

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
“___” _____ 2026 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за першим (бакалаврським) рівнем вищої освіти
на тему
“Програмне забезпечення системи Wireless Intrusion Prevention
System”

Виконав здобувач вищої освіти
IV курсу, групи КІ-23-МБ
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Коропець В.С.
«___» _____ 2026 р.

Керівник проекту
кандидат фізико-математичних наук, доцент
_____ Петренюк В.І.
«___» _____ 2026 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Освітній ступінь бакалавр
Галузь знань 12 "Інформаційні технології"
Спеціальність 123 "Комп'ютерна інженерія"
Освітньо-професійна (освітньо-наукова) програма "Комп'ютерна інженерія"

ЗАТВЕРДЖУЮ
Завідувач кафедри
д.т.н., проф.
_____ Олексій СМІРНОВ
« 27 » лютого 2026 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ПЕРШИМ (БАКАЛАВРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Коронцю Владиславу Сергійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи *Програмне забезпечення системи Wireless Intrusion Prevention System*

2. Керівник роботи *Петренюк Володимир Ілліч, кандидат фізико-математичних наук, доцент*

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 126-02 від 27.02.2026 року

3. Строк подання студентом роботи до захисту 23.05.2026 р.

4. Мета та завдання випускної кваліфікаційної роботи: *Метою роботи є розробка програмного забезпечення системи Wireless Intrusion Prevention System*

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Призначення та область використання.

2. Перегляд аналогічних існуючих систем.

3. Опис і обґрунтування проектних рішень.

4. Етапи програмування системи.

5. Впровадження системи в промислову експлуатацію.

6. Висновки

6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Структурна схема системи 1 аркуш

Функціональна схема системи 1 аркуш

Діаграма процесів 1 аркуш

Блок-схема алгоритму роботи додатку 2 аркуша

7. Дата видачі завдання « 27 » лютого 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.03.2026 р.	
2.	Постановка задачі, оформлення ТЗ	15.03.2026 р.	
3.	Розробка моделі компонента	20.03.2026 р.	
4.	Розробка структур даних	25.03.2026 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.03.2026 р.	
6.	Програмування алгоритмів	10.04.2026 р.	
7.	Оформлення ПЗ	17.04.2026 р.	
8.	Попередній захист роботи	23.05.2026 р.	

Дата видачі завдання
« 27 » лютого 2026 р.

Підпис керівника

Петренюк В.І.
(прізвище та ініціали)

Завдання прийнято до виконання
« 27 » лютого 2026 р.

Підпис здобувача

Коропець В.С.
(прізвище та ініціали)

АНОТАЦІЯ

Коропець В.С. Програмне забезпечення системи Wireless Intrusion Prevention System. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2026.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи Wireless Intrusion Prevention System.

Метою розробки є програмне забезпечення системи Wireless Intrusion Prevention System.

Результат роботи – програмна реалізація системи Wireless Intrusion Prevention System.

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'ютерна інженерія, Wireless Intrusion Prevention System

ABSTRACT

**Koropets V.S. Software for the Wireless Intrusion Prevention System.
123 Computer Engineering. Central Ukrainian National Technical University.
Kropyvnytskyi. 2026.**

In this final qualification work for the first (bachelor's) level of higher education, software has been developed that is intended for the Wireless Intrusion Prevention System system.

The purpose of the development is the software for the Wireless Intrusion Prevention System system.

The result of the work is the software implementation of the Wireless Intrusion Prevention System system.

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

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

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

The program was developed in the Python environment.

Keywords: computer engineering, Wireless Intrusion Prevention System

3MIST

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	2
ВСТУП.....	3
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	5
1.1 Призначення системи.....	5
1.2 Область застосування.....	6
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	9
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти.....	9
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	13
2.3 Розгорнута постановка завдання	16
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	17
3.1 Опис функціонування системи	17
3.2 Розробка структурної схеми.....	20
3.3 Розробка функціональної схеми	27
3.4 Розробка діаграми процесів.....	31
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	33
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	33
4.2 Захист розробленого програмного забезпечення.....	56
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	58
6 ОСНОВНІ ВИСНОВКИ.....	61
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	63

					ВКРБ-123.26.0030.00.00.ПЗ				
Вим	Арк.	№ докум.	Підп.	Дата	Програмне забезпечення системи Wireless Intrusion Prevention System	Лім.	Аркуш	Аркушів	
Розроб.		Коропець В.С.				Б		1	70
Перев.		Петренюк В.І.							
Н.контр.		Коваленко А.С.							
Затв.		Смірнов О.А.				ЦНТУ КІ-23-МБ			

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

Ad-hoc	–	режим роботи, у якому пристрій, підключене до LAN, роздає Wi-Fi навколишнім і, як наслідок, надає доступ в LAN
AMs	–	Air Monitors, спеціальні бездротові сенсори
AP	–	звичайна точка доступу
APs	–	Access Points, точки доступу
DoS	–	атака на відмову в обслуговуванні
IPS	–	Intrusion Prevention System, системи виявлення вторгнення
MAC-spoofing	–	підробка адреси відправника повідомлень із метою видати себе за клієнта/AP
RogueAP	–	зловірідна точка доступу
WIDS	–	система виявлення вторгнень
WIPS	–	Wireless Intrusion Prevention System, корпоративні системи виявлення вторгнення по Wi-Fi, які допомагають виявити й блокувати атаки через радіоефір
WLAN	–	Wireless Local Area Network, бездротова локальна мережа

ВСТУП

Актуальність теми. Wireless Intrusion Prevention System (WIPS) – корпоративні системи виявлення вторгнення по Wi-Fi допомагають виявити й блокувати атаки через радіоефір. Перші версії подібних застосунків застосовувалися для моніторингу продуктивності й пошуку неавторизованих точок доступу, але в наступних релізах стали здатні виконувати повноцінне блокування неавторизованих пристроїв і виявлення атак аж до DDoS.

Ідеологія BYOD і доступність Wi-Fi на смартфонах, ноутбуках, планшетах змушує компанії по-іншому дивитися на застосунки для захисту радіоефіру. Служби інформаційної безпеки забороняють використання флешок і знімних дисків, але ніхто не заборонить підключити корпоративний ноутбук до мобільного телефону й злити корпоративну інформацію в хмару. Тому важливо бачити які точки доступу або Wi-Fi пристрої з'являються в периметрі корпоративної мережі й чи правильно себе ведуть. WIPS дозволить надійно контролювати стан радіоефіру й забезпечити ідеальну чистоту в місцях, де це потрібно. Забезпечили там повну неможливість використання чужого Wi-Fi устаткування в периметрі.

Замовники вибирають одну з моделей впровадження Wireless Intrusion Prevention System:

- Точка доступу частина часу працює як сенсор WIPS. Перестає надавати сервіс і збирає статистику.
- Точка доступу обладнана окремим радіо модулем і працює в режимі надання сервісу й у режимі моніторингу безпеки.
- WIPS на базі окремих сенсорів, які не надають ніякого доступу й на 100% контролюють стан радіоефіру й навіть аналізують спектр.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

Перша модель не зовсім логічна, але при цьому коштує не дорого. На виході замовник одержує відсутність і сервісу й системи безпеки. Дві інші моделі найбільше широко поширені й кожна зі своїми перевагами й недоліками.

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи Wireless Intrusion Prevention System.

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

- Огляд існуючих систем Wireless Intrusion Prevention System.
- Дослідження системи Wireless Intrusion Prevention System.
- Програмна реалізація системи Wireless Intrusion Prevention System.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі Wireless Intrusion Prevention System.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи Wireless Intrusion Prevention System, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Глобальний ринок систем виявлення та запобігання бездротовим вторгненням (wipds) у 2026 році становив 2,06 мільярда доларів США, зростаючи до 5,52 мільярда доларів США до 2035 року з розрахунковим середньорічним темпом зростання 13,12% з 2026 по 2035 рік.

Бездротова система виявлення та запобігання вторгненням (WIPDS) – це апаратне або програмне забезпечення, яке розпізнає, коли хтось намагається зламати систему чи мережу. Феноменальна експансія та використання Інтернету викликають нові занепокоєння щодо його безпеки та безпечної передачі цифрової інформації. Для бізнесу, урядів та промисловості в усьому світі вторгнення є серйозною проблемою.

Моніторинг мережевої активності на предмет потенційних нещасних випадків, порушень або неминучих кіберзагроз політиці безпеки організації здійснюється за допомогою систем виявлення та запобігання вторгненням. Процес виявлення вторгнень та подальшого припинення цих випадків відомий як бездротові системи виявлення та запобігання вторгненням (WIPDS). Системи запобігання вторгненням (IPS) та системи виявлення вторгнень (IDS), які розробляються як компонент комп'ютерної мережі для зупинки та виявлення потенційних подій, доступні як ці заходи безпеки. Деякі мережі також використовують IDS як для виявлення проблем політики безпеки, так і для запобігання порушенню політик безпеки користувачами. IDS стала значним доповненням до структур безпеки більшості адміністрацій.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1.2 Область застосування

Використання Інтернету для контрольованих та адміністративно керованих атак на кінцеві точки, IT-інфраструктуру, мережі та дані, які призводять до втрати даних урядами, підприємствами та приватними особами, є прикладом масштабної кібератаки у світовому масштабі. Кіберзлочини трапляються все частіше в результаті прискорення темпів розвитку цифрових комунікацій у глобальній вертикалі. Бездротові системи виявлення та запобігання вторгненням (WIPDS) виникають через порушення безпеки корпоративних даних або витік даних.

Сегментація ринку бездротових систем виявлення та запобігання втрученню (WIPDS)

Аналіз за типом

За типом ринок сегментується на бездротові системи виявлення вторгнень (WIDS) та бездротові системи запобігання вторгненням (WIPS).

За допомогою аналізу додатків

Залежно від застосування, ринок класифікується на фінанси, уряд, IT та телекомунікації, охорону здоров'я, комунальні послуги та освітні установи.

Рушійні фактори

Розширення цифровізації для прискореного зростання ринку

Використання передових технологій, таких як хмарні обчислення, штучний інтелект та інтернет речей (IoT), а також зростаюча тенденція цифровізації, є основними рушійними силами ринкового попиту на системи виявлення та запобігання вторгненням (IDS/IPS). Організації змогли прийняти IDS/IPS наступного покоління через низку недоліків безпеки та низьку IT-інфраструктуру, що стимулює розширення ринку. Стратегію розробки нових продуктів та географічного зростання компанії обрали для сприяння ринковим інноваціям та обслуговування ширшої клієнтської бази.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

Розширення впровадження технологій у різних секторах для стимулювання зростання ринку

Через занепокоєння щодо втрати офіційних даних та компрометації будь-якої інформації, системи виявлення вторгнень стають дедалі більш затребуваними в різних галузях промисловості, включаючи телекомунікації, BFSI та оборону. Оскільки вони стежать за вашою комп'ютерною системою та виявляють її небезпеки протягом очікуваного періоду, впровадження систем виявлення вторгнень зростає. Системи виявлення вторгнень на основі хостів та мережеві системи входять до складу двох різних типів систем виявлення, що складають систему.

Стримуючі фактори

Зростання кількості випадків кіберзагроз, що перешкоджають розширенню ринку

Кіберзагрози поширюються швидше, ніж будь-коли раніше, і компанії в усьому світі стикаються з тиском у пошуку блискучих фахівців, які можуть допомогти їм зменшити ці ризики. Організаціям важко задовольнити свої мінливі вимоги до ІТ-безпеки, оскільки не вистачає кваліфікованих фахівців з безпеки. Тому здатність світового ринку до зростання обмежується браком кваліфікованого персоналу з безпеки. Очікується, що ці фактори перешкоджатимуть зростанню ринку систем бездротового виявлення та запобігання вторгненням (WIDPS).

Північна Америка лідирує на світовому ринку завдяки існуванню країн, що підтримують стабільність

Очікується, що Північна Америка лідируватиме на світовому ринку завдяки добре розвиненим та зручним для обслуговування країнам, що дозволяє регіону отримувати вигоду від дослідницької та розробницької діяльності. Регіон зосереджений на забезпеченні кібербезпеки та отриманні більшого бюджету для боротьби із загрозами конфіденційності.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

Очікується, що Азіатсько-Тихоокеанський регіон продемонструє значне зростання завдяки зростанню доходу на душу населення в поєднанні з вищими показниками крадіжок зі зломом.

