Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

**КВАЛІФІКАЦІЙНА РОБОТА**

Нечипорука Михайла Вікторовича

на здобуття ступеня вищої освіти Бакалавра

# Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека

Освітня програма Кібербезпека

КРБКБ.2102158.21.02.36 ПЗ

Виконав студент 4 курсу, група КБ-21-2 _____ 25.05.25 Михайло НЕЧИПОРУК
Підпис, дата Ініціали, прізвище

Керівник канд. тех. наук, доцент _____ 10.06.25 Віктор ЧЕШУН
Науковий ступінь, вчене звання Підпис, дата Ініціали, прізвище

Нормоконтролер старший викладач _____ 16.06.25 Сергій МОСТОВИЙ
Науковий ступінь, вчене звання Підпис, дата Ініціали, прізвище

До захисту допускаю:

Зав. кафедри кібербезпеки _____ Юрій КЛЬОЦ
Підпис, дата Ініціали, прізвище

16 06 2025р.

Хмельницький, 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Кібербезпеки

Рівень вищої освіти Бакалавр

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека

Освітня програма Кібербезпека

ЗАТВЕРДЖУЮ

Завідувач кафедри кібербезпеки

Юрій КЛЬОЦ

15 лютого 2025 р.

**ЗАВДАННЯ**

**НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Нечипоруку Михайлу Вікторовичу

1 Тема роботи Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення

Керівник роботи канд. техн. наук, доцент, Віктор Миколайович Чешун

Затверджено наказом ректора університету від 7 лютого 2025 № 23

2 Строк подання студентом кваліфікаційної роботи на кафедру

3 Вихідні дані до роботи Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення , а саме система виконана в програмному вигляді. В програмі зберігаються всі дані клієнтів у безпечному вигляді, виявляє небезпечні запити, журнал активності. Програма створена у вигляді бази даних. Створено sql-таблиці кожна має свої дані і функції, це вже детально розписано в самій роботі.

4 Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Основні поняття інформаційної безпеки та значення інформаційних ресурсів інтернет-провайдера, класифікація загроз та аналіз видів шкідливого програмного забезпечення, методи та засоби захисту від шкідливого ПЗ, формулювання задачі та вимог до системи захисту, проєктування архітектури захисної системи, створення структури бази даних користувачів, вибір та реалізація методу шифрування персональних даних, реалізація засобів захисту від

шкідливого ПЗ (фільтрація запитів тощо), тестування та оцінка ефективності запропонованої системи.

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

Схема системи бази даних інтернет-провайдера, Схема пропонованої система захисту інформаційних ресурсів, Схема у вигляді структури ключових параметрів конфігурації захисту системи.

6 Консультанти розділів кваліфікаційної роботи

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

7 Дата видачі завдання 16 лютого 2025 р.

## КАЛЕНДАРНИЙ ПЛАН

| Назва етапів (розділів) кваліфікаційної роботи | Строк виконання етапів роботи | Примітка |
|---|---|---|
| Вибір і затвердження теми кваліфікаційної роботи | Січень-Лютий | |
| Ознайомлення з предметною областю | Лютий | |
| Дослідження існуючих рішень | Лютий | |
| Постановка задачі | Березень | |
| Визначення загальних принципів рішення задачі | Березень | |
| Деталізація принципів рішення задачі | Квітень | |
| Розробка проєктних рішень | Квітень | |
| Апробація проєктних рішень | Травень | |
| Оформлення пояснювальної записки згідно вимог | Травень | |
| Оформлення графічної частини | Червень | |
| Захист КР | Червень | |

Студент _____ Михайло НЕЧИПОРУК

Керівник кваліфікаційної роботи _____ Віктор ЧЕШУН

# АНОТАЦІЯ

Тема кваліфікаційної роботи: Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення.

Автор роботи: Нечипорук Михайло Вікторович

Керівник роботи: канд. техн. наук, доц. Чешун Віктор Миколайович

Загальний обсяг роботи: 70 сторінок, 2 додатки, 14 рисунків, 15 таблиць, 40 джерел.

Графічна частина: 3 плакати, 10 презентаційних слайдів.

Ключові слова: інформаційна безпека, інтернет-провайдер, база даних, хешування, шифрування.

Кваліфікаційна робота бакалавра присвячена дослідженню теоретичних засад, проєктуванню та реалізації елементів захисту інформаційних ресурсів у середовищі інтернет-провайдера. Акцент роботи зроблено на проблемі поширення шкідливого програмного забезпечення (ШПЗ) в інфраструктурі ресурсів інтернет-провайдерів та способам його виявлення і нейтралізації.

