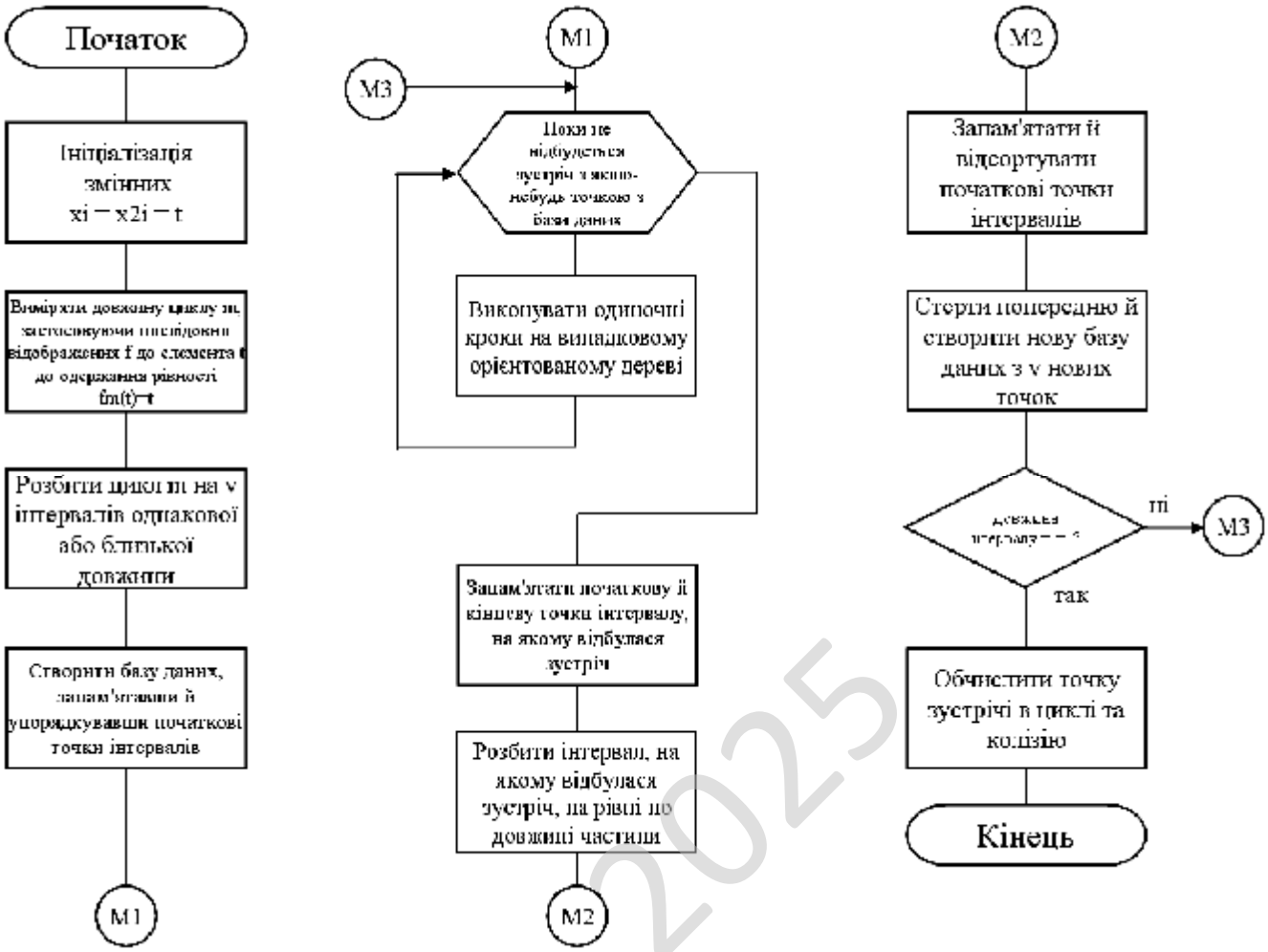

Рисунок 4.2 – Блок-схема роботи підпрограми

Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності. Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності.

Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку.

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.

Опис системи

Система інтелектуального аналізу ризиків безпеки при використанні міжмережевих екранів реалізується на мові Python як модульний програмний комплекс. Вона підтримує читання журналів подій міжмережевого екрана, завантаження поточної конфігурації правил, формування узагальнених показників та побудову інтелектуальної оцінки ризику.

Система призначається для використання в магістерській випускній кваліфікаційній роботі як інструмент дослідження поведінки міжмережевого екрана та обраних політик фільтрації трафіку.

Основою реалізації є набір моделей предметної області. Створюється клас FirewallRule, який описує правило міжмережевого екрана. Кожне правило має ідентифікатор, опис джерела і призначення трафіку, порти, протокол, дію дозволити або заборонити, а також перелік міток.

Мітки застосовуються для логічної класифікації правил, наприклад зовнішній периметр, демілітаризована зона, доступ до критичних сервісів. Для опису реальних подій створюється клас FirewallLogRecord, який зберігає час події, IP адреси, порти, протокол, дію міжмережевого екрана, обсяг переданих даних та ідентифікатор правила, яке спрацьовує.

Таке подання дає змогу будувати як статистичні, так і евристичні оцінки.

Для відображення результатів аналізу створюється модель RiskFinding. Вона описує окреме виявлене відхилення або небажану ситуацію. У записі зберігається унікальний код фактора ризику, назва, текстовий опис, числовий бал ризику, рівень важливості, а також посилання на правила або журнальні записи які породжують цей ризик.

Для підсумкової оцінки використовується клас RiskSummary. Він накопичує загальну кількість знайдених ризиків, середній та максимальний бал, кількість подій за рівнями важливості, а також інтегральний індекс ризику для всієї конфігурації міжмережевого екрана.

Логіка інтелектуального аналізу концентрується в модулі RiskEngine. Цей компонент приймає на вхід список правил та журналів, після чого послідовно застосовує набір експертних правил.

Окремий клас RiskKnowledgeBase зберігає довідник факторів ризику. Наприклад, фактор надто відкритий доступ описує ситуацію коли правило дозволяє будь який трафік з будь якої адреси у внутрішню мережу.

Інший фактор описує правила які ніколи не спрацьовують згідно журналу,

що вказує на потенційно помилкову конфігурацію. Третій фактор описує надмірну кількість спроб доступу до певного сервісу, що може відповідати розвідці портів або спробам грубої сили.

RiskEngine під час роботи обчислює допоміжну статистику. Для кожного правила система підраховує кількість спрацювань, а також розподіл дій дозволити та заборонити.

Для кожної IP адреси джерела підраховується кількість відхилених з єднань, а також кількість різних портів, до яких виконується доступ. Така статистика дає змогу виявляти вузли які сканують мережу або намагаються обійти політику міжмережевого екрана. Для кожної пари адреса призначення та порт система оцінює обсяг трафіку та співвідношення дозволених і заборонених подій, що дає змогу виявляти сервіс з аномальним навантаженням.

Процес оцінки ризику складається з декількох етапів. Спочатку система виконує аналіз конфігурації. Виявляються правила з надто широким діапазоном адрес або портів.