Щоб розширити свій асортимент рішень безпеки, компанії, що працюють на ринку систем виявлення/запобігання вторгненням, надають великого значення впровадженню передових технологій для розробки нових продуктів. Підприємства формують стратегічні альянси та співпрацюють, щоб розширити свій міжнародний охоплення та задовольнити потреби конкретних клієнтів. Наприклад, Symantec та NTT Security почали співпрацювати у квітні 2018 року з метою посилення своїх зусиль у сфері кібербезпеки. Завдяки цьому партнерству користувачі локальних, хмарних та мобільних платформ матимуть доступ до розширених можливостей служби безпеки веб-додатків.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи Wireless Intrusion Prevention System, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

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

Якщо потрібно незалежна від виробника точок доступу система WIPS, то краще розглядати модель під номером три й в основі вибору брати до уваги наступні сім критеріїв: керування сенсорами, виявлення атак, відповідність вимогам стандартів, аналіз трафіку, захист від атак, продуктивність і ціна.

Ці критерії не вичерпні, але ці критерії послужать відправною точкою для попереднього відбору застосунку. У рамках цієї роботи ми зрівняємо застосунки від наступних виробників: Mojo Networks (раніше Airtight) WIPS, HP Software (раніше Aruba) Rfprotect, Cisco Adaptive Wireless IPS, NETSCOUT (раніше Fluke Networks) Airmagnet Enterprise, HP Mobility Security IDS/IPS і Zebra Technologies (раніше Motorola) Airdefense.

Критерій #1: Керування сенсорами

Будь-яка система повинна мати зручний механізм керування всіма компонентами. Основним елементом системи WIPS є сенсор або точка доступу в режимі сенсора. Якщо система WIPS того ж виробника, що й Wifi мережа, то оцінювати треба й зручність керування компонентами, що залишилися, – контролери, точки доступу. Але в даному ключі тільки Cisco і Aruba може надати такі можливості. Застосунки Mojo Networks and HP Mobility Security IDS/IPS дозволяють через єдину консоль управляти політиками безпеки бездротових пристроїв.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

General **

AP Name	1240-1
Ethernet MAC	00:1d:45:23:d5:a0
Base Radio MAC	00:1d:46:7e:8a:60
Country Code	US ▼
IP Address	209.165.200.232
Admin Status	☒ Enabled
AP Static IP	☐ Enabled
AP Mode	Monitor ▼
Enhanced WIPS Engine	☒ Enabled
Monitor Mode Optimization	WIPS ▼
AP Failover Priority	Low ▼

273129

Рисунок 2.1 – Переведення точки доступу Cisco у режим WIPS

З досвіду проведення проектів, коли доходить до системи WIPS, то Wifi мережа вже побудована й працює, і система безпеки ставиться поверх неї. Тому є сенс вибирати універсальний застосунок, який зможе забезпечити безпеку мережі, побудованої на будь-якому виробнику.

Критерій #2: Виявлення атак

Усі виробники WIPS виявляють атаки на бездротові пристрої й наявність неавторизованих пристроїв у периметрі. Найбільш активні виробники, з маркетингової точки зору, пишуть про здатність розуміти більш 100 атак по сигнатурах, але ніхто з них на жодному ресурсі не приводить список таких сигнатур або навіть атак. Що робить порівняння застосунків за даним критерієм умовним. І тільки пробне тестування дозволить зрозуміти, що є можливістю, а що маркетингом і спробою виділитися.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

Усі виробники мають можливість виявити неавторизовану точку доступу або пристрій, а також побачити підключення до нього, а також побачити Ad-hoc підключення. І це база, яка в сучасних умовах не влаштовує жодного замовника. Усі застосунки в огляді, крім Zebra Technologies Airdefense (недостатньо інформації у відкритих джерелах) можуть:

- виявити DDoS (denial-of-service attacks);
- виявити «атаку посередника» («людей посередині», man-in-the-middle);
- відобразити на місцевості фізичне розташування бездротових пристроїв, як точок доступу, так і клієнтів.

Два застосунки, Cisco Adaptive Wireless IPS, NETSCOUT Airmagnet Enterprise і HP Mobility Security IDS/IPS, можуть виявляти спроби злому шифрування й автентифікації, що робить ці застосунки найбільш відповідаючих вимогам даного критерію.

Критерій #3: Відповідність вимогам стандартів

Важливо, щоб система WIPS уміла на базі накопиченої статистики становити звіти на відповідність міжнародним стандартам в області безпеки. Це дозволить заощаджувати час для фахівців і мати актуальний звіт для внутрішнього або зовнішнього аудита.

Розглянуті системи створюють звіти в автоматичному режимі або на вимогу. П'ять систем: Aruba Rfprotect, Cisco Adaptive Wireless IPS, NETSCOUT Airmagnet Enterprise, HP Mobility Security IDS/IPS і Zebra Technologies Airdefense – активно позиціонуються для створення звітів на відповідність стандартів в області фінансів і охорони здоров'я. При виборі застосунку звертайте увагу на гнучкість створення звітів і можливість внесення власних заголовків, контактів або логотипів організації.

Критерій #4: Аналіз трафіку

На жаль, у технічних описах такий функціонал відсутній у більшості виробників. Але цей функціонал не менш важливий, чому виявлення атак. Тому що два застосунки Cisco Adaptive Wireless IPS і NETSCOUT Airmagnet Enterprise

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

породжені в одних стінах, то дозволяють захоплювати бездротової трафік і далі розшифрувати його вбудованим декодувальником або аналізатором протоколів Wireshark. Захоплений трафік корисний при аналізі атак, виявлення джерела й характеру поширення. Застосунки Airtight WIPS і Zebra Technologies Airdefense записують події, які виявлені в радіоефірі й надалі звістці балка, але це не повноцінний аналіз трафіку з метою пошуку аномалій.

Критерій #5: Захист від атак

Кожний застосунок WIPS готова захистити бездротову мережу від атак, інакше це буде бездротова система виявлення вторгнень (WIDS), а не WIPS.

Критерій #6: Продуктивність

Є багато аспектів продуктивності систем запобігання вторгнень для бездротових мереж, але два на наш погляд є найбільш значимими – це масштабованість і висока доступність. З погляду масштабованості кращим є варіант із єдиним сервером керування, який може підтримувати велика кількість сенсорів. Усі ліцензії повинні встановлюватися на сервер і не впливає, яка кількість сенсорів ви використовуєте у своєму проекті. Наприклад, HP Mobility Security IDS/IPS може підтримувати до 250 сенсорів за допомогою одного сервера керування й до 600 сенсорів за допомогою віртуального сервера керування, NETSCOUT Airmagnet Enterprise підтримує до 1500 сенсорів за допомогою одного сервера, що досить для покриття декількох корпоративних територіально розподілених будинків.

Висока доступність важлива для систем безпеки. Сенсор повинен бути доступний і виконувати свої завдання, не дивлячись на наявність зв'язку із центральним сервером керування. Наприклад, сенсор NETSCOUT Airmagnet Enterprise, одержавши налаштування від сервера керування, далі може працювати автономно й виконувати свої завдання, не маючи зв'язку із сервером керування. Також можуть функціонувати й сенсори компанії HP Mobility Security IDS/IPS.

Деякі застосунки: Aruba Rfprotect, Cisco Adaptive Wireless IPS і NETSCOUT Airmagnet Enterprise – пропонують найвищий ступінь доступності

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

шляхом впровадження резервного сервера керування, і при виході з ладу одного другий підхоплює миттєве керування. А застосунок Aruba Rfprotect має навіть балансувальник навантаження між основним і резервним сервером.

Критерій #7: Ціна

Моделі ціноутворення дуже широко варіюються від застосунку до застосунку. Це зв'язане, багато в чому, з різною побудовою систем WIPS. У деяких виробників це програмні ліцензії для існуючого устаткування – точок доступу. В інших виробників, це виділені сенсори, які відносяться поверх будь-якої мережі й можуть навіть працювати в хмарі, як це пропонує Mojo Networks (Airteght WIPS). Деякі виробники зобов'язують здобувати технічну підтримку на сенсори й сервер, наприклад, NETSCOUT. Це захищає ваші інвестиції, але при цьому робить більш тяжким проект.

При оцінці бездротових систем запобігання вторгнень, запитуйте вартість сервера керування, сенсорів і вартості підтримки. Обов'язкова вона на перший рік чи ні. А також уточнюйте систему ліцензування функціональних можливостей – чи вимагається щорічна плата, щоб усе працювало. Ідеальний застосунок повинний працювати автономно, не вимагаючи постійних щорічних витрат.

Як вибрати ідеальний застосунок?

Відповідь очевидна – краще вибрати незалежну Wireless Intrusion Prevention System, яка працює з будь-якими клієнтами й точками доступу, не вимагає постійних платежів і активно виявляє й блокує існуючі типи атак по повітрю.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Як мова програмування обрана Python. Python – високорівнева мова програмування загального призначення з акцентом на продуктивність розроблювача й читаність коду. Синтаксис ядра Python мінімалістичний. У той

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

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

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

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

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

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

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

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

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

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

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

Python – стабільна й розповсюджена мова. Він використовується в багатьох проектах і в різних якість: як основна мова програмування або для створення розширень і інтеграції застосунків. На Python реалізоване велика кількість проектів, також він активно використовується для створення прототипів майбутніх програм. Python використовується в багатьох великих компаніях.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи Wireless Intrusion Prevention System.

В процесі розробки випускної кваліфікаційної роботи за першим (бакалаврським) 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) сформулювати висновки про виконаний обсяг робіт та одержані результати.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Думаю, кожний мережний інженер за своє життя зустрічався з поняттям IPS/IDS. У сучасних реаліях кожний поважаючий себе міжмережевий екран пропонує дану функціональність. Виробники устаткування голосно стукають себе кулаком у груди й міряються «незалежними» результатами експертиз – чий NGFW краще ловить різних зловредів. Не становить проблеми знайти гарну статтю з рекомендаціями з налаштування даного класу устаткування.

На жаль, з WIPS усі не так райдужно: розумного матеріалу (особливо українською) небагато й в основному це маркетингові брошури, а не опису налаштувань. Документація вендора, звичайно, допомагає, але найчастіше відповіді на, що цікавлять питання перебувають у різних джерелах і щоб догадатися, де шукати, потрібно знати половину відповіді на запитання.

Різні системи WIPS застосовують різні алгоритми виявлення «порушників» і механізми запобігання їх діяльності. Однак, незважаючи на відмінності в методиках, перелік погроз, з якими кожна система намагається боротися, приблизно однаковий (як показує мій особистий досвід, не завжди заявлений функціонал виявляється на 100% ефективний).

– RogueAP – зловідна точка доступу, яка повністю або частково копіює «вид» справжньої (те ж назва SSID або з невеликими відмінностями) і призначена для реалізації атаки Mitm і/або компрометації облікових даних необережного користувача.

– Ad-hoc – режим роботи, у якому пристрій, підключене до LAN, роздає Wi-Fi навколишнім і, як наслідок, надає доступ в LAN.

– DoS – атака на відмову в обслуговуванні. У реаліях бездротових мереж це по більшій частині флуд повідомленнями деавтентифікації користувачів або

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

захламлення радіоефіру пошуками повідомленнями (шумом) з метою погіршення якості роботи мережі.

- MAC-spoofing – підrobка адреси відправника повідомлень із метою видати себе за клієнта/AP.

- BruteForce – добір пароля Wi-Fi шляхом повного перебору/атаки по словникові.

- Атаки на відомі уразливості протоколів шифрування й авторизації – уразливості WEP, WPA, WPS і т.п.

Розроблювальне в даній роботі програмне забезпечення системи Wireless Intrusion Prevention System – «WIPS» – комплексний застосунок по наданню бездротового доступу й організації WIPS.

Система має традиційну архітектуру: точки доступу й Wi-Fi-контролер + система керування WIPS. Надається також хмарний варіант контролера, але не думаю, що дана опція користується попитом у нашій країні.