У роботі досліджено основні поняття інформаційної безпеки, типи загроз, класифікацію шкідливого програмного забезпечення (ШПЗ) та сучасні методи захисту від програмних атак. Сформульовано вимоги до захисної системи, спроєктовано її архітектуру, обґрунтовано вибір методів шифрування та створено базу даних з урахуванням вимог конфіденційності. Розроблено функціональну систему, що включає модулі автентифікації, збереження даних, захисту від SQL-ін'єкцій, логування дій користувачів і зберігання попереджень. Реалізація виконана на основі PHP і MySQL та супроводжується документацією – ER-діаграмами, таблицями й тестовими сценаріями. Запропоноване рішення забезпечує захист персональних даних, виявлення загроз і фіксацію інцидентів безпеки відповідно до чинного законодавства України щодо захисту інформації.

25.05.2025 _____

# ABSTRACT

Theme of qualification work: Information Resource Protection System of an Internet Service Provider Against Malware.

Author of the work: Nechyporuk Mykhailo Viktorovych

Mentor: Ph.D. Cheshun Viktor Mykolaiovych

Total volume of work: 70 pages, 2 appendices, 14 figures, 15 tables, 40 references.

Graphic Materials: 3 posters, 10 presentation slides.

Keywords: information security, internet service provider, database, hashing, encryption

This bachelor's qualification thesis focuses on the study of theoretical principles, design, and implementation of protection elements for information resources within the environment of an internet service provider. The work emphasizes the issue of malicious software (malware) spreading within provider infrastructure and explores methods for its detection and neutralization.

The thesis examines the fundamental concepts of information security, types of threats, malware classification, and modern protection methods against software attacks. It formulates system requirements, designs the system architecture, justifies the choice of encryption methods, and develops a database in accordance with confidentiality requirements. A functional system was developed that includes modules for authentication, data storage, SQL injection protection, user activity logging, and warning storage. The system was implemented using PHP and MySQL and is accompanied by documentation including ER diagrams, tables, and test scenarios. The proposed solution ensures the protection of personal data, threat detection, and security incident logging in accordance with current Ukrainian legislation on information protection.

25.05.2025 

ЗМІСТ



| | | | | | КРБКБ.2102158.21.02.36 ПЗ | | |
|---|---|---|---|---|---|---|---|
| Зм. | Арк. | № докум. | Підпис | Дата | | | |
| Розробив | | Нечипорук М.В | | 25.05.25 | Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення<br>Пояснювальна записка | Літера: Н | Аркуш: 6 / Аркушів: 70 |
| Перевірив | | Чешун В.М | | 10.06.25 | | | |
| Н.контр. | | Мостовий С.В. | | 16.06.25 | | ХНУ, КБ-21-2 | |
| Затвер. | | Кльоц Ю.П. | | 16.06.25 | | | |

# ВСТУП

У сучасних умовах інтенсивного розвитку інформаційних технологій та глобальної цифровізації особливу актуальність набуває проблема захисту інформаційних ресурсів. Інтернет-провайдери як ключові учасники цифрової інфраструктури забезпечують безперервний доступ до мережевих сервісів, виступаючи при цьому важливими суб'єктами інформаційної безпеки. Водночас, їхні інформаційні ресурси дедалі частіше стають об'єктами цілеспрямованих кібератак, головною метою яких є порушення конфіденційності, цілісності та доступності критичних даних. Одним із найпоширеніших інструментів атак є шкідливе програмне забезпечення, яке постійно еволюціонує та ускладнює завдання захисту інформаційного середовища.

Складність забезпечення кібербезпеки інтернет-провайдера полягає у необхідності одночасного захисту великої кількості вузлів, користувацьких даних, каналів зв'язку, а також у відповідності нормативним вимогам і стандартам. Особливу загрозу становить шкідливе програмне забезпечення, здатне несанкціоновано збирати, змінювати чи знищувати інформацію, використовувати системні ресурси або створювати бекдори для подальших атак.

У зв'язку з цим, розробка ефективної, адаптивної та надійної системи захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення є надзвичайно важливим та актуальним завданням.

Метою цієї кваліфікаційної роботи є дослідження загроз, пов'язаних із шкідливим програмним забезпеченням, та розробка програмно-апаратної системи захисту інформаційних ресурсів інтернет-провайдера, зокрема користувацьких баз даних.

В роботі вирішено такі завдання:

– здійснено аналіз основних загроз інформаційній безпеці в середовищі інтернет-провайдерів;

– вивчено типи шкідливого програмного забезпечення та їхній вплив на інформаційні системи;

– розглянуто сучасні підходи та інструменти захисту від програмних загроз;

– спроєктовано архітектуру захисної системи з урахуванням особливостей інфраструктури провайдера;

– реалізовано прототип системи з механізмами шифрування персональних даних користувачів;

– додано базові засоби фільтрації небезпечних запитів на рівні веб-інтерфейсу;

– виконано тестування системи з подальшим аналізом її ефективності.