Якщо правило дозволяє трафік з будь якої адреси у внутрішню мережу та використовує протокол з високими вимогами до захисту, таке правило отримує високий бал ризику. Далі система шукає правила які жодного разу не спрацьовують. Це може означати дублювання більш загальних правил або проектну помилку.

Для таких правил формується окреме сповіщення. Також аналізуються конфлікти між правилами, коли дозволяюче правило перекривається за змістом забороняючим або навпаки.

Після цього RiskEngine переходить до аналізу журналів подій. Для кожної IP адреси джерела система обчислює частку відхилених спроб відносно загальної кількості подій. Якщо ця частка перевищує заданий поріг, фіксується підвищений ризик.

Додатковий аналіз стосується кількості різних портів до яких звертається той самий відправник. Якщо кількість портів значна, система розглядає це як

сканування портів та формує відповідний фактор ризику. Подібні алгоритми також застосовуються до адрес призначення.

На основі цих показників формується зважений бал ризику, який зберігається у відповідних об єктах RiskFinding.

Важливою частиною дослідження є можливість налаштування ваг факторів та порогів. Для цього створюється клас RiskFactorDefinition. Кожен фактор має унікальний код, коротку назву, базову вагу, а також рівень важливості за замовчуванням.

Адміністратор або дослідник може змінювати базову вагу, пороги спрацювання та встановлювати власні пріоритети залежно від політики безпеки організації. RiskKnowledgeBase зберігає всі визначення та надає зручні методи для пошуку та оновлення параметрів. Таким чином система підтримує дослідницькі експерименти та сценарії порівняння різних політик.

Для організації введення даних використовується окремий модуль FirewallLogParser. Він реалізує методи перетворення текстових рядків журналу у структуровані об єкти FirewallLogRecord. Формат журналу задається у вигляді полів час, адреса джерела, адреса призначення, порт джерела, порт призначення, протокол, дія, ідентифікатор правила, обсяг переданих даних.

Парсер перевіряє коректність типів, перетворює рядкові значення на елементи перерахувань протоколу та дії, а також обробляє можливу відсутність деяких полів.

Щоб спростити демонстрацію роботи в межах магістерської роботи, у системі створюється фабрика тестових даних SampleDataFactory. Вона формує невеликий набір правил, що відображають типові приклади конфігурацій міжмережевих екранів.

Також генеруються журнали подій, у яких присутні як нормальні з єднання, так і сценарії які відповідають підвищеним ризикам. Наприклад, можна моделювати зовнішню адресу яка виконує велику кількість спроб доступу до закритих портів, або правило, що дозволяє надто широкий доступ до веб сервера.

Це дає змогу перевірити коректність реалізованих алгоритмів, а також продемонструвати формування звітів.

Формування звітів виконується класом RiskReportBuilder. Він перетворює список окремих факторів ризику на зведену інформацію. Обчислюється інтегральний індекс ризику, який представляє собою середньозважене значення за всіма знайденими факторами.

Окремо підраховується кількість ризиків з низьким, середнім, високим та критичним рівнем. Також формується текстовий звіт, який містить короткий опис системи, узагальнений індекс, таблицю кількостей ризиків та текстові описи найважливіших знахідок. Такий звіт може включатися у пояснювальну записку як результат експериментальної частини.

Керування усім процесом виконується класом RiskAnalysisSystem. Він інкапсулює завантаження даних, виклик RiskEngine, побудову звітів та проведення досліджень. Система надає метод аналізу одиничного сценарію, а також метод послідовного запуску декількох сценаріїв з різними налаштуваннями.