Точки доступу можуть застосовуватися в 3-х режимах:

- AP – звичайна точка доступу.
- Sensor – точка доступу в режимі WIPS/WIDS.
- Network Detector – точка доступу, що здійснює сканування провідних сегментів..

WIPS Wireless Intrusion Prevention System (WIPS) – віртуалка, сполучає в собі функціонал контролера бездротової мережі й WIPS, ліцензується по кількості підключених точок (у будь-якому режимі). Окрема ліцензія на функціонал WIPS не потрібна.

Принцип роботи

Принцип роботи WIPS досить простий – усі виявлені користувачі й точки доступу класифікуються по 5 категоріях: Authorized, Misconfigured, External, Unknown, Rogue.

- Authorized – довірені точки доступу й клієнти, підключені до них.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

– Misconfigured – точки доступу, які є довіреними, але настроєні некоректно щодо політики IPS.

– External – точки доступу й клієнти, підключені до них, які не є ні довіреними, ні зловливими. Wi-Fi сусідів, у загальному.

– Unknown – точки доступу й клієнти, підключені до них, які ще не пройшли класифікацію або для класифікації яких недостатньо інформації.

– Rogue – зловливі точки доступу й клієнти, підключені до них.

Класифікація здійснюється автоматично, але у випадку невірної класифікації адміністратор системи може вручну привласнити будь-яку категорію (крім Misconfigured).

Відповідно авторизованими вважаються точки доступу, безпосередньо підключені до сервера керування WIPS; точки доступу, підключені до іншого контролера, з яким інтегрована система (про можливості інтеграції розповім трохи пізніше); точки, не підключені до WIPS, але налаштування яких відповідають політиці безпеки IPS.

Misconfigured – точки, налаштування яких не відповідають політиці WIPS (наприклад, точка використовує WPA, а політика WIPS вимагає тільки WPA2-personal). Можуть бути підключені до сервера керування WIPS або до іншого контролера Wi-Fi, з яким інтегрована система.

Зловливими (rogue) вважаються точки доступу, підключені до провідної мережі, яка моніториться засобами WIPS, але, що не є авторизованими (випадок, коли хтось застромив у мережу зайву точку доступу, яка не обслуговується корпоративним контролером безпроводового зв'язку). Також у список зловливих попадають точки, що передають корпоративні SSID (але, що не є авторизованими за критеріями, описаним вище) і SSID, що перебувають у чорному списку.

Ну а всі інші точки класифікуються як зовнішні.

За результатами класифікації до точок і користувачам застосовуються дії, описані в політику WIPS. Політика припускає безперешкодну взаємодію Authorized об'єктів, інші категорії можна обмежувати. Рекомендується

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

заборонити взаємодія с Rogue і Misconfigured об'єктами, але не застосовувати ніяких обмежень до об'єктів групи External.

Слід пам'ятати, що система вкрай «злопам'ятна». Якщо точка або користувач одного разу були прокласифіковані як rogue, то такими вони й залишаться до того моменту, поки адміністратор їх не «виправдає». Тому можлива наступна ситуація: сусіди по бізнес-центру випадково застромили свою точку доступу не в той комутатор у комутаційному приміщенні. Через деякий час вони зрозуміли свою помилку й виправили помилку, але WIPS уже знайшов і прокласифікував їх точку. І почав застосовувати до неї дії, певні політикою – наприклад, глушити спроби підключення до них користувачів. Після цього WIPS перекласифікує інші видимі точки з такими ж параметрами (той же SSID, та ж модель точок і т.п.), що належать сусідському Wi-Fi, і їх чекає та ж доля.

3.2 Розробка структурної схеми

Первісне налаштування

Докладніше зупинюся на процесі установки й налаштування системи.

Оскільки MAC-адреса віртуальної машини може змінюватися при її міграції по середовищу віртуалізації, рекоменду призначити вручну. Якщо адреса зміниться, то ліцензія «злетить» і прийде або просити вендора перевипустити ліцензію на новий MAC, або повернути старий MAC і перезавантажитися.

Уводимо дефолтні логін/пароль config/config і запускаємо візард первісної ініціалізації.

Під час первісної ініціалізації важливо вказати робітник NTP і DNS сервера. Якщо цього не зробити, то може не запуститися веб-інтерфейс і подальше налаштування системи буде неможливе, буде потрібно переналаштування цих параметрів через консоль і перезавантаження. Також необхідно вказати DNS-suffix. Цей параметр потрібний для того, щоб коректно відробив механізм автознаходження сервера точками доступу. Він повинен

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

збігатися на сервері WIPS, на DNS-сервері й на кожній точці доступу. Після цього в нас є готова до налаштування система, але в ній не вистачає найважливішого – точок доступу. Із точками доступу є окремий нюанс: одноразово точка доступу може перебувати тільки в одному з 3-х режимів (AP, Sensor, Network Detector).

Головний підступ полягає в тому, що одночасно передавати інформацію й сканувати ефір, запобігати вторгненням точка не може. Так що для роботи WIPS потрібно планувати розміщення окремих точок, що виконують тільки дане завдання.

Щоб точки доступу автоматично підключилися до сервера WIPS, WIPS потрібно, щоб вони одержали IP-адресу, адресу DNS-сервера й DNS-suffix по DHCP. Також на DNS-сервері повинна бути присутнім запис для імені «wifi-security-server», що вказує на сервер WIPS. У принципі кожную точку можна налаштувати руками, але це не наш метод.

При роботі через міжмережеві екрани необхідно простежити, щоб описані нижче взаємодії були дозволені.

Порт – Використання

– Тср/21 – FTP – передача файлів може використовуватися для завантаження.

– Тср/22 – SSH.

– Тср/25 – Інтеграція з SMTP – використовується для відправлення оповіщень.

– Тср/80 – Завантаження відновлень ОС точками доступу в сервера WIPS.

– Тср/443 – Адмінський GUI.

– Тср/1035 – Взаємодія між серверами WIPS, що працюють у кластері.

– Тср/3851 – Взаємодія з Airtight Mobile – endpoint-клієнт від WIPS Networks.

– Тср/3852 – Open VPN при використанні Cloud Integration Point (CIP).

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

– Tcr/4433 – Використовується при автентифікації клієнтів по сертифікатах/ смарт-картам.

– Tcr/5432 – Взаємодія між Postgresql базами кластера серверів WIPS.

– Tcr/2002 – Порт, використовуваний для траблшутингу (packet capture).

На цьому порту точка доступу очікує команди від сервера на ініціацію сесії збору трафіку.

– Udp/23 – NTP – синхронізація часу (взаємодіють і точки, і сервер).

– Udp/161 – SNMP (приймання повідомлень) – потрібно для інтеграції зі сторонніми контролерами (наприклад Cisco WLC).

– Udp/162 – SNMP – (відправлення повідомлень) – потрібно для інтеграції зі сторонніми контролерами (наприклад, Cisco WLC).

– Udp/389 – LDAP – інтеграція з LDAP для автентифікації користувачів.

– Udp/694 – Heartbeat між серверами кластера WIPS.

– Udp/514 – Syslog – інтеграція із зовнішніми SIEM.

– Udp/1194 – Open VPN між серверами в кластері.

– Udp/1812 – Radius – автентифікація користувачів через radius-сервер.

– Udp/3851 – Обмін інформацією із точками доступу по протоколу Spectratalk.

За замовчуванням точки підключаються до системи в режимі «AP». Після цього їх можна перемкнути в режим, що вимагається. Також можливо налаштувати профіль пристроїв, який буде автоматично перемикає в необхідний режим усі, що підключаються AP.

Але, як водиться, є один нюанс: якщо точка довго не може знайти контролер, то вона від прикrostі впадає в летаргічний сон і лікує цю проблему тільки перезавантаження після попереднього налаштування файрвола/маршрутизації/DNS.

Точки в режимі network detector необхідно налаштовувати вручну через SSH/консольне підключення. Зміст цього режиму в тому, що точка по транку підключається до всіх VLAN ЛОМ і шукає в ній ознаки провідного підключення

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

неавторизованих точок доступу. Якщо знаходить – починає спуфити їх MAC-адреси й усіляко заважати їхньому життю. Така поведінка пристрою може викликати невдоволення комутаторів так що їх теж рекомендується попередньо налаштувати. Саме по собі налаштування network detector досить проста: необхідно змінити режим роботи, налаштувати транк (вказати перелік VLAN), налаштувати параметри мережі для кожного VLAN (задати IP-адреса й маску мережі), вказати, через який VLAN стукатися на сервер WIPS і по якій адресі.

Властиво, етап початкового налаштування системи завершений, коли сервер керування й точки доступу побачили один одного. Я не буду зупинятися докладно на налаштуванні Wi-Fi засобами WIPS – даний процес нічим не відрізняється від налаштування бездротової мережі на устаткуванні будь-якого іншого вендору. Далі я розповім про налаштування сервісів, необхідних для роботи WIPS, і самого WIPS зокрема.

Налаштування локацій

Далі необхідно налаштувати «карту» мережі. Для цього потрібно налаштувати «топологію»: створити спеціальні сутності – «будинку» і «поверхи» (у жодному разі не використовуйте кирилицю – система її не підтримує). Для кожного «поверху» – завантажити план приміщення й відзначити розташування всіх сенсорів і легітимних точок доступу.

Створення елементів топології здійснюється в меню Locations>>Add Location. Завантаження карти – Locations>>Building X> Floor Y>Add layout. При завантаженні карти потрібно вказати довжину й ширину приміщення й вибрати одиниці виміру. На жаль, карта – це просто рисунок (у форматі jpg) без можливості вказати на ньому які-небудь об'єкти (наприклад, стіни).

Додавання сенсора/точки доступу на план здійснюється в меню: Locations>>Building X>Floor Y>Show device list. Зі списку, що з'явився, пристроїв сенсор/точку необхідно «перетягнути» на місце розміщення на карті.

Карта повинна бути максимально точною, як і розміщення на ній сенсорів і точок доступу. Це потрібно для коректної роботи механізму позиціонування.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

Одна галочка відокремлює нас від початку запобігання згідно з настроєними політиками.

Інтеграція з існуючим Wi-Fi

Одним з переваг даної системи є можливість інтеграції зі стороннім Wi-Fi контролером. У такий спосіб можна рознести функціонал – Wi-Fi роздає один пристрій (і адмініструє його 1 людина), а стежить за ним – інше (і відповідно це може бути зоною відповідальності іншої людини).

Підтримується інтеграція з Cisco WLC, Aruba Mobility Controllers і HP MSM Controller. Взаємодія здійснюється по SNMP. Тут нічого налаштовувати особливо не потрібно – необхідно вказати IP-адреса з контролера, вибрати версію SNMP, вказати назва ком'юніти, облікові дані (у випадку Snmpv3) і відзначити галочками, які параметри ми прагнемо одержати від контролера. Для WIPS у принципі досить тільки інформації про AP і клієнтах, але можна позначити галочкою й інформацію про точки, які бачить кожна точка доступу, що обслуговується контролером, з яким ми налаштовуємо взаємодію. Це може пригодитися в наступній ситуації – допустимо контролер, з яким ми інтегруємося, обслуговує одночасно кілька будинків. У будинку «А» у нас є свої сенсори WIPS – вони самі все бачать, і допомога від AP їм не потрібна; а от у будинку «Б» сенсорів у нас ні, і тоді ми можемо одержати інформацію про учасників радіоефіру від контролера. Звичайно, змусити працювати «на придушення» точки, керовані контролером, система не зможе, а от прокласифікувати побачені в такий спосіб – цілком. І якщо в будинку «Б» ми прокласифікуємо даним способом rogue точку, то система згенерує Alarm і адміністратор зможе побачити це в інтерфейсі й вжити заходів. Не автоматично, звичайно, але хоч якось.

У цілому це все по налаштуванню інтеграції. Ще раз уточню: щоб точки, отримані від стороннього контролера Wi-Fi, стали авторизованими, необхідно зробити налаштування IPS, як я писав вище. Якщо цього не зробити, то вони стануть misconfigured. Правда, даний механізм не завжди працює коректно. Наприклад, довелося зіштовхнутися з багом, через який усі точки Cisco 1600,

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

отримані від WLC, чомусь не додавалися автоматично в авторизовані та й взагалі відображалися як кілька окремих точок (по кількості SSID). Довелося класифікувати їхніми руками, щоб уникнути проблем з WIPS.