Об'єктом дослідження є процес захисту інформаційних ресурсів інтернет-провайдера.

Предметом дослідження виступають методи і засоби протидії шкідливому програмному забезпеченню в умовах провайдерської інформаційної інфраструктури.

Практична цінність роботи полягає у створенні функціональної моделі захисної системи, яку можна інтегрувати у реальні середовища діяльності інтернет-провайдерів.

# 1 ТЕОРЕТИЧНІ АСПЕКТИ ЗАХИСТУ ІНФОРМАЦІЙНИХ РЕСУРСІВ ІНТЕРНЕТ-ПРОВАЙДЕРА

## 1.1 Основні поняття інформаційної безпеки

Інформаційна безпека (ІБ) у сучасному цифровому середовищі виступає фундаментальною складовою національної безпеки, стратегічної стійкості держави та стабільного функціонування комерційних і державних інформаційно-телекомунікаційних систем. З урахуванням стрімкого розвитку цифрових технологій та зростання залежності критичних секторів від інформаційних систем, забезпечення цілісності, конфіденційності та доступності інформації набуло статусу пріоритетного напряму державної політики в сфері кібербезпеки [23].

В українському правовому полі інформаційна безпека визначається як стан захищеності інформації та інформаційних ресурсів, при якому унеможливлюється їх несанкціонований доступ, порушення цілісності або блокування доступу до них, зокрема в межах інформаційно-телекомунікаційних систем. Закон України «Про захист інформації в інформаційно-телекомунікаційних системах» встановлює організаційні та технічні вимоги до функціонування систем захисту, зобов’язує суб’єктів критичної інфраструктури впроваджувати сертифіковані засоби захисту, вести облік інцидентів безпеки та регулярно здійснювати аудит [23].

Сутність поняття ІБ не обмежується лише технічним аспектом  вона є комплексною категорією, що охоплює правову, соціальну, організаційну, інформаційну та технічну компоненти. У межах функціонування інтернет-провайдерів, які забезпечують передачу даних на транспортному рівні та виступають посередниками між інформаційними ресурсами та кінцевими користувачами, ІБ включає захист каналів зв’язку, серверного обладнання, маршрутизаторів, DNS-серверів, логів, а також інформації про користувачів, у тому числі персональних даних. У цьому контексті важливим є дотримання не лише внутрішньої політики ІБ, а й відповідність вимогам міжнародних стандартів, зокрема ISO/IEC 27001.

Сучасна наукова думка трактує ІБ як динамічний стан інформаційної системи, що забезпечується шляхом безперервного моніторингу ризиків, впровадження адаптивних систем реагування на інциденти, багаторівневого шифрування даних, а також забезпечення фізичного захисту серверного обладнання. Особливістю інформаційної безпеки в інтернет-сервісах є високий ступінь відкритості середовища, велика кількість точок доступу до мережі та взаємодія з численними зовнішніми вузлами, що значно ускладнює процеси ідентифікації загроз і потребує використання складних систем виявлення аномальної активності [17].

Ключовими властивостями інформації, які мають бути гарантовані в межах системи ІБ, є:

- конфіденційність – забезпечення доступу до інформації лише для уповноважених суб'єктів;
- цілісність – гарантування відсутності несанкціонованих змін, випадкових або зловмисних пошкоджень даних;
- доступність – можливість безперервного отримання інформації авторизованими користувачами в необхідний час.

У деяких класифікаціях до основних властивостей також додають незаперечність (незаперечення факту дій з боку суб'єкта) та автентичність (перевірка відповідності джерела інформації) [17].

Питання інформаційної безпеки активно досліджується у вітчизняній науковій літературі. Жилін А. В. та співавт. зазначають, що ефективна система ІБ повинна містити в собі як організаційні, так і апаратно-програмні елементи, доповнені політикою безпеки, інструкціями користувачів та технічними протоколами реагування на інциденти [8, с. 38–42].

Водночас сучасні тенденції вказують на зміщення акцентів із суто захисних технологій у бік активного реагування: застосування систем виявлення вторгнень (IDS), засобів аналізу поведінкових шаблонів користувачів, впровадження Zero Trust моделей та використання засобів штучного інтелекту для передбачення атак.

Такий перехід обумовлений як зростанням складності загроз, так і недоліками традиційних засобів захисту у виявленні складних, цілеспрямованих атак (APT) [32].