Завдяки цьому дослідник може змінювати ваги факторів, налаштовувати пороги спрацювання та спостерігати, як змінюється інтегральний індекс ризику та розподіл знайдених факторів. У вихідному коді реалізується функція main, яка створює демонстраційний набір даних, виконує аналіз та виводить результат у консоль. Розглянемо опис частини класів.


```
# Система інтелектуального аналізу ризиків безпеки
# при використанні міжмережевих екранів

from __future__ import annotations

import statistics
from dataclasses import dataclass, field
from datetime import datetime
from enum import Enum
from typing import List, Dict, Optional, Tuple, Iterable
```

```
# Перерахування для дій міжмережевого екрана
class Action(Enum):
    ALLOW = "ALLOW"
    DENY = "DENY"

# Перерахування для типів протоколів
class Protocol(Enum):
    TCP = "TCP"
    UDP = "UDP"
    ICMP = "ICMP"
    OTHER = "OTHER"


# Клас описує правило міжмережевого екрана
@dataclass
class FirewallRule:
    rule_id: str
    src: str
    dst: str
    src_port: str
    dst_port: str
    protocol: Protocol
    action: Action
    description: str = ""
    tags: List[str] = field(default_factory=list)

# Клас описує один запис журналу міжмережевого екрана
@dataclass
class FirewallLogRecord:
    timestamp: datetime
    src_ip: str
    dst_ip: str
    src_port: int
    dst_port: int
    protocol: Protocol
    action: Action
    bytes_sent: int
    rule_id: Optional[str] = None
    verdict: str = ""

# Клас описує визначення фактора ризику
@dataclass
```

```
class RiskFactorDefinition:
    code: str
    name: str
    base_weight: float
    default_severity: str
    description: str

# Клас описує окреме знайдене відхилення або ризик
@dataclass
class RiskFinding:
    factor_code: str
    title: str
    description: str
    score: float
    severity: str
    related_rules: List[str] = field(default_factory=list)
    related_ips: List[str] = field(default_factory=list)
    meta: Dict[str, float] = field(default_factory=dict)

# Клас описує підсумкову оцінку ризику
@dataclass
class RiskSummary:
    total_findings: int
    avg_score: float
    max_score: float
    severity_counts: Dict[str, int]
    global_risk_index: float

# Клас зберігає базу знань факторів ризику
class RiskKnowledgeBase:
    def __init__(self) -> None:
        self._factors: Dict[str, RiskFactorDefinition] = {}

    def register_factor(self, factor: RiskFactorDefinition) -> None:
        # Метод реєструє новий фактор ризику у базі знань
        self._factors[factor.code] = factor

    def get_factor(self, code: str) -> Optional[RiskFactorDefinition]:
        # Метод повертає визначення фактора ризику за кодом
        return self._factors.get(code)

    def all_factors(self) -> List[RiskFactorDefinition]:
```

```
        # Метод повертає список усіх зареєстрованих факторів
        return list(self._factors.values())

# Клас реалізує парсер журналів міжмережевих екранів
class FirewallLogParser:
    def parse_line(self, line: str) -> Optional[FirewallLogRecord]:
        # Метод перетворює один текстовий рядок журналу у структурований запис
        parts = [p.strip() for p in line.split(",")]
        if len(parts) < 9:
            return None
        try:
            timestamp = datetime.fromisoformat(parts[0])
        except ValueError:
            return None
        src_ip = parts[1]
        dst_ip = parts[2]
        try:
            src_port = int(parts[3])
            dst_port = int(parts[4])
        except ValueError:
            return None
        protocol = self._parse_protocol(parts[5])
        action = self._parse_action(parts[6])
        rule_id = parts[7] or None
        try:
            bytes_sent = int(parts[8])
        except ValueError:
            bytes_sent = 0
        verdict = parts[9] if len(parts) > 9 else ""
        return FirewallLogRecord(
            timestamp=timestamp,
            src_ip=src_ip,
            dst_ip=dst_ip,
            src_port=src_port,
            dst_port=dst_port,
            protocol=protocol,
            action=action,
            bytes_sent=bytes_sent,
            rule_id=rule_id,
            verdict=verdict,
        )
```

```
    def parse_lines(self, lines: Iterable[str]) -> List[FirewallLogRecord]:
        # Метод обробляє послідовність рядків та повертає список записів журналу
        records: List[FirewallLogRecord] = []
        for line in lines:
            line = line.strip()
            if not line:
                continue
            record = self.parse_line(line)
            if record is not None:
                records.append(record)
        return records

    def _parse_protocol(self, value: str) -> Protocol:
        # Метод перетворює рядкове значення на протокол
        upper = value.upper()
        if upper in ("TCP",):
            return Protocol.TCP
        if upper in ("UDP",):
            return Protocol.UDP
        if upper in ("ICMP",):
            return Protocol.ICMP
        return Protocol.OTHER

    def _parse_action(self, value: str) -> Action:
        # Метод перетворює рядкове значення на дію
        upper = value.upper()
        if upper == "ALLOW":
            return Action.ALLOW
        return Action.DENY

# Клас рахує статистику за журналами
class TrafficStatistics:
    def __init__(self) -> None:
        self.hits_per_rule: Dict[str, int] = {}
        self.hits_per_src: Dict[str, int] = {}
        self.denies_per_src: Dict[str, int] = {}
        self.ports_per_src: Dict[str, set[int]] = {}
        self.hits_per_dst_service: Dict[Tuple[str, int], int] = {}
        self.denies_per_dst_service: Dict[Tuple[str, int], int] = {}

    def update_with_record(self, record: FirewallLogRecord) -> None:
        # Метод оновлює статистику за одним записом журналу
```

```
        if record.rule_id:
            self.hits_per_rule[record.rule_id] =
self.hits_per_rule.get(record.rule_id, 0) + 1
        self.hits_per_src[record.src_ip] = self.hits_per_src.get(record.src_ip, 0)
+ 1
        if record.action == Action.DENY:
            self.denies_per_src[record.src_ip] =
self.denies_per_src.get(record.src_ip, 0) + 1
        if record.src_ip not in self.ports_per_src:
            self.ports_per_src[record.src_ip] = set()
        self.ports_per_src[record.src_ip].add(record.dst_port)
        key = (record.dst_ip, record.dst_port)
        self.hits_per_dst_service[key] = self.hits_per_dst_service.get(key, 0) + 1
        if record.action == Action.DENY:
            self.denies_per_dst_service[key] =
self.denies_per_dst_service.get(key, 0) + 1

    def build_from_logs(self, logs: List[FirewallLogRecord]) -> None:
        # Метод заповнює всі статистики на основі списку журналів
        for record in logs:
            self.update_with_record(record)

# Клас реалізує основний інтелектуальний аналіз ризиків
class RiskEngine:
    def __init__(self, knowledge_base: RiskKnowledgeBase) -> None:
        self.knowledge_base = knowledge_base

    def analyze(self, rules: List[FirewallRule], logs: List[FirewallLogRecord]) ->
List[RiskFinding]:
        # Метод виконує повний аналіз конфігурації та журналів
        stats = TrafficStatistics()
        stats.build_from_logs(logs)
        findings: List[RiskFinding] = []
        findings.extend(self._detect_overly_permissive_rules(rules, stats))
        findings.extend(self._detect_unused_rules(rules, stats))
        findings.extend(self._detect_suspicious_sources(stats))
        findings.extend(self._detect_stressed_services(stats))
        return findings
```

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм Blowfish, який є симетричним алгоритмом шифрування, тобто таким, у якому ключ шифрування дорівнює ключу дешифрування. Він є мережею Фейштеля, у якій кількість ітерацій дорівнює 16. Довжина блоку дорівнює 64 бітам, ключ може мати будь-яку довжину в межах 448 біт. Хоча перед початком будь-якого шифрування виконується складна фаза ініціалізації, саме шифрування даних виконується досить швидко.

Алгоритм призначений в основному для додатків, у яких ключ міняється нечасто, до того ж існує фаза початкового рукостискання, під час якої відбувається автентифікація сторін і узгодження загальних параметрів і секретів. При реалізації на 32-бітних мікропроцесорах з більшим кешем даних Blowfish значно швидше DES.

Алгоритм складається із двох частин: розширення ключа й шифрування даних. Розширення ключа перетворює ключ довжиною, принаймні, 448 біт у кілька масивів підключів загальною довжиною 4168 байт.

В основі алгоритму лежить мережа Фейштеля з 16 ітераціями. Кожна ітерація складається з перестановки, що залежить від ключа, і підстановки, що залежить від ключа й даних. Операціями є XOR і додавання 32-бітних слів.

Blowfish використовує велику кількість підключів. Ці ключі повинні бути обчислені заздалегідь, до початку будь-якого шифрування або дешифрування даних. Елементи алгоритму:

1. P – масив, що складається з вісімнадцяти 32-бітних підключів:

$$P_1, P_2, ..., P_{18}.$$

2. Чотири 32-бітних S-boxes с 256 входами кожний. Перший індекс означає номер S-box, другий індекс – номер входу.

$$S_{1,0}, S_{1,1}, \ldots S_{1,255};$$

$$S_{2,0}, S_{2,1}, \ldots S_{2,255};$$

$$S_{3,0}, S_{3,1}, \ldots S_{3,255};$$

$$S_{4,0}, S_{4,1}, \ldots S_{4,255};$$

Шифрування

Входом є 64-бітний елемент даних X, що ділиться на дві 32-бітні половини, X_l і X_r.

$$X_l = X_l \text{ XOR } P_i$$

$$X_r = F(X_l) \text{ XOR } X_r$$

$$\text{Swap } X_l \text{ and } X_r$$

Функція F

Розділити X_l на чотири 8-бітних елементи A, B, C, D.

$F(X_l) = ((S_{1,A} + S_{2,B} \bmod 2^{32}) \text{ XOR } S_{3,C}) + S_{4,D} \bmod 2^{32}$

Дешифрування відрізняється від шифрування тим, що P_i використовуються у зворотному порядку.

Генерація підключів

Підключи обчислюються з використанням самого алгоритму Blowfish.

1. Ініціалізувати перший P-масив і чотири S-boxes фіксовані рядки.

2. Виконати операцію XOR P_1 з першими 32 бітами ключа, операцію XOR P_2 із другими 32 бітами ключа й т.д. Повторювати цикл доти, поки весь P-масив не буде побітово складний з усіма бітами ключа. Для коротких ключів виконується конкатенація ключа із самим собою.

3. Зашифрувати нульовий рядок алгоритмом Blowfish, використовуючи підключи, описані в пунктах (1) і (2).