Позиціонування об'єктів

Однієї із самих цікавих фіч системи є позиціонування об'єктів Wi-Fi мережі. Теоретично налаштування даного механізму не потрібно й він працює з коробки. На практиці – щоб усе було гарно, необхідне дуже ретельне налаштування карти й калібрування сенсорів. Калібрування являє собою побудову системою графіка тріангуляції об'єктів і визначення відстані до них. Система демонструє фактичний і очікуваний графіки, і в ідеалі вони повинні збігатися. Але, враховуючи наявність стін і інших генеруючих перешкоди об'єктів, гарним результатом для точного позиціонування є 70% відповідності графіків. Усе зводиться до оптимального розташування точок у приміщенні й правильному налаштуванню карти (нюансів отут, звичайно, море). Є й інші особливості – по-перше, система одержує інформацію від точок доступу, керованих стороннім Wi-Fi контролером через певні проміжки часу, і якщо використовувати для позиціонування об'єкта, що рухається, дану інформацію, то при зіставленні поточних даних від сенсорів з даними від точки доступу п'ятихвилинної давнини позиція буде «злегка» неточною. По-друге, при створенні карти приміщення немає можливості вказати об'єкти, що погіршують проходження сигналу. Наприклад, електрощитова, розташована на поверсі, може суттєво спотворити картину.

Як це працює?

Наприклад, у мережі з'явився злісний порушник і нам необхідно його знайти.

Переходимо у вкладку Monitoring>Security, знаходимо клієнта, що цікавить нас, і вибираємо опцію Locate.

Після цих нескладних маніпуляцій одержимо синенький квадратик на карті (якщо всі правильно настроїли й відкалібрували).

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

Усі тести по виявленню тестових порушників, зайвих точок, підключених до провідної мережі, і запобігання підключення клієнтів до бездротових мереж у цілому були успішні. При протидії стандартним атакам виду connection flood, forced disconnection і т.п. деградація пропускну здатності мережі однаково помітна. Але знайти джерело все-таки стає простіше.

3.3 Розробка функціональної схеми

Поширення бездротових мереж і велика кількість нових мобільних обчислювальних пристроїв розмили традиційні границі між довіреними й не довіреними мережами, а також перемістили пріоритети безпеки з мережного периметра на захист інформації й на забезпечення безпеки користувачів. До проблем ІТ-безпеки відносяться: нелегальні точки доступу, що створюють т.зв. «чорний хід», DDoS-атаки (distributed denial-of-service), мережна розвідка « по повітрю», підслуховування, виламування трафіку й необхідність демонструвати відповідність галузевим вимогам.

Цінність застосунку WIPS

WIPS – це всеосяжний застосунок по забезпеченню бездротової безпеки, з метою виявлення, локалізації, блокування й обмеження провідних і бездротових погроз на мережних рівнях 1 і 2. Інтеграція бездротової системи запобігання вторгнень в інфраструктуру бездротової локальної мережі (WLAN) забезпечує економічну й операційну ефективність завдяки використанню єдиної інфраструктури й для сервісів WIPS, і для сервісів WLAN.

До складу описуваного застосунку входять наступні компоненти:

- стандартні точки доступу WIPS Aironet з модулями бездротової безпеки Wireless Security Module;
- контролери бездротової локальної мережі WIPS;
- платформа WIPS Mobility Services Engine;
- інфраструктура WIPS Infrastructure.

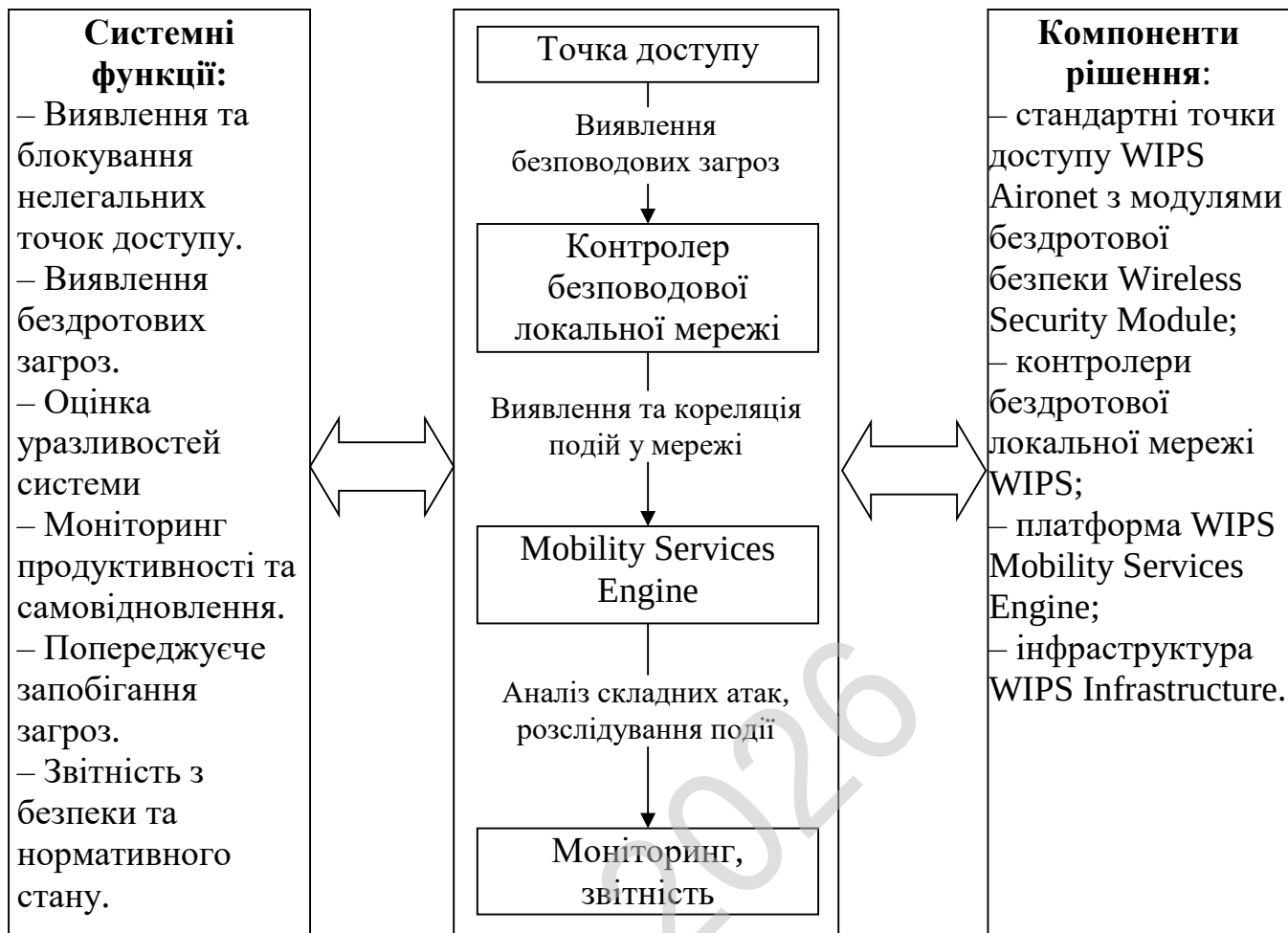


Рисунок 3.2 – Функціональна схема системи

Переваги застосунку WIPS

Застосунок WIPS пропонує розширений набір засобів, які з погляду архітектури неможливо реалізувати у випадку окремих застосунків, що підключаються, для запобігання вторгнень у бездротові мережі. Архітектура застосунку WIPS, інтегрована в інфраструктуру, дозволяє мережним адміністраторам зробити наступне:

– Забезпечити всеосяжний захист. Застосунок WIPS ідентифікує й локалізує бездротові атаки на мережа, що захищається їм (у т.ч. нелегальні точки доступу, мережна розвідка, злом автентифікації й шифрування, DoS-атаки, атаки типу man-in-the-middle, спроби запозичення прав, атаки нульового дня, нові

невідомі атаки), щоб забезпечити всеосяжний захист у всьому бездротовому середовищі.

– Бачити повну картину. Застосунок WIPS аналізує мережний трафік, щоб виявити нестандартну активність і бездротові атаки в зонах дії точок доступу й контролерів бездротової локальної мережі. Крім того, воно відслідковує інвентаризаційний перелік пристроїв, проводить аудитів мережної конфігурації й здійснює моніторинг показників функціонування мережного середовища.

– Здійснення коригувальних дій. Застосунок WIPS не тільки виявляє погрози, уразливості й проблеми продуктивності; воно повідомляє адміністраторів про триваючі бездротові погрози, локалізує атакуючого, реєструє релевантні дані для розслідування й автоматично ініціює протидіючі заходи, якщо це можливо.

– Задіяти всі ресурси бездротової локальної мережі. Застосунок WIPS здатне використовувати всі точки доступу мережі, що захищається, для виявлення, локалізації й блокування нелегальних пристроїв. Це підвищує точність локалізації, знижує кількість ложнопозитивних реакцій і прискорює блокування.

– Пропонувати безперервний актуальний захист. Завдяки автоматизованій оцінці уразливостей і актуальній бібліотеці погроз адміністратор бездротової мережі має необхідні знання для захисту бездротової мережі, навіть не будучи фахівцем з безпеки.

– Скористатися гнучкими архітектурами для розгортання. Застосунок WIPS може використовувати точки доступу для моніторингу на постійній основі, точки доступу, що обслуговують користувачів бездротової мережі з одночасним забезпеченням захисту в каналі або виділений модуль бездротової безпеки Wireless Security Module, що забезпечує захист у діапазонах 2,4 і 5 ГГц. При цьому показники роботи радіопристроїв, що обслуговують дані, не погіршуються.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

– Використовувати застосунок корпоративного рівня, що забезпечує можливості керування. Інфраструктура WIPS Infrastructure здатна управляти сотнями контролерів бездротової локальної мережі WIPS і до 15000 точок доступу WIPS Aironet. Застосунок WIPS використовує платформу WIPS Mobility Services Engine для локалізації бездротових погроз і кореляції подій безпеки, таких як нелегальні точки доступу, джерела перешкод і активні вторгнення.

Характеристики

Застосунок WIPS має наступні ключові характеристики й перевагами:

– Виявлення, локалізація, класифікація й блокування нелегальних точок доступу. Застосунок WIPS виявляє, автоматично класифікує на основі правил, що настроюються, і блокує нелегальні точки доступу, клієнтів нелегальних точок доступу, неправильних клієнтів і клієнтські ad hoc з'єднання.

– Виявлення бездротових атак. Застосунок WIPS ідентифікує й локалізує атаки на бездротову мережу, у т.ч. нелегальні точки доступу, DoS-атаки на валідних клієнтів і мережа, атаки типу man-in-the-middle, спроби використання чужого пароля, атаки нульового дня й нові невідомі атаки.

– Моніторинг уразливостей захисту. Застосунок WIPS самостійно виконує автоматичний цілодобовий моніторинг і оцінку бездротових уразливостей шляхом постійного сканування, що попереджає й, бездротової мережі на предмет ослаблення захисту й конфігурацій, невідповідних політикам.

– Керування, моніторинг і звітність. Застосунок WIPS повністю інтегроване в інфраструктуру WIPS Infrastructure, що забезпечує єдина уніфікована вистава для керування провідною й бездротовою мережею. Інфраструктура WIPS Infrastructure пропонує вбудовані звіти про відповідність галузевим вимогам (у тому числі, обов'язкові звіти на відповідність стандартам Payment Card Industry (PCI) 2.0).

3.4 Розробка діаграми процесів

Відповідно до методичних рекомендацій розроблення графічної частини кваліфікаційної бакалаврської роботи розглянемо розроблену діаграму процесів яка зображена на рисунку 3.3.



Рисунок 3.3 – Діаграма взаємодії процесів

Розроблена діаграма взаємодії процесів використовується для представлення та візуалізації процесів обробки даних тобто структурного проектування бакалаврської роботи.

Основні складові елементи діаграми взаємодії процесів це потоки даних:

- Репозиторії, потік сховища даних.
- Потоки зовнішні по відношенню до системи сутності.
- Процеси які являють собою трансформацію даних в рамках описуваної системи.
- Потоки даних гібридні між елементами трьох попередніх типів.

Відповідно до документації основна будова діаграми процесів полягає у графічному представленні складу сукупностей даних, що характеризуються як співвідношення різних частин кожної з сукупностей. Склад статистичної сукупності графічно може бути представлений як за допомогою абсолютних, так і відносних показників. Графічне зображення складу сукупності по абсолютними і відносними показниками сприяє проведенню більш глибокого аналізу і дозволяє проводити аналіз системи.

Для схематичного представлення системи що розробляється необхідно спочатку представити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи в цілому у подальшому. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі. Розроблена діаграма взаємодії процесів системи в подальшому уточнюється шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

Таким чином у результаті після розгляду, вищеописаної системи, схеми структурної, функціональної, діаграми взаємодії процесів перейдемо до опису та розгляду блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Розглянемо реалізацію бакалаврської дипломної роботи. Були проведені розрахунки і підібрані набори тестових даних для перевірки правильності реалізації проектних рішень.

Блок-схеми показують весь процес роботи системи з підсистемами та частково доказують правильність вибраних проектних рішень. 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 високого рівня декомпозиції задач на класи. 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 підсистеми 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

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

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.

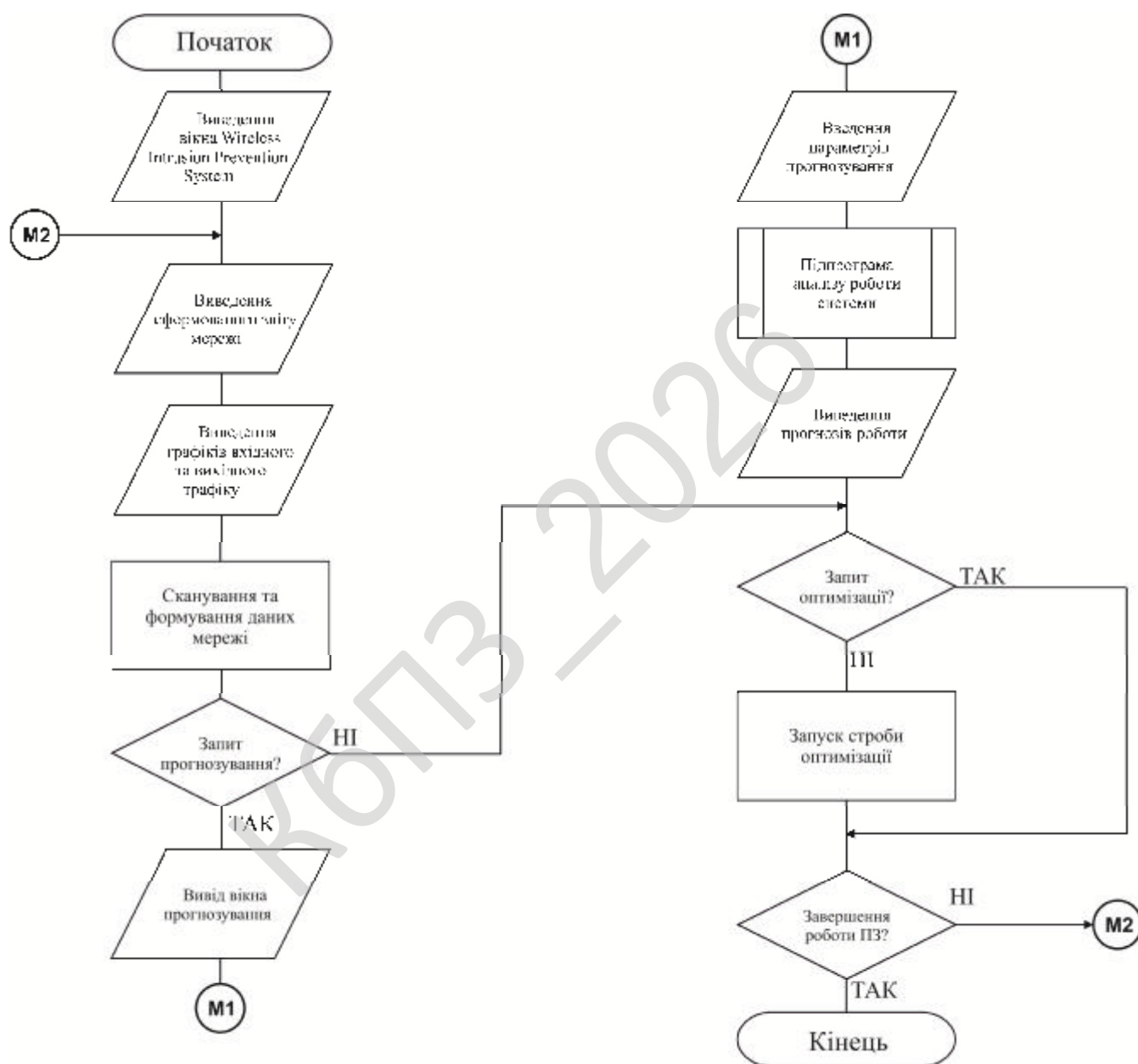


Рисунок 4.1 – Блок-схема основної програми

Було використано наступні підходи UML: діаграма діяльності (діаграми поведінки типу); діаграма прецедентів (діаграми поведінки типу); Діаграма класів.

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.

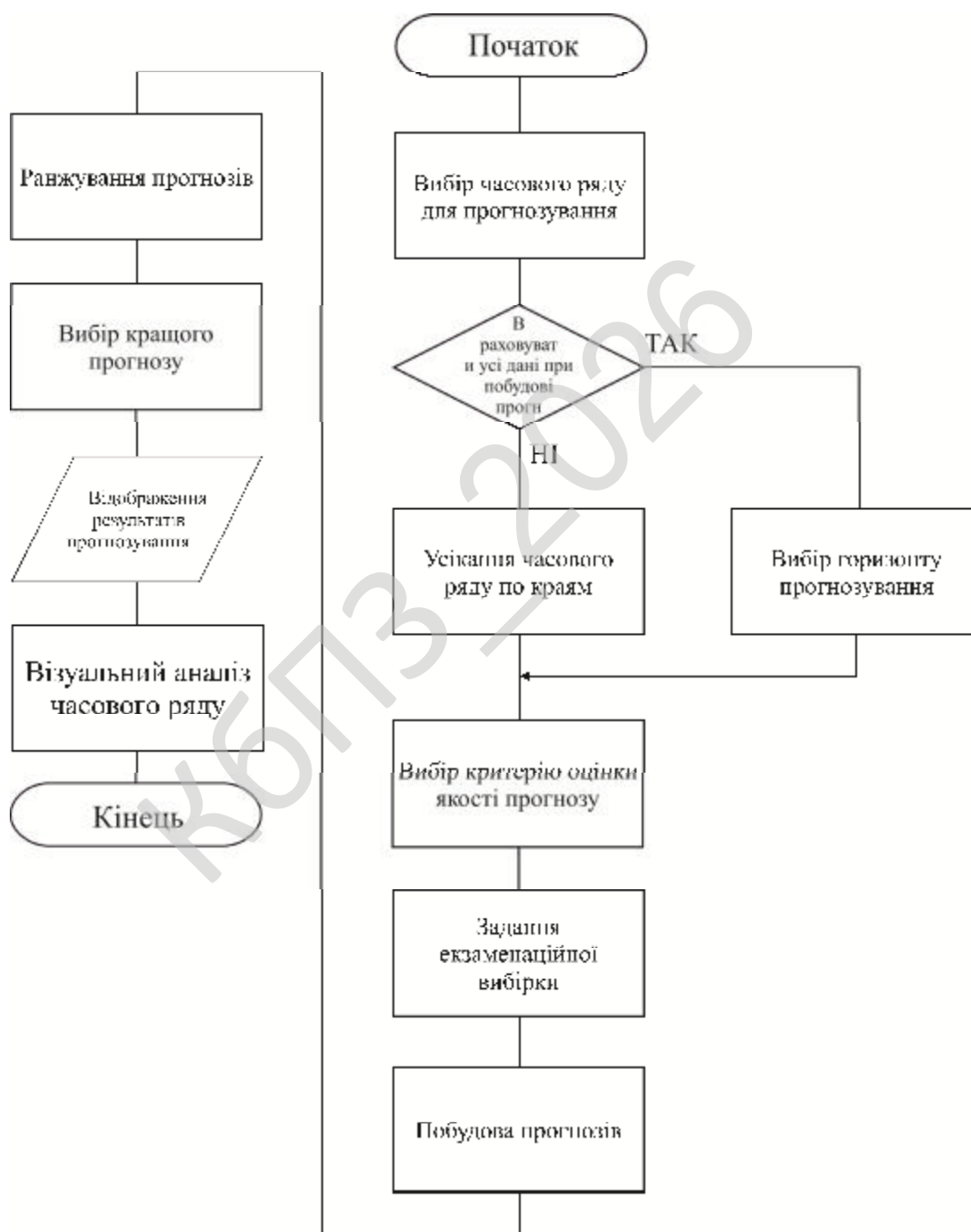


Рисунок 4.2 – Блок-схема роботи підпрограми

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);

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

- 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".

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

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 зафарбовани ромбиком.

Система керування базами даних (СКБД, Database Management System, DBMS) – набір взаємопов'язаних даних (база даних) і програм для доступу до цих даних. Надає можливості створення, збереження, оновлення та пошуку інформації в базах даних з контролем доступу до даних.

Першим поколінням СКБД прийнято вважати ієрархічні й мережеві системи. Ці системи отримали широке поширення в 1970-х роках, а першою комерційною системою цього типу була система IMS компанії IBM.

У 1980-х роках ці системи були витіснені системами другого покоління – повсюдно використовуваними і донині реляційними СКБД. У цих системах використовувалися непроцедурні мови управління даними (SQL) і передбачався значний ступінь незалежності даних. Реляційні системи внесли значні удосконалення в управління даними: графічний користувацький інтерфейс (GUI),

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

клієнт-серверні застосунки, розподілені бази даних, паралельний пошук даних та інтелектуальний аналіз даних.

Але вже до кінця 1980-х років існуюча тоді реляційна модель перестала задовольняти розробників в силу низки обмежень. Відповіддю на зростаючу складність програм баз даних стали два нових напрямки розвитку СКБД: об'єктно-орієнтовані СКБД і об'єктно-реляційні СКБД.

У 1991 був утворений консорціум ODMG, основною метою якого стало вироблення промислового стандарту об'єктно-орієнтованих баз даних. Між 1993 та 2001 роками ODMG опублікувала п'ять ревізій своїх специфікацій. Остання версія стандарту має індекс 3.0, після чого група розпустилася. До кінця 1990-х існувало близько десяти компаній, що виробляли комерційні продукти, що позиціонуються на ринку як ООСКБД. Найбільш відомими системами даного класу стали Objectivity, Versant виробництва однойменних компаній, а також СКБД Jasmine, випущена компанією CA. Незважаючи на переваги, що дозволяють ефективніше вирішувати певний ряд завдань, об'єктно-орієнтовані системи так і не змогли завоювати значущу частку ринку СКБД, залишившись «нішевим» продуктом.

Постачальниками традиційних реляційних СКБД також була проведена значна робота з об'єднання об'єктно-орієнтованих і реляційних систем. Розробники постаралися розширити мову SQL, щоб включити в неї концепції об'єктно-орієнтованого підходу, зберігаючи переваги реляційної моделі (об'єктні розширення мови SQL були зафіксовані в стандарті SQL:1999).

Основний принцип – це еволюційний розвиток можливостей СКБД без поломки попередніх підходів та зі збереженням наступності з системами попереднього покоління.

Поняття СКБД третього покоління, якими, власне кажучи, і є об'єктно-реляційні СКБД, з'явилося після опублікування групою відомих фахівців в області баз даних «Маніфесту систем баз даних третього покоління». Основні принципи СКБД третього покоління, позначені в маніфесті:

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		40

Крім традиційних послуг з управління даними, СКБД третього покоління повинні забезпечити підтримку розвиненіших структур об'єктів і правил. Розвинутіша структура об'єктів характеризує засоби, необхідні для зберігання і маніпулювання нетрадиційними елементами даних (тексти, просторові дані, мультимедіа).