4. Замінити P_1 і P_2 виходом, отриманим на кроці (3).

5. Зашифрувати вихід кроку (3), використовуючи алгоритм Blowfish з модифікованими підключами.

6. Замінити P_3 і P_4 виходом, отриманим на кроці (5).

7. Продовжити процес, заміняючи всі елементи P-масиву, а потім всі чотири S-boxes, виходами відповідним чином модифікованого алгоритму Blowfish. Для створення всіх підключів потрібна 521 ітерація.

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено інтерфейс програмного забезпечення, розробленого у результаті виконання магістерської роботи.

Розроблене програмне забезпечення системи аналізу ризиків безпеки при використані міжмережевих екранів складається з наступних функціональних блоків:

– Блоку правил обробки пакетів даних.

– Вікна додавання правил обробки пакетів даних.

– Вікно виведення результату роботи системи.

– Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.

– Функціональних кнопок ПЗ.

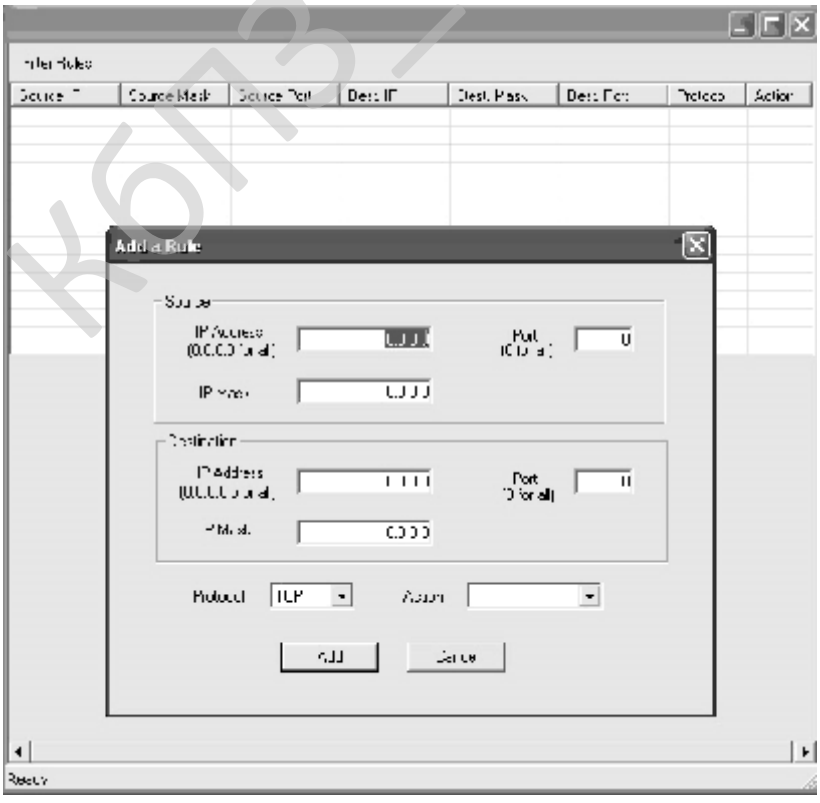

Рисунок 5.1 – Головне вікно розробленого ПЗ

Для перегляду короткої довідки про програму слід натиснути на основному вікні кнопку авторського права, після чого на екрані з’явиться вікно показане на рисунку 5.2.

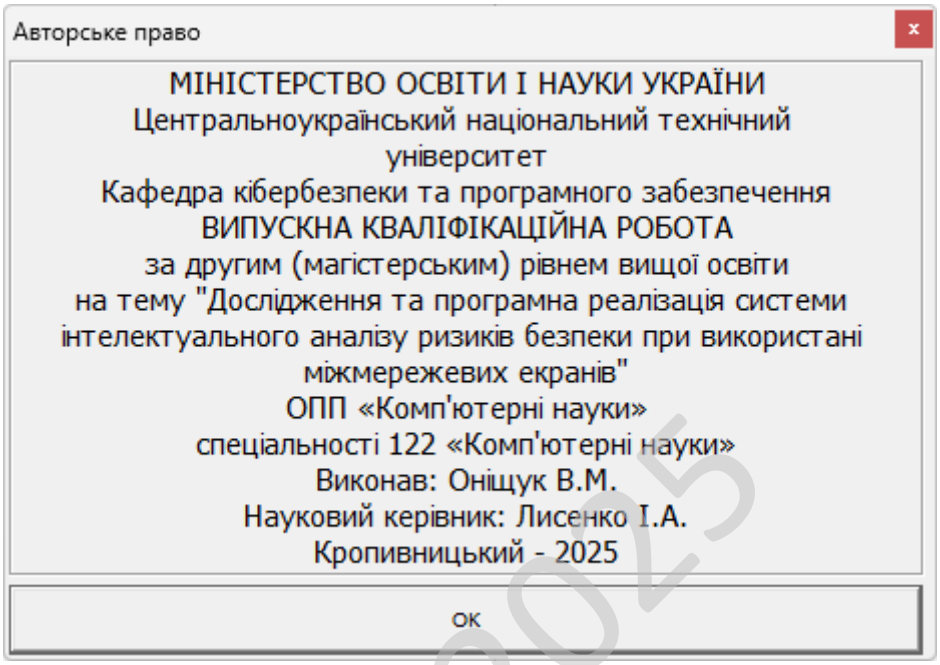

Рисунок 5.2 – Вікно розробника ПЗ

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 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.

Обрано умови розповсюдження – Shareware.

Під умовно-безплатним програмним забезпеченням можна розуміти спосіб або метод розповсюдження комерційного ПЗ на ринку (тобто на шляху до кінцевого користувача), при якому випробувачеві пропонується обмежена за можливостями (не повнофункціональна або демонстраційна версія), терміном дії (тріал версія) або версія з вбудованим набридливим нагадуванням про необхідність оплати використання програми.

В угоді про використання (ліцензії для кінцевого користувача, EULA) також може бути обумовлена заборона на комерційне або професійне (не тестове) її використання.

Основний принцип умовно-безплатного ПЗ – «спробуй, перш ніж купити» (try before you buy). ПЗ що поширюється як умовно-безплатний, надається користувачам безоплатно. Звичайно користувач платить тільки за час завантаження файлів через Інтернет або за носій (CD диск, флешку, ключ). Протягом певного терміну, що становить зазвичай тридцять днів, він може користуватися програмою, тестувати її, освоювати її можливості.

Якщо після закінчення цього терміну користувач вирішить продовжити використання ПЗ, він зобов'язаний купити його (зареєструватися), заплативши авторові певну суму.

В іншому випадку користувач повинен припинити використання ПЗ та видалити його зі свого комп'ютера.

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Метою розробки є дослідження та програмна реалізація системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Об'єктом дослідження є процес інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Предметом дослідження є методи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Методи дослідження базуються на методах аналізу даних, методах математичної статистики, методах розробки програмного забезпечення.