СКБД третього покоління повинні включити в себе СКБД другого покоління. Системи другого покоління внесли вирішальний вклад у двох областях – непроцедурний доступ за допомогою мови запитів SQL і незалежність даних. Ці досягнення обов'язково повинні враховуватися в системах третього покоління.

СКБД третього покоління повинні бути відкриті для інших підсистем. Це включає оснащення різноманітними інструментами підтримки прийняття рішень, доступом з багатьох мов програмування, інтерфейсами до існуючих популярних систем і бізнес-застосунків, можливістю запуску програм з бази даних на іншій машині і розподілені СКБД. Весь набір інструментів і СКБД має ефективно функціонувати на різноманітних апаратних платформах з різними операційними системами.

Крім того, СКБД, що розраховує на широку сферу застосування, повинна бути оснащена мовою четвертого покоління (4GL).

У середині 1990 років було лише кілька дослідних прототипів СКБД, які поєднали найкращі риси реляційних і об'єктно-орієнтованих СКБД. Першим комерційним продуктом, якому були властиві об'єктно-реляційні риси, став Universal Server компанії Informix(згодом була поглинена IBM). В даний час більшість цих ідей вже втілено в реальних комерційних рішеннях, в тому числі і в продуктах основних постачальників СКБД (Oracle Database і IBM DB2).

Розвиток індустрії систем керування базами даних базується на значних фундаментальних наукових дослідженнях. Найчастіше, між самими дослідженнями та їхньою конкретною реалізацією в прикладних рішеннях минають роки, а іноді й десятиліття. Роботу в області управління даними

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

проводять як університетські дослідницькі групи (MIT, Berkeley), так і центри розробок основних постачальників СКБД (Oracle, IBM, Microsoft). Інвестування в управління даними – це довгострокове, і разом з тим, вигідне вкладення коштів. В даний час дослідники мають у своєму розпорядженні засоби, що дозволяють ефективно реалізувати найскладніші запити, що маніпулюють терабайтами й петабайтами різних даних.

Основними тенденціями, які дали привід для проведення різних масштабних досліджень в області баз даних стали:

Експонентний ріст даних. Обсяг даних, у тому числі синтетичних, що генеруються автоматизованими системами, значно зріс. Збільшилося і число прикладних областей, в яких вимагається обробка великих обсягів даних. До таких областей тепер відносяться не тільки традиційні корпоративні програми, пошук у веб, але також і наукові дослідження, обробка природних мов, аналіз соціальних мереж тощо.

Значне ускладнення структур використовуваних даних. Прості види даних у вигляді чисел і символьних рядків стали доповнятися численною мультимедійною інформацією, просторовими, процедурними даними та великою кількістю інших складних форматів.

Широке поширення дешевих високопродуктивних апаратних засобів. Щорічно ми спостерігаємо зростання обчислювальних можливостей мікропроцесорів, збільшення ємності і зниження вартості доступних і зручних в експлуатації пристроїв дискової оперативної пам'яті.

Активний розвиток засобів комунікації та «всесвітньої павутини» World Wide Web. WWW стає єдиним інформаційним середовищем, що пронизує весь світ і об'єднує величезне число користувачів та електронних пристроїв.

Поява нових важливих областей застосування СКБД. У першу чергу, це пов'язано з інтелектуальним аналізом даних, сховищами даних, а останнім часом – з паралельними обчисленнями і хмарними технологіями.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		42

Основні характеристики СКБД

1. Контроль за надлишковістю даних.
2. Несуперечливість даних.
3. Підтримка цілісності бази даних (коректність та несуперечливість).
4. Цілісність описується за допомогою обмежень.
5. Незалежність прикладних програм від даних.
6. Спільне використання даних.
7. Підвищений рівень безпеки.

Можливості СКБД

1. Дозволяється створювати БД (здійснюється за допомогою мови визначення даних DDL (Data Definition Language)).

2. Дозволяється додавання, оновлення, видалення та читання інформації з БД (за допомогою мови маніпулювання даними DML, яку часто називають мовою запитів)

3. Можна надавати контрольований доступ до БД за допомогою: Системи забезпечення захисту, яка запобігає несанкціонованому доступу до БД; Системи керування паралельною роботою прикладних програм, яка контролює процеси спільного доступу до БД; Система відновлення – дозволяє відновлювати БД до попереднього несуперечливого стану, що був порушений в результаті збою апаратного або програмного забезпечення.

Основні компоненти середовища СКБД

1. Апаратне забезпечення.
2. Програмне забезпечення.
3. Дані.
4. Процедури – інструкції та правила, які повинні враховуватись при проектуванні та використанні БД.
5. Користувачі: адміністратори даних (керування даними, проектування БД, розробка алгоритмів, процедур) та БД (фізичне проектування, відповідальність за безпеку та цілісність даних); розробники БД; прикладні програмісти; кінцеві користувачі.

Архітектура СКБД

Існує трирівнева система організації СКБД ANSI-SPARC, при якій існує незалежний рівень для ізоляції програми від особливостей представлення даних на нижчому рівні.

Рівні:

1. Зовнішній – представлення БД з точки зору користувача.
2. Концептуальний – узагальнене представлення БД, описує які дані зберігаються в БД і зв'язки між ними. Підтримує зовнішні представлення, підтримується внутрішнім рівнем.
3. Внутрішній – фізичне представлення БД в комп'ютері.

Логічна незалежність – повна захищеність зовнішніх моделей від змін, що вносяться в концептуальну модель.

Фізична незалежність – захищеність концептуальної моделі від змін, які вносяться у внутрішню модель.

MySQL – вільна система керування реляційними базами даних. Розробка та підтримка сайту MySQL здійснює корпорація Oracle, яка отримала права на торговельну марку разом з поглиненої Sun Microsystems, яка раніше придбала шведську компанію MySQL AB.

Продукт поширюється як під GNU General Public License, так і під власною комерційною ліцензією. Крім цього, розробники створюють функціональність за замовленням ліцензійних користувачів. Саме завдяки такому замовленню майже в найраніших версіях з'явився механізм реплікації.

MySQL є рішенням для малих і середніх додатків. Входить до складу серверів WAMP, AppServ, LAMP і в портативні збірки серверів Денвер, ХАМРР, VertrigoServ. Зазвичай MySQL використовується як сервер, до якого звертаються локальні або видалені клієнти, проте в дистрибутив входить бібліотека внутрішнього сервера, що дозволяє включати MySQL в автономні програми.

Гнучкість СКБД MySQL забезпечується підтримкою великої кількості типів таблиць: користувачі можуть вибрати як таблиці типу MyISAM, що

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		44

підтримують повнотекстовий пошук, так і таблиці InnoDB, що підтримують транзакції на рівні окремих записів. Більш того, СКБД MySQL поставляється із спеціальним типом таблиць EXAMPLE, що демонструє принципи створення нових типів таблиць. Завдяки відкритій архітектурі і GPL-ліцензуванню, в СКБД MySQL постійно з'являються нові типи таблиць.

26 лютого 2008 року Sun Microsystems придбала MySQL AB за 1 млрд доларів, 27 січня 2010 року Oracle придбала Sun Microsystems за 7,4 млрд доларів і включила MySQL в свою лінійку СКБД.

Спільнотою розробників MySQL створені різні відгалуження коду, такі як Drizzle (англ.), OurDelta, Percona Server і MariaDB. Всі ці відгалуження вже існували на момент поглинання компанії Sun корпорацією Oracle.

MySQL має подвійне ліцензування. MySQL може розповсюджуватися відповідно до умов ліцензії GPL. Але за умовами GPL, якщо якась програма використовує бібліотеки MySQL, то вона теж повинна розповсюджуватися за ліцензією GPL. Проте це може розходитися з планами розробників, які не бажають відкривати сирцеві тексти своїх програм. Для таких випадків передбачена комерційна ліцензія компанії Oracle, яка також забезпечує якісну сервісну підтримку.

В разі використання та розповсюдження програмного забезпечення з іншими вільними ліцензіями, такими як BSD, Apache, MIT та інші, MySQL дозволяє використання бібліотек MySQL за ліцензією GPL.

MySQL виникла як спроба застосувати mSQL до власних розробок компанії: таблиць, для яких використовувалися ISAM – підпрограми низького рівня для індексного доступу до даних. У результаті був вироблений новий SQL-інтерфейс, але API-інтерфейс залишився в спадок від mSQL. Звідки походить назва «MySQL» – достеменно не відомо. Розробники дають два варіанти: або тому, що практично всі напрацювання компанії починалися з префікса Му, або на честь дівчинки на ім'я Му, дочки Майкла Монті Віденіуса, одного з розробників системи.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		45

Логотип MySQL у вигляді дельфіна носить ім'я «Sakila». Він був обраний з великого списку запропонованих користувачами «імен дельфіна». Ім'я «Sakila» було відправлено Open Source-розробником Ambrose Twebaze.

В січні-лютому 2008 Sun Microsystems придбала розробника системи керування базами даних MySQL за \$1 млрд. Після поглинання у 2009 році Sun Microsystems компанією Oracle Corporation MySQL стала власністю Oracle.

За час розвитку під орудою Oracle дедалі більше відокремлює MySQL від спільноти і робить процес розробки все менш прозорим. Наприклад, повернута практика поставки власницьких розширених функцій в Enterprise-версії MySQL, спостерігається приховування інформації про вразливості, зі складу виключений тестовий набір, закритий доступ до більшої частини системи відстеження помилок та припинено публікація згрупованого логу змін, що дозволяє судити про прив'язку патчів до конкретних змін.

Можливості

MySQL – компактний багатопотоковий сервер баз даних. Характеризується високою швидкістю, стійкістю і простотою використання.

MySQL вважається гарним рішенням для малих і середніх застосувань. Сирцеві коди сервера компілюються на багатьох платформах. Найповніше можливості сервера виявляються в UNIX-системах, де є підтримка багатопоточності, що підвищує продуктивність системи в цілому.

Можливості сервера MySQL:

1. Простота у встановленні та використанні.
2. Підтримується необмежена кількість користувачів, що одночасно працюють із БД.
3. Кількість рядків у таблицях може досягати 50 млн.
4. Висока швидкість виконання команд.
5. Наявність простої і ефективної системи безпеки.

Подійно-орієнтована архітектура (Event-driven architecture, надалі EDA) – шаблон архітектури програмного забезпечення, який призначений для створення подій, їх виявлення, споживання і реагування на них.

Подія може бути визначена як значна зміна стану. Наприклад, коли споживач купує автомобіль, стан автомобіля змінюється з "на продаж" до "продано". Архітектура системи дилера автомобілів може трактувати цю зміну стану як подію, поява якої може стати відомою іншим програмам даної архітектури.

З формальної точки зору, те, що виробляється, публікується, поширюється, виявляється і споживається (як правило, асинхронно) є повідомленням, яке називають сповіщенням про подію (або нотифікацією), а не самою подією, яка є зміною стану, що викликає появу повідомлення.

Події не подорожують, вони просто відбуваються. Проте термін подія часто використовується метонімічно для позначення самого нотифікаційного повідомлення, що може призвести до певної плутанини.

Цей архітектурний шаблон може застосовуватися при проектуванні і реалізації ПЗ і систем, які передають події між слабкозв'язаними компонентами програмного забезпечення і сервісами (службами).

Подійно-орієнтована система як правило складається з емітерів подій (або агентів) і споживачів подій (або стоків).

Стоки несуть відповідальність за здійснення реагування на появу події. Реакція не завжди може бути повністю забезпечена самим стоком. Наприклад, стік, може бути відповідальним лише за фільтрацію, трансформацію і відправку події до іншого компонента або він може забезпечити повністю самостійну реакцію на таку подію. Перша категорія стоків може бути заснована на традиційних компонентах, таких як проміжне програмне забезпечення, орієнтоване на обробку повідомлень (message oriented middleware, MOM), в той час, як друга категорія стоків (самостійна реакція в режимі он-лайн) може вимагати більш придатної платформи (фреймворку) для виконання транзакцій.

Розробка ПЗ і систем в подійно-орієнтованій архітектурі дозволяє їм бути сконструйованими способом, який більш відповідає вимогам до їх створення, оскільки такі системи в більшій мірі пристосовуються до непередбачуваних і

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

асинхронних середовищ. Подійно-орієнтована архітектура (EDA) може доповнювати сервісно-орієнтовану архітектуру (SOA), оскільки сервіси (служби) можуть бути активовані тригерами, які ініціюються при настанні подій.

Ця парадигма особливо корисна, коли стік не забезпечує власного виконання будь-яких дій.

Подійно-орієнтована SOA (SOA-2) розвиває архітектури SOA і EDA для забезпечення більш глибокого і надійного рівня сервісів за рахунок використання раніше невідомих причинно-наслідкових зв'язків, щоб сформувати новий шаблон подій. Цей новий шаблон бізнес-аналітики дає поштовх до небаченого раніше зростання рівня автоматизації підприємства за рахунок привнесення додаткової цінної інформації в описану раніше модель діяльності.

Обчислювальна техніка та сенсорні пристрої (сенсори, датчики, контролери) можуть виявляти зміни стану об'єктів або умов і створювати події, які потім можуть бути оброблені сервісом (службою) або системою.

Подія може складатися з двох частин: заголовка події та тіла події. Заголовок події може включати в себе інформацію таку як, наприклад, назва події, часова мітка події і тип події. Тіло події – це частина, яка описує факт, що стався в дійсності. Тіло події не слід плутати з шаблоном або логікою, яка може бути застосована як реакція на саму подію.

Архітектура, керована подіями, складається з чотирьох логічних рівнів (шарів). Вона починається з виявлення факту, його технічного подання у формі події і закінчується непустою множиною реакцій на цю подію.

Генератор подій.

Першим логічним шаром є генератор подій, який виявляє факт і представляє цей факт подією. Оскільки фактом може бути практично все, що може бути сприйнято, то ним може бути і генератор подій. Наприклад, генератором може бути клієнт електронної пошти, система електронної комерції або певний тип датчика.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

Перетворення різних даних, отриманих від датчиків, в єдину стандартизовану форму даних, які можуть бути оцінені, є основною проблемою при розробці та реалізації цього шару. Однак, враховуючи, що подія є строго декларативною, можна легко застосовувати будь-які операції трансформації, тим самим усуваючи необхідність забезпечення високого рівня стандартизації.

Канал подій.

Канал подій – це механізм, через який інформація від генератора подій передається до обробника подій (event engine) або стоку.

Це може бути з'єднання TCP/IP або вхідний файл будь-якого типу (простий текст, формат XML, e-mail тощо). В один і той же час може бути відкрито кілька каналів подій. Як правило, оскільки обробник подій повинен працювати в режимі, наближеному до реального часу, канали подій зчитуються асинхронно. Події зберігаються в черзі, очікуючи наступної обробки механізмом обробки подій.

Механізм обробки подій.

Механізм обробки подій (event processing engine) є місцем, де подія ідентифікується і вибирається відповідна реакція на нього, яка потім виконується. Це також може призвести до породження ряду тверджень. Якщо подія, яка надійшла до механізму обробки подій, є наприклад такою «Запаси продукту ID досягли нижнього допустимого рівня», це може ініціювати, наприклад, такі реакції як «Замовити продукт ID» і «Сповістити персонал».

Наступна подійно-орієнтована дія (післядія).

Щодо того, як можуть проявлятися наслідки події, слід відмітити, що вони можуть проявитись багатьма різними способами і у різноманітних формах (наприклад, повідомлення електронної пошти, надіслане комусь, або ПЗ, що виводить деяке попередження на екран). Залежно від рівня автоматизації, який забезпечується стоком (механізмом обробки подій), ці дії можуть виявитись зайвими.

Є три основні стилі обробки подій: простий, потоковий і складний. Часто ці три стилі використовуються спільно у розвинутій подійно-орієнтованій архітектурі.

Проста обробка подій.

Проста обробка подій стосується подій, які безпосередньо належать до специфічних вимірних змін умов. У випадку простої обробки подій, мають справу з появою відомих подій, що ініціюють післядію (післядії). Проста обробка подій зазвичай використовується для управління потоком робіт в реальному часі, скорочуючи тим самим час затримки і вартість робіт.

Наприклад, прості події можуть створюватись (породжуватись) датчиком, що виявляє зміну тиску в шині або температуру навколишнього середовища.

Обробка потоку подій.

При обробці потоку подій (event stream processing, далі ESP) відбуваються як звичайні, так і відомі події. Звичайні події (заявки, передачі RFID) перевіряються на те, чи є вони відомими, і передаються інформаційним передплатникам. Обробка потоку подій зазвичай використовується для управління потоком інформації в реальному часі і на рівні підприємства, що дозволяє своєчасно приймати рішення.

Обробка складних подій.

Обробка складних подій (Complex event processing (CEP)) дозволяє за шаблонами простих і звичайних подій проводити аналіз того, чи напустила складна подія. Обробка складних подій полягає в оцінюванні взаємного впливу подій і в наступному виконанні дій. При цьому, типи подій (відомих або звичайних) можуть перетинатись, а події можуть виникати протягом тривалого періоду часу.

Кореляція подій може бути причинною, тимчасовою або просторовою. CEP вимагає використання складних інтерпретаторів подій, визначення і підбору шаблонів подій, а також відповідних кореляційних методів. Обробка складних подій зазвичай використовується для виявлення і реагування на аномальну

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		50

поведінку, загрози і можливості у бізнесі.

При розробці ПЗ було використано підходи ризик-менеджменту – це система управління ризиками, яка включає в себе стратегію та тактику управління, направлені на досягнення основних цілей. Ефективний ризик-менеджмент включає:

- систему управління;
- систему ідентифікації і вимірювання;
- систему супроводження (моніторингу та контролю).

Сучасна наука представляє ризик як вірогідну подію, в результаті настання якої можуть відбутися позитивні, нейтральні або негативні наслідки. Якщо ризик припускає наявність як позитивних, так і негативних результатів, він відноситься до спекулятивних ризиків. Якщо ж наслідки негативні, або відсутні взагалі, такий ризик іменується чистим.

Мета ризик-менеджменту – підвищення конкурентоспроможності господарюючих суб'єктів за допомогою захисту від реалізації чистих ризиків.

Теорія ризик-менеджменту ґрунтується на трьох базових поняттях: корисності, регресії і диверсифікації.

У 1738 швейцарський математик Даніель Бернуллі доповнив теорію вірогідності методом корисності або привабливості того або іншого результату подій. Ідея Бернуллі полягала в тому, що в процесі ухвалення рішення люди приділяють більше уваги розміру наслідків різних результатів, ніж їх вірогідність.

В кінці XIX століття англійський дослідник Ф. Гальтон запропонував вважати регресію або повернення до середнього значення універсальною статистичною закономірністю. Суть регресії трактувалася ним як повернення явищ до норми з часом. Згодом було доведено, що правило регресії діє в найрізноманітніших ситуаціях, починаючи з азартних ігор та розрахунку вірогідності виникнення нещасних випадків, і закінчуючи прогнозуванням коливань економічних циклів.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

У 1952 аспірант Університету Чикаго Гарі Марковіц в статті «Диверсифікація вкладень» («Portfolio Selection») математично обґрунтував стратегію диверсифікації інвестиційного портфеля, зокрема, він показав, як шляхом продуманого розподілу вкладень мінімізувати відхилення прибутковості від очікуваного показника. У 1990 Г. Марковіцу присуджена Нобелівська премія за розробку теорії і практики оптимізації портфеля фондових активів.

Етапи ризик-менеджменту

У ризик-менеджменті прийнято виділяти декілька ключових етапів:

- на першому етапі відбувається виявлення ризику з супутньою оцінкою вірогідності його реалізації і масштабу наслідків;
- на другому етапі здійснюється розробка ризик-стратегії з метою зниження вірогідності реалізації ризику і мінімізації можливих негативних наслідків;
- на третьому етапі вибираються методи і інструменти управління виявленим ризиком;
- на четвертому етапі проводиться безпосереднє управління ризиком;
- на завершальному етапі оцінюються досягнуті результати і коректується ризик-стратегія.

За ключовий етап ризик-менеджменту вважається етап вибору методів і інструментів управління ризиком.

Методи і інструментарій ризик-менеджменту

Базовими методами ризик-менеджменту є відмова від ризиків, зниження, передача і ухвалення.

Ризик-інструментарій значно ширший. Він включає політичні, організаційні, правові, економічні, соціальні інструменти, причому ризик-менеджмент як система допускає можливість одночасного застосування декількох методів і інструментів ризик-управління.

Найбільш часто вживаним інструментом ризик-менеджменту є страхування. Страхування припускає передачу відповідальності за відшкодування передбачуваного збитку сторонній організації (страхової компанії).

Прикладами інших інструментів можуть бути відмова від надмірно ризикової діяльності (метод відмови), профілактика або диверсифікація (метод зниження), аутсорсинг витратних ризикових функцій (метод передачі), формування резервів або запасів (метод ухвалення).

Незважаючи на те що я працював над ПЗ один в реалізації програми я використовував підходи пришвидшення розробки на основі методологій Agile – Extreme Programming.

Екстремальне програмування (Extreme Programming, далі XP) це методологія розробки програмного забезпечення, найпопулярніша серед так званих гнучких методологій. Має на меті поліпшення якості програмного забезпечення та чутливість до змін у вимогах замовників. Як вид гнучких методологій, XP радить часті "випуски" програми у коротких циклах розробки, що має на меті поліпшити продуктивність праці та покращити можливості виконання вимог замовника що змінюються. Авторами даної методології є Кент Бек, Ворд Каннінгем, Мартін Фаулер та інші.

Інші елементи екстремального програмування включають в собі: парне програмування, проведення обширної перевірки сирцевого коду, модульне тестування всього коду, уникання створення функціональності до того як вона дійсно необхідна, простота та ясність коду, очікування на зміну вимог замовників з плином часу та коли вимоги до продукту стають ясніші, досить часте спілкування із замовником та між самими програмістами.

Назва методології походить від ідеї застосувати корисні методи і практики розробки програмного забезпечення, піднявши їх до "екстремальних" рівнів.

Критики XP зауважують на потенційні недоліки цієї методології – нестабільні вимоги, незадокументовані компроміси конфліктів користувачів, відсутність загального документу дизайну програми.

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		53

Технологія екстремального програмування була розроблена Кентом Беком, Уардом Каннінгхемом та Роном Джеффріесом під час роботи над Chrysler Comprehensive Compensation System (C3). У 1996 Кент Бек став лідером проекту і почав вдосконалювати методи розробки, що застосовувалися в роботі над проектом. Свій метод він виклав у книзі «Extreme Programming Explained», котру було видано у жовтні 1999. Після купівлі Крайслера компанією Даймлер–Бенц проект C3 було скасовано у лютому 2000.

Хоча саме екстремальне програмування є відносно новим, багато її практик вже існували і використовувались протягом певного часу; однак, методологія підносить "найкращі практики" до екстремального рівня. Для прикладу, практика по плануванню і написанню тестів перед написанням кожної маленької частини коду було використано раніше в проекті НАСА "Меркурій". Для зменшення часу на розробку ПЗ деякі формальні документи тестування (такі як приймальне тестування) писались паралельно (або й раніше) з написанням самого ПЗ. Незалежна група тестування НАСА може писати процедури тестування базуючись на формальних вимогах до продукту до того як програмне забезпечення розроблене та інтегроване в систему. В ХР ця концепція піднесена до "екстремального рівня" завдяки написанню автоматичних тестів які перевіряють поведінку навіть малих частинок коду, а не тільки значних функціональних частин ПЗ.

Посібник Extreme Programming Explained: Embrace Change описує Екстремальне Програмування, як:

- Спроба примирити гуманність і продуктивність.
- Механізм для соціальної зміни.
- Шлях до удосконалення.
- Стиль розвитку.

Дисципліна розробки програмного забезпечення.

Головною метою Екстремального Програмування є скорочення вартості неочікуваних змін. У традиційних методах розробки (на кшталт SSADM) вимоги

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		54

до розвитку системи визначаються на початку роботи над проектом, і часто виправляються пізніше. Це означає, що вартість проекту через зміни буде більшою за заплановану (традиційна особливість для програмного забезпечення, що проектується).