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

– Удосконалено метод інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

– Розроблено вітчизняний продукт інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів, 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 цього дослідження можуть зацікавити насамперед компанії, які мають розгалужену мережеву інфраструктуру і зберігають великі обсяги конфіденційних даних. Для таких організацій, як банки, телекомунікаційні оператори, державні установи, навчальні заклади чи ІТ-компанії, здатність швидко виявляти потенційні ризики безпеки є питанням не лише фінансової стабільності, а й репутації. Інтелектуальний аналіз у поєднанні з міжмережевими екранами дозволяє побудувати систему раннього попередження, яка зменшує ризик витоків інформації або зупинки критичних процесів.

Крім того, така система буде цікава розробникам програмного забезпечення в галузі кібербезпеки, які шукають нові методи автоматизації моніторингу загроз. Для них інтеграція елементів штучного інтелекту в міжмережеві екрани відкриває можливість створення нових продуктів і комерційних рішень. Також проєкт може зацікавити наукові установи, що працюють у сфері машинного навчання, адже він дає змогу застосовувати теоретичні напрацювання на практиці.

Особливу увагу система може викликати серед компаній, які працюють із віддаленими офісами або користуються хмарними сервісами. У таких випадках важливо не лише фільтрувати трафік, а й аналізувати поведінку користувачів і процесів, що відбуваються у мережі. Таким чином, дослідження має міждисциплінарний характер і може знайти практичне застосування у різних секторах економіки.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Для оцінки привабливості такого проєкту доцільно застосувати метод експертного аналізу, який базується на думках фахівців із кібербезпеки, фінансів та управління IT-інфраструктурою. Експертам пропонується оцінити систему за кількома критеріями: технологічна новизна, рівень автоматизації, потенційна економія ресурсів, можливість масштабування та попит на ринку. Кожен критерій оцінюється за шкалою від 1 до 10, після чого визначається середній індекс привабливості.

Наприклад, якщо технологічна інноваційність оцінюється у 9 балів, автоматизація – у 8, а комерційна перспектива – у 10, то середній показник становить близько 9, що свідчить про високий потенціал продукту. Такий результат підтверджує, що система має перспективу для подальшої розробки, комерціалізації та масштабування на інші типи інфраструктур.

Проведення експертних оцінок дозволяє не лише визначити технічну доцільність, але й оцінити стратегічне значення проєкту – наскільки він здатний зменшити ризики, покращити контроль за інформаційними потоками та підвищити рівень кіберстійкості підприємства в цілому.

7.3 Вибір методу оцінки вартості ПЗ

У випадку з таким типом проєктів доцільно застосовувати комбінований підхід – поєднання витратного та порівняльного методів оцінки. Витратний метод дозволяє визначити фактичну вартість створення системи, включаючи розробку програмного забезпечення, закупівлю серверного обладнання, ліцензії, а також навчання персоналу. Порівняльний метод дає змогу врахувати ринкову ситуацію і порівняти вартість рішення з аналогічними системами інших виробників.

Наприклад, якщо подібні рішення провідних компаній, таких як Palo Alto або Fortinet, коштують близько 2 мільйонів гривень, а власна система при цьому має розширену функціональність і нижчу вартість, це створює конкурентну перевагу. У поєднанні ці методи дозволяють досягти об'єктивності – визначити не лише витрати на розробку, а й реальну ринкову ціну, за якою продукт може бути запропонований споживачам.

Такий підхід допомагає розробникам і керівникам ІТ-проєктів приймати економічно обґрунтовані рішення, забезпечуючи баланс між якістю, інноваційністю та прибутковістю проєкту.

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

Підприємство має розгалужену корпоративну мережу з кількома філіями. Безпека побудована на базових міжмережевих екранах, які реагують лише на відомі загрози.

У середньому за рік компанія стикається з 30 інцидентами безпеки, 5 з яких призводять до фінансових збитків.

Мета впровадження інтелектуальної системи – знизити ризики кіберінцидентів і скоротити час реагування на потенційні атаки за рахунок автоматичного аналізу логів, поведінкових моделей і прогнозування загроз. Вхідні дані зафіксовано в таблиці 7.1.

Розрахунок економічного ефекту демонструє наступне: економія на усуненні інцидентів -1 080 000 грн/рік, зниження прямих збитків – 160 000 грн/рік, економія на ІТ-персоналі – 200 000 грн/рік, загальний річний ефект – 1 440 000 грн/рік, чистий ефект – 60 000 грн (окупність < 1 року), термін окупності ≈ 1,04 року, рентабельність інвестицій – 96%.

Таблиця 7.1 – Вихідні дані для розрахунку

Показник	**До впровадження**	**Після впровадження**	**Ефект**
Кількість інцидентів на рік	30	8	–73%
Середня вартість усунення одного інциденту	40 000 грн	15 000 грн	–62%
Прямі збитки від атак (річні)	200 000 грн	40 000 грн	–80%
Витрати на IT-персонал (опрацювання логів і реагування)	600 000 грн	400 000 грн	–200 000 грн
Вартість впровадження системи	—	1 500 000 грн	—
Витрати на обслуговування системи	—	150 000 грн/рік	—

Нефінансові результати впровадження: зменшення людського фактору – автоматизований аналіз зменшує кількість помилкових рішень при оцінці ризиків, покращення рівня захищеності – система виявляє невідомі загрози та аномальні патерни у трафіку, підвищення ефективності реагування – скорочення часу реагування з 4 годин до 30 хвилин, аналітика для управлінських рішень – звіти дозволяють планувати інвестиції у безпеку з урахуванням реальних

ризиків, відповідність міжнародним стандартам (ISO 27001, NIST) – автоматизація ризик-менеджменту покращує аудит безпеки.

Таким чином, система стає не просто технічним інструментом, а елементом корпоративного управління ризиками, який перетворює інформаційну безпеку на інвестицію, а не на витрату.

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Просування проєкту має починатися з підготовки демонстраційної версії системи, яка показує реальні можливості аналізу трафіку, виявлення аномалій та прогнозування атак. Цей прототип стане основою для презентацій перед потенційними клієнтами. Важливо зосередитися на освітньому маркетингу – пояснити підприємствам, чому класичні міжмережеві екрани вже не забезпечують достатній рівень захисту і чому інтелектуальний аналіз ризиків є наступним кроком у розвитку кібербезпеки.

Далі варто представити продукт на галузевих виставках, форумах і конференціях, де збираються фахівці з безпеки та IT-директори. Паралельно потрібно налагодити партнерства з інтеграторами систем безпеки, які можуть пропонувати рішення клієнтам у складі комплексних проєктів.

Ключовим етапом стане запуск онлайн-платформи або сайту продукту з аналітичними кейсами, демонстраційними відео і калькулятором економічного ефекту. Це допоможе зацікавити не лише технічних спеціалістів, а й керівників компаній, які ухвалюють фінансові рішення.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для ефективної реалізації проєкту важливо диверсифікувати канали збуту, поєднуючи прямі продажі з партнерськими. Власна команда продажів може працювати з великими корпоративними клієнтами, тоді як реселери та

інтегратори зосередяться на середньому бізнесі. Також доцільно створити модель “software as a service” (SaaS), коли клієнт сплачує не за весь продукт одразу, а за щомісячне користування. Такий підхід спрощує залучення нових користувачів і підвищує доступність системи. Для просування серед технічної аудиторії можна використовувати професійні форуми, публікації у профільних ЗМІ та участь у вебінарах. Додатково важливо забезпечити можливість інтеграції продукту з популярними рішеннями на ринку – це збільшить довіру клієнтів і спростить впровадження. Таким чином, оптимізація каналів збуту повинна базуватися на принципі “простота доступу + гнучкість використання”, щоб зробити систему привабливою як для великих підприємств, так і для малих організацій, які не мають власних IT-відділів.

7.7 Визначення ключових факторів успіху конкретного проєкту

Ключовими факторами успіху є поєднання технологічної інноваційності та практичної цінності. Система має не лише демонструвати високу точність виявлення загроз, а й бути простою в інтеграції, гнучкою у налаштуванні та зручною у використанні. Важливо, щоб інтерфейс дозволяв адміністраторам швидко отримувати аналітику без потреби у глибоких знаннях машинного навчання. Велике значення має й рівень автоматизації – чим менше ручного втручання потребує система, тим вищою є її ефективність. Також успіх залежить від якості технічної підтримки, оновлень баз даних загроз і постійного розвитку алгоритмів штучного інтелекту. Ще одним вирішальним чинником є довіра користувачів. Якщо система здатна довести свою надійність у реальних умовах, забезпечити стабільність і швидку окупність, вона матиме всі шанси стати конкурентоспроможним рішенням на ринку кібербезпеки. У підсумку успіх визначатиметься тим, наскільки розробка не лише захищає, а й допомагає бізнесу працювати ефективніше та впевненіше в умовах цифрових загроз.

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Стрімкий розвиток галузі інформаційних технологій (IT), який відбувається не тільки у сфері створення програмних продуктів, а й в управлінні виробництвом, банківській системі, бізнесі, системі освіти, на транспорті, у сфері обслуговування призвів до того, що десятки мільйонів людей у всьому світі виявились втягнутими у взаємодію людини з комп'ютером. Природно виникає запитання: настільки безпечною є ця взаємодія для людини? Адже відома аксіома про те, що будь-яка взаємодія людини та засобів праці двостороння.

Впровадження комп'ютерних технологій принципово змінило характер праці різних категорій фахівців. Працівники, використовують комп'ютерну техніку, на своєму досвіді оцінили її величезні можливості. Одночасно виникла певна безтурботність при її експлуатації.

Недотримання вимог безпеки призводить до того, що й через кілька днів роботи за комп'ютером співробітник починає відчувати певний дискомфорт: в нього виникає головний біль і різь у власних очах, з'являються почуття виснаження й дратівливість. В окремих людей порушується сон, погіршується зір, занедужують руки, шия, поперек тощо.

До недоліків умов праці користувачів комп’ютерної техніки можна віднести:

– недостатню площу і обсяг виробничого приміщення;

– недотримання вимог, мікроклімату на робочих місцях;

– низький рівень освітленості у приміщеннях і на робочих поверхнях апаратури;

– підвищений рівень низькочастотних магнітних полів від моніторів;

– порушення вимог організації робочих місць;

– недотримання вимог до режимам праці та відпочинку;

– надмірне виробничу навантаження працівників;

– відсутність навичок зниження впливу психоемоційного напруги.

Відповідно до ст.14 Закону «Про охорони праці» [1] на роботодавця покладено обов'язок забезпечити: безпеку працівників при експлуатації устаткування; застосування коштів індивідуальної захисту працівників; відповідні вимоги охорони праці, умови праці в кожному робоче місце; дотримання режиму праці та відпочинку працівників; навчання безпечним методам і прийомам виконання; інструктаж з охорони праці; організацію контролю над станом умов праці в робочих місць; проведення атестації робочих місць в умовах праці.

Максимально зменшити кількість шкідливих впливів на людину при високій продуктивності праці, створити комфортні умови для роботи людей – ось одна з головних задач охорони праці.

8.2 Аналіз умов праці

Приміщення розташовано на третьому поверсі п'ятиповерхового будинку. У приміщенні розташовано 3 робочих місць з комп'ютерами (далі ПК). Відповідно до норм «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98 [2] площа, що відводиться для робочого місця з комп'ютером повинна бути не менше 6 м^2, об'єм не менше 20 м^3. Розміри даного приміщень складають: довжина – 6 м, ширина – 4,5 м, висота – 3,5 м, тобто загальна фактична площа складає 27 м^2. Необхідна площа на 3 робочих місця із установленими ПК складає 18 м^2, що не перевищує фактичну. Обсяг приміщення, що припадає на одного працюючого, складає 31,5 м^3, отже відповідає нормі ДСанПіН 3.3.2-007-98 – не менше 20 м^3 [2].

При роботі з ПК людина може піддатися впливу шкідливих та небезпечних факторів. Під шкідливими виробничими факторами розуміють фактори, тривалий вплив яких викликає розвиток професійних захворювань. Небезпечні виробничі фактори – вплив яких на працюючого викликає травму, тобто пошкодження організму. Шкідливі і небезпечні чинники, з якими стикається бібліограф при роботі з ПК, приведені в таблиці 8.1.

Таблиця 8.1 – Перелік шкідливих та небезпечних виробничих факторів

Найменування факторів	Можливі джерела їх виникнення	Характер дії
Небезпека ураження електричним струмом	Мережа живлення	Небезпечний
Пожежонебезпечність приміщень	Наявність матеріалів, що згорають і джерел запалення (електроапаратура)	Небезпечний та шкідливий
Іонізація повітря	Статична електрика випромінювання	Шкідливий
Підвищений рівень шуму	Шум створюється перетворювачем напруги ЕОМ, її технічною периферією, а також людьми, що працюють в приміщенні	Шкідливий
Несприятлива освітленість	Недостатнє штучне і природне освітлення	Шкідливий
Незадовільні параметри мікроклімату	Незадовільний стан системи опалення і вентиляції	Шкідливий
Психофізіологічні напруження	Монотонність праці, перенапруженість зорових аналізаторів, розумова напруженість, незручність і статичність пози	Шкідливий

За категорією вибухо- і пожежонебезпеки, дане приміщення відноситься до категорії В – пожежонебезпечне, тому що присутні тверді матеріали, що горять, такі як дерев'яні столи, папір і інше. Виходячи з категорії пожежонебезпеки і поверховості будинку, ступінь вогнестійкості будівлі II. Згідно з ДБН В 1.1-7-2016 «Пожежна безпека об’єктів будівництва» [13] ЕОМ повинні розташовуватись в будівлі не менше ніж II ступню вогнестійкості.

Зв ступенем небезпеки поразки людей електричним струмом відділ, класифікується як приміщення з підвищеною небезпекою, тому що не виключена можливість одночасного дотику людини до маючих з'єднання з землею конструкціям будинку, з одного боку, і до металевих корпусів електроустаткування, що можуть виявити під напругою – з іншого.