ХР використовується для скорочення вартості змін, завдяки представленню простих значень, принципів і методів. При використанні екстремального програмування, проект повинен стати гнучкішим щодо змін.

Extreme Programming Explained описує екстремальне програмування як дисципліну розробки програмного забезпечення яка змушує людей створювати високоякісне ПЗ якомога швидше.

ХР намагається зменшити ціну зміни вимог до ПЗ завдяки малим циклам розробки, а не одним довгим циклом. Екстремальне програмування сприймає зміни до вимог як звичайні, неминучі та бажані аспекти розробки ПЗ, і ці зміни мають бути очікуваним. Основна ідея полягає в тому що неможливо розробити самодостатній пакет вимог до ПЗ, зміни в вимогах – неминучі.

Екстремальне програмування також вводить набір практик та принципів на основі методології гнучкої розробки програмного забезпечення.

Екстремальне програмування описує чотири базові активності що виконуються при розробці програмного забезпечення: написання коду, тестування, слухання та дизайн. Написання коду. Прихильники ХР заявляють що єдиним дійсно важливим результатом розробки ПЗ є код: без готового коду нема продукту.

Тестування. Методологія екстремального програмування заявляє, що якщо дрібне тестування може перевірити незначну частину функціональності, то багато дрібних тестів можуть перевірити набагато більше частинок і продукт в цілому.

Основні прийоми ХР. Дванадцять основних прийомів екстремального програмування (за першим виданням книги Extreme programming explained) можуть бути об'єднані в чотири групи:

1. Короткий цикл зворотного зв'язку (Fine scale feedback).

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		55

- 1.1. Розробка через тестування (Test driven development).
- 1.2 Гра в планування (Planning game).
- 1.3. Замовник завжди поруч (Whole team, Onsite customer).
- 1.4 Парне програмування (Pair programming).
- 2. Безперервний, а не пакетний процес.
- 2.1 Безперервна інтеграція (Continuous Integration).
- 2.2 Рефакторинг (Design Improvement, Refactor).
- 2.3 Часті невеликі релізи (Small Releases).
- 3. Розуміння, що поділяється всіма учасниками.
- 3.1 Простота (Simple design).
- 3.2 Метафора системи (System metaphor).
- 3.3 Колективне володіння кодом (Collective code ownership) або обраними шаблонами проектування (Collective patterns ownership).
- 3.4 Стандарт кодування (Coding standard or Coding conventions).
- 4. Соціальна захищеність програміста (Programmer welfare), а саме 40 годинний робочий тиждень (Sustainable pace, Forty hour week).

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм SEED – у криптографії симетричний блоковий криптоалгоритм на основі Мережі Фейстеля, розроблений Корейським агентством інформаційної безпеки (Korean Information Security Agency, KISA) в 1998 році. В алгоритмі використовується 128-бітний блок і ключ довжиною 128 біт. Алгоритм одержав широке поширення й використовується фінансовими й банківськими структурами, виробничими підприємствами й бюджетними установами Південної Кореї, оскільки 40-бітний SSL не забезпечує на даний момент мінімально необхідного рівня безпеки. Агентством по захисту інформації специфіковане використання шифру SEED у протоколах TLS і S/MIME. У той же

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

час, алгоритм SEED не реалізований у більшості сучасних браузерів і інтернет-додатків, що утрудняє його використання в даній сфері поза межами Південної Кореї.

SEED являє собою мережу Фейстеля з 16 раундами, 128-бітовими блоками й 128-бітовим ключем. Алгоритм використовує дві 8×8 таблиці підстановки, які, як такі з Safer, виведені з дискретного зведення в ступінь (у цьому випадку, x^{247} і x^{251} – плюс деякі «несумісні операції»). Це є деякою подібністю с MISTY1 у рекурсивності його структури: 128-бітовий повний шифр – мережа Фейстеля з F-функцією, що впливає на 64-бітові половини, у той час як сама F-функція – Мережа Фейстеля, складена з G-функції, що впливає на 32-розрядні половини. Однак рекурсія не простягнеться далі, тому що G-функція – не Мережа Фейстеля. В G-функції 32-розрядне слово розглядають як чотири 8-бітових байта, кожен з яких проходить через одну або іншу таблицю підстановки, потім поєднується в помірковано комплексному наборі булевих функцій таким чином, що кожен біт виводу залежить від 3 з 4 вхідних байтів.

SEED має складний ключовий розклад, генеруючи тридцять два 32-розрядних додаткових символу, використовуючи G-функції на серіях обертань вихідного неопрацьованого ключа, комбінованого зі спеціальними раундовими константами (як в TEA) від «Золотого співвідношення» (англ. Golden ratio).

Згідно з дослідженнями KISA, алгоритм SEED «надійно протистоїть відомим атакам».

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Програма має простий та інтуїтивно зрозумілий інтерфейс, який зображений на рисунку 5.1. З рисунку головного вікна можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

– Блок головного меню. Головне меню забезпечує швидкий перехід між розділами системи. Користувач відкриває моніторинг, список бездротових мереж, журнал загроз, події, налаштування, звіти та інформацію про програму.

– Блок стану системи. Блок показує поточний режим роботи Wireless Intrusion Prevention System. Статус «Активна» означає, що система виконує моніторинг мереж, аналіз трафіку та запобігання несанкціонованим підключенням.



Рисунок 5.1 – Головне вікно ПЗ

– Блок бездротових мереж. Блок відображає кількість активних Wi-Fi мереж, які контролюються системою. Він використовується для оцінювання поточного радіосередовища та виявлення сторонніх точок доступу.

– Блок виявлених загроз. Блок показує кількість загроз, які потребують уваги адміністратора. До таких загроз належать підозрілі точки доступу, атаки деаутентифікації, підміна мережі та надмірні запити клієнтів.

– Блок заблокованих атак. Блок демонструє кількість атак, які система вже зупиняє. Він підтверджує роботу механізму активного захисту та фільтрації небезпечної активності.

– Блок останніх подій. Таблиця містить журнал останніх інцидентів. У ній показано час події, тип загрози, короткий опис і рівень небезпеки.

– Блок стану захисту. Блок показує активність основних захисних служб. До них належать моніторинг мереж, виявлення загроз, запобігання вторгненням, фільтрація пакетів і логування подій.

– Блок службової інформації. Нижня частина вікна містить версію програми, час оновлення бази сигнатур і службові дані. Це дозволяє контролювати актуальність системи захисту.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення. Розроблена програма має дуже простий і зрозумілий інтерфейс з користувачем.

Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий.

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

Процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом.

Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

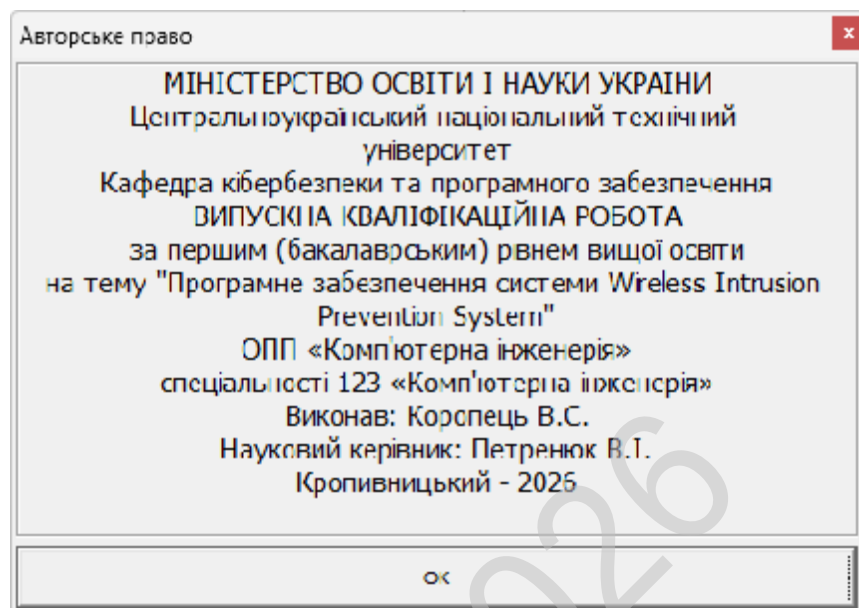


Рисунок 5.2 – Авторське право

Таким чином у результаті вищерозглянутого можна стверджувати що розроблено інтерфейс системи у відповідності з вибраною метою роботи. Система містить максимальний необхідний набір функцій придатних для виконання будь-яких дій для забезпечення повноцінної роботи програми. Далі розглянемо висновки та використані літературні джерела.

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи Wireless Intrusion Prevention System.

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

Рішення завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем Wireless Intrusion Prevention System.
- Досліджена система Wireless Intrusion Prevention System.
- На основі отриманих результатів досліджень створена програмна реалізація системи Wireless Intrusion Prevention System.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання Wireless Intrusion Prevention System.

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 створенні 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 призначені для системи Wireless Intrusion Prevention System. 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

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		61

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 SEED.

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.

К6ПЗ-2026

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Loren Kohnfelder. Designing Secure Software. No Starch Press. 2022. 332 p.
2. Samir Kumar Rakshit. Ethical Hacker's Penetration Testing Guide. BPB Online. 2022. 509 p.
3. Corey J. Ball. Hacking APIs. No Starch Press. 2022. 353 p.
4. Kevin Beaver. Hacking for Dummies. John Wiley & Sons. 2022. 419 p.
5. Mark S. Merkow. Practical Security for Agile and DevOps. CRC Press. 2022. 236 p.
6. Derek Fisher. Application Security Program Handbook. Manning Publications. 2021. 155 p.
7. Cameron Wyatt PH.D. Kali Linux Tutorial. Independently published. 2021. 60 p.
8. Kuznetsov O., Frontoni E., Kuznetsova Y., Chevardin V., Smirnov O. «Architectural foundations for adaptive security in edge computing systems». *Cybersecurity Defensive Walls in Edge Computing*, 2025. pp. 21-61.
9. 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.
10. 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
11. Kuznetsov, O., Frontoni, E., Kuznetsova, K., Arnesano, M., Smirnov, O. «A secure biometric authentication architecture for blockchain-driven cyber-physical

					ВКРБ-123.26.0030.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

systems». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 193–224.

12. 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.

13. 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.

14. 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.

15. 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.

16. 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.

17. 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.

18. 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.

19. Смірнова Т.В., Коноплицька-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.

20. 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.

21. 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.

22. 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.

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

24. 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

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

26. 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.

27. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskyi, V., Khorolska, K., Bebesko, B. «Selection of a Rational Composition of Information 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.

28. 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.

29. 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 розвитку інформаційних технологій (ІПШРІТ-2023)»* м.Черкаси 6 грудня 2023 року – Черкаси: ЧДТУ.– 2023. – С.251-252.

30. 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.

31. 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.

32. 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.

33. 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,

34. 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.

35. 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.

36. 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.

37. 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.

38. 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.

39. 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

40. 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.

41. 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.

42. 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.

43. 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.

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

45. 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.

46. 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.
47. 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.
48. 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.
49. 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.
50. 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.
51. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Ukraine, Kyiv, May 14-18. 2020. P. 172-177.
52. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) *Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies*, vol 48. Springer, Cham. 2021. pp 557-587.

53. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 125-136.

54. Smirnov, O., Shekhanin, K., Kuznetsov, A., Krasnobayev, V. «Detecting Hidden Information in FAT». *International Journal of Computer Network and Information Security (IJCNIS)*. Vol. 12, No. 3, 2020. PP.33-43.

55. Smirnov, O., Kuznetsov, A., Gorbacheva, L., Babenko, V., «Hiding data in images using a pseudo-random sequence», *CEUR Workshop Proceedings* Volume 2608, 2020, Pages 646-660.

56. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». *International Journal of Computing*; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

57. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

58. Smirnov, O., Ulichev, O., Meleshko, Y., Khokh, V., Goncharenko, I. «Method of Choosing Objects for Informational Influence in Social Networks during Information Campaign Based on the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, Vol 2588, P. 215-227, 2019.

59. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». *CEUR Workshop Proceedings*, Vol 2588, P. 90-106, 2019.