Для забезпечення вищевказаних оптимальних метеорологічних умов у помешканні передбачена система опалення (загальне парове) в холодному періоді, та вентиляція і кондиціювання в теплий період року, згідно ДБН2.5–67–2013 «Опалення, вентиляція та кондиціювання» [4]. При виконанні замірів параметрів мікроклімату, значення їх відповідали оптимальним та допустимим параметрам відповідно до ДСанПіН3.3.2.007 – 98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно – обчислювальних машин» [2].

Припустимий рівень іонізації повітря помешкання відповідно до СН 21.52-80 повинен складати 1500 – 3000 один./м^3.

Нормування освітлення здійснюється відповідно до ДБН В.2.5 – 28 – 2006 «Природне та штучне освітлення». [5]

Відділ забезпечений комбінованим освітленням. В темний час доби передбачається загальне і/або місцеве рівномірне штучне, а в світлий – бокове одностороннє природне освітлення два віконних прорізи.

8.3 Техніка безпеки та протипожежна профілактика

Відповідно ДБН В 1.1-7-2016 «Пожежна безпека об’єктів будівництва» [13] будинок можна віднести до II групи по ступені вогнестійкості й до категорії Д по ступені пожежонебезпеки.

Від розподільного щита по праву й ліву сторони встановлені кондиціонери, зовнішня електропроводка, поміщена в ізольований кабель. Висота проводки становить 2,2 м від рівня підлоги, її кріплення здійснюється за допомогою металевих власників. Біля кожного стола організований розподільний щит, розташований на текстолітовій пластинці, закріпленої на стіні на рівні 1м від підлоги. Усього до складу входять п'ять розеток і дві клеми заземлення. Всі обчислювальні машини з'єднані із клемами заземлення. Чотири з п'яти розеток забезпечують подачу напруги 220 В, а одна, забезпечує подачу напруги в 36 В. Про це є відповідні написи на кожному розподільному щиті.

Вимог до пожежної безпеки на підприємстві неухильно повинен дотримуватися кожен співробітник, а організаційна складова при цьому покладається на посадових осіб за відповідним рішенням керівництва і прописується в посадових інструкціях і положеннях по структурним підрозділам.

Зокрема, вказуються конкретні території, ділянки, зони, об’єкти, цілі будівлі і їх частини, поверхи, на яких відповідального співробітника повинне проводити такі організаційні роботи.

Відповідальні особи зобов’язуються розробити, впровадити та підтримувати в певному інструкцією і положенням на ввірених їм об’єктах протипожежний режим і інструкції відповідно до вимог, викладених в нормативних актах.

Передбачено також створення підрозділу добровільної пожежної охорони та пожежно-рятувальної команди в його складі.

Встановлений режим включає порядки з описом місць спеціального призначення та правила їх користування та утримання, наприклад:

– евакуаційних шляхів;

– так званих «курилок»;

– місць складування продукції та сировини;

– стоянки транспорту.

Також встановлюється порядок роботи та технічного обслуговування:

– вентиляційного устаткування;

– засобів пожежогасіння і захисту від загорянь;

– нагрівальних приладів;

– електрообладнання.

Розробляються і впроваджуються правила роботи з відкритим вогнем і горючими матеріалами. Створюються графіки проходження інструктажів з пожежної безпеки співробітників, а також порядок і терміни перевірок знань пожежно-технічного мінімуму, в тому числі, тих працівників, які відповідальні за цю ділянку роботи на підприємстві. При цьому можуть передбачатися внутрішні лекції, семінари, тренінги та практичні заняття на підприємстві, а також зовнішні – на базі спеціалізованих навчальних центрів з професійними викладачами.

Важливою складовою протипожежного режиму на будь-якому об’єкті є розробка і впровадження порядку дій при виникненні пожежі. Неодмінно має бути план евакуації, описано, як повинні відключатися електроустановки, що і в якій послідовності необхідно робити співробітникам.

Відповідно, для кожного об’єкта, кожного приміщення (крім коридорів, санвузлів, басейнів і подібних приміщень), окремих видів робіт складаються інструкції, за якими повинен працювати персонал, залучений на певних ділянках і в виконанні окремих видів робіт. За інструкціями проводиться навчання (інструктаж) персоналу з подальшим контролем знань.

Відповідно НПАОП 40.1-1.21-98 “Правил безпечної експлуатації електроустановок споживачів” [6], приміщення можна віднести до приміщень без підвищеної небезпеки, оскільки це приміщення, сухе, з нормальною

температурою й ізолюючими підлогами, що не має заземлених металоконструкцій.

Персональні ЕОМ можна віднести до першого класу електротехнічних виробів по способі захисту людини від поразки електричним струмом, оскільки їхні корпуси зроблені з ізолюючої пластмаси й кожен пристрій має заземлення. Відповідно правилам пристрою електроустановок ЕОМ можна віднести до електроустановок з робочою напругою до 1000 В.

Однією з достовірних причин пожежі в приміщенні з обчислювальною технікою може бути коротке замикання, що спричиняє спалах електропроводки. Для його попередження вся обчислювальна техніка, а також інші електричні пристрої повинні бути обладнані плавкими запобіжниками, а на вході електромережі повинен бути передбачений автомат захисту. Не слід користуватися електричними подовжувачами й трійниками, що не мають сертифікатів відповідності вимогам безпеки.

Необхідно передбачити наявність у межах досяжності первинних засобів гасіння пожежі (вогнегасників) для локалізації вогню власними засобами до приїзду команди пожежної охорони. Повинен бути розроблений план екстреної евакуації персоналу при виникненні загоряння. Кількість евакуаційних виходів повинне бути не менш двох. Допускається використання одного евакуаційного виходу, якщо відстань найбільш віддаленого робочого місця до цього виходу не перевищує 25 м.

8.4 Розрахункова частина

Для захисного штучного заземлення застосовуються вертикальні електроди: металевий куток 63х63х6 мм, (згідно з ДСТУ 2251-93 «Кутики сталеві гарячекатані рівнополичні. Сортамент») довжиною L=1,6 м., та горизонтальний електрод – металева полоса з перетином 60х5 мм. Напруга – 220/380 В. Розрахункова схема розташування заземлюючих електродів – по контуру

прямокутником (рис. 8.1). Розрахунок проведемо за допустимим опором розтіканню струму заземлювача.

Початкові дані для розрахунку захисного заземлення: тип верхнього шару ґрунту – чорнозем, нижнього шару ґрунту – глина (питомий опір ρ_2 = 40 Ом·м). Умовна товщина верхнього шару ґрунту: H=0,55 м. Відстань між вертикальними заземлювачами (електродами) A=3 м. Глибина закладення горизонтального контура заземлення t=0,6 м. Опір заземлювача, який нормується: $R_{ЗН}$ = 4 Ом. Необхідно визначити необхідну кількість вертикальних заземлювачів та довжину полоси (горизонтального заземлювача).

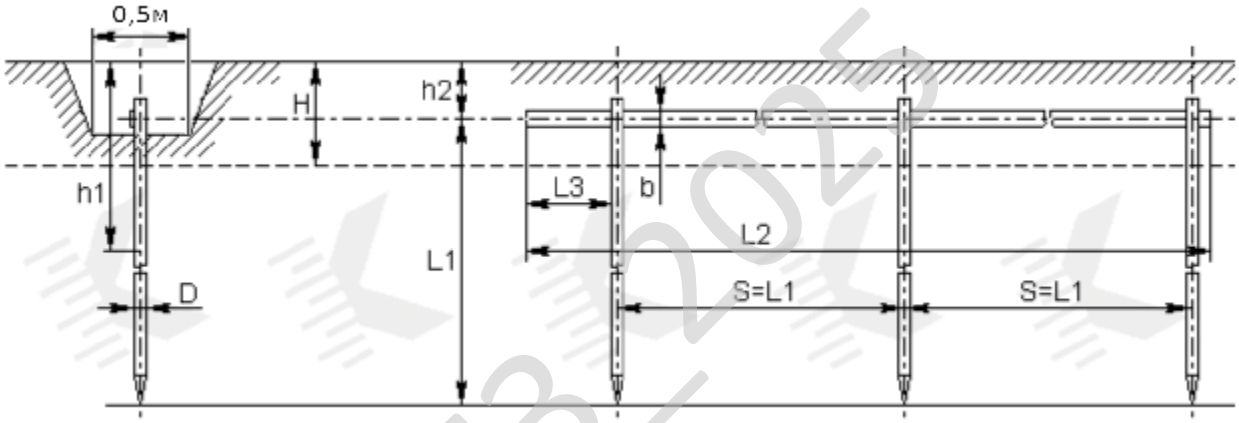

Рисунок 8.1 – Схема штучного заземлення

Виконаємо розрахунок.

Відстань від центра вертикального заземлювача до поверхні землі:

$$T=t + L/2 = 0,6 +1,6/2 =1,4 \text{ м.}$$

Розрахунковий питомий опір ґрунту (з врахуванням того, що фактично вся конструкція заземлювача розташовується у нижньому шарі ґрунту):

$$\rho = \psi \cdot \rho =1,36 \cdot 40=54,5 \text{ Ом·м.}$$

де ψ = 1,36 – табличне значення коефіцієнта сезонності для відповідної кліматичної зони у багатошаровому ґрунті [12]; ρ_2 = 40 Ом·м – табличне значення питомого опору нижнього шару ґрунту (глина) [12].

Еквівалентний діаметр вертикального електрода (кутка) [12]:

$$D_в=0,95\cdot К= 0,95\cdot 63 = 59,85 \text{ мм} = 0,0598 \text{ м}.$$

де К= 63 мм – розмір металевого кутка (заданий).

Відношення A/L=3/1,6 = 1,87.

Опір розтіканню електричного струму одного електрода вертикального заземлювача з урахуванням заглиблення заземлювача [12]:

$$R_О=0,366\cdot(\rho/L)\cdot[\lg(2L/D_в)+(1/2)\cdot\lg((4T+L)/(4T-L))]=$$

$$=0,366\cdot(54,5/1,6)\cdot[(\lg(2\cdot1,6/0,0598)+(1/2)\lg((4\cdot1,4+1,6)/(4\cdot1,4-1,6))]=23 \text{ Ом}.$$

Визначаємо коефіцієнт екранування вертикальних електродів $К_{ев}$=0,62 при орієнтовній кількості вертикальних електродів, яке дорівнює 5 [12].

Визначаємо необхідну кількість вертикальних електродів заземлювача (без врахування горизонтального заземлювача), при $R_{ЗН}$ = 4 Ом:

$$N=R_О /(Кев\ R_{ЗН})= 23 / (0,62\cdot4)= 9,3 \approx 9 \text{ шт}.$$

Визначаємо довжину з'єднуючої полоси:

$$L_П =1,05\cdot A\cdot N= 1,05\cdot3\cdot9=28,35\approx 28 \text{ м}.$$

Опір розтіканню електричного струму з'єднуючої полоси з урахуванням кліматичного коефіцієнта питомого опору ґрунту $К_П$ [12]:

$$R_П =0,366\cdot(\rho\cdot К_П/L_П)\lg(2\cdot L_П^{\,2}/(B\cdot t))=$$

$$=0,366\cdot(40\cdot5/28)\cdot\lg((2\cdot28^{\,2})/(0,06\cdot0,6))=12,2 \text{ Ом}.$$

де $К_П$=5 – табличне значення кліматичного коефіцієнта питомого опору ґрунту для відповідної кліматичної зони для з'єднуючої полоси [12]:

B = 60 мм = 0,06 м – ширина з'єднуючої полоси (задана).

Загальний опір розтіканню електричного струму заземлювача [12]:

$$R =(R_О\cdot R_П)/(R_О\cdot\eta_П + N\cdot R_П\cdot Кев)=$$

$$=(23\cdot12,2)/(23\cdot0,6 + 9\cdot12,2\cdot 0,62) = 3,48 \text{ Ом}.$$

де $\eta_П$ = 0,6 – табличне значення коефіцієнта екранування з'єднуючої полоси [12].

Умова $R \le R_{ЗН}$ виконується ($3,48 \le 4$).

Оскільки при 9 вертикальних електродах R суттєво менше $R_{ЗН}$, зменшимо кількість вертикальних електродів N до 8 і виконаємо перерахунок. У результаті остаточно отримали: R = 3,91 Ом. при кількості вертикальних електродів N=8.

8.5 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 приміщення, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи.

Тільки повна усвідомленість працівника про можливі небезпеки, що можуть підстерігати його на робочому місці та дотримання вимог нормативних актів з питань охорони праці та відповідних рекомендацій фахівців, дозволять значною мірою знизити негативний вплив шкідливих та небезпечних факторів при роботі з комп'ютером на організм людини.

Виконано розрахунок захисного штучного заземлення, як одного з ключових факторів безпеки програміста.

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

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 аналізу ризиків безпеки при використані міжмережевих екранів.

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

– Досліджена система інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

– На основі отриманих результатів досліджень створена програмна реалізація системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів.

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 Blowfish.

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. Оніщук В.М. Дослідження та програмна реалізація системи інтелектуального аналізу ризиків безпеки при використані міжмережевих екранів // Збірник праць молодих науковців ЦНТУ. – Вип. 15. – Кропивницький: ЦНТУ, 2025.

2. Awais Rashid, Howard Chivers, George Danezis, Emil Lupu, Andrew Martin. CyBOK The Cyber Security Body of Knowledge. The National Cyber Security Centre. 2019. 854 p.

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

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

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

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

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

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

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

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

11. 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.

12. 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

13. 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.

14. 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.

15. 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.

16. 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.

17. 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.

18. 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.

19. 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.

20. 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.

21. 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.

22. 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.

23. 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.

24. 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.

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

26. 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

27. 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.

28. 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.

29. 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.

30. 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.

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

32. 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.

33. 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.

34. 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.

35. 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,

36. 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.

37. 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.

38. 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.

39. 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.

40. 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.

41. 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

42. 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.

43. 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.

44. 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.*

45. 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.

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

47. 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.

48. 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.

49. 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.

50. 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.

51. 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.

52. 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.