Крім того, важливим напрямом досліджень є класифікація загроз інформаційній безпеці. Зокрема, Харченко С. О. у своїй праці виокремлює такі категорії загроз:

- за джерелом походження – зовнішні та внутрішні;
- за характером – навмисні (цілеспрямовані) та випадкові (викликані технічними помилками, збоями);
- за масштабом – локальні, регіональні, глобальні;
- за наслідками – незначні, суттєві, критичні [29, с. 193].

Ця класифікація дозволяє не лише систематизувати ризики, а й визначити пріоритети в побудові системи інформаційного захисту, особливо в масштабі інфраструктури провайдера, що обслуговує велику кількість абонентів та має справу з чутливою інформацією, зокрема персональними даними, платіжною інформацією тощо.

Таким чином, у межах цифрової трансформації та зростаючих кіберзагроз, поняття інформаційної безпеки слід розглядати як цілісну систему технічних, організаційних і правових заходів, що діють узгоджено та адаптивно відповідно до поточних умов. Для інтернет-провайдерів, які виконують критичну функцію у передачі, зберіганні та обробці даних, впровадження ефективної системи інформаційної безпеки є не лише технічною вимогою, а необхідною умовою конкурентоспроможності та правової відповідальності.

Окрім технічних і нормативних засад, важливим компонентом сучасної концепції інформаційної безпеки є інформаційна культура та рівень обізнаності користувачів і персоналу з основними правилами захисту інформації. Низький рівень підготовки співробітників часто є причиною інцидентів – від розголошення облікових даних до відкриття фішингових вкладень, що спричиняють зараження

всієї мережі. З огляду на це, навчання персоналу основам кібергігієни стає частиною політики безпеки компаній, включно з інтернет-провайдерами.

У контексті кіберзагроз особливе місце займає питання відповідальності – як юридичної, так і операційної. Компанії, що обробляють персональні дані, відповідно до українського законодавства та міжнародних норм (GDPR, ISO/IEC 27001), зобов'язані не лише запровадити захисні механізми, а й документально підтверджувати їх застосування, зберігати лог-файли дій, мати план реагування на інциденти (IRP) і регулярно проводити аудити.

З технічної точки зору, на сьогодні ключовими принципами побудови систем ІБ є:

- принцип найменших привілеїв (Least Privilege) – користувачі та процеси повинні мати лише той рівень доступу, який необхідний для виконання їхніх функцій;
- захист за глибиною (Defense in Depth) – використання декількох рівнів захисту: від фаєрволів до шифрування, логування та виявлення вторгнень;
- модель Zero Trust – кожна дія або запит повинні вважатися потенційно небезпечними, незалежно від їхнього джерела;
- аудит і логування – постійне збирання та аналіз дій користувачів і систем для своєчасного виявлення аномалій.

Інтернет-провайдери, як оператори критичної інфраструктури, відіграють ключову роль у забезпеченні національної інформаційної безпеки. Саме через їхні мережі проходить більшість користувацького трафіку, що робить провайдерське середовище ціллю як для масових атак (ботнети, фішинг, DDoS), так і для цілеспрямованих дій (APT, проникнення до вузлів авторизації або маршрутизаторів). З цього погляду ІБ для провайдера – це не тільки захист власної інфраструктури, але й активна участь у формуванні безпечного цифрового простору для клієнтів, бізнесу та держави.

Крім того, сучасні виклики – гібридні війни, кібершпигунство, зловживання персональними даними – вимагають нових підходів до оцінки ризиків. ІБ більше

не є лише технічною дисципліною, вона охоплює стратегії цифрової стійкості (resilience), політику взаємодії з державними органами, а також питання цифрового суверенітету.

Таким чином, поняття інформаційної безпеки сьогодні слід розглядати у ширшому, міждисциплінарному контексті – як невід'ємну складову стратегічного управління ризиками в умовах цифрового суспільства, де інтернет-провайдери виступають як провідники і як захисники водночас. Системний, комплексний та адаптивний підхід до побудови ІБ стає основою для сталого розвитку інформаційної інфраструктури будь-якої організації.

## 1.2 Інформаційні ресурси інтернет-провайдера: структура та значення

Інформаційні ресурси (ІР) інтернет-провайдера є сукупністю даних, які створюються, зберігаються, обробляються, передаються або використовуються в межах діяльності компанії, що надає доступ до мережі Інтернет. Вони охоплюють не лише внутрішні корпоративні дані, а й персональну, технічну, аналітичну та комерційну інформацію, пов'язану з клієнтською базою, інфраструктурою доступу та взаємодією з мережевими вузлами.

Згідно з нормативним підходом, до інформаційних ресурсів відносяться дані, які мають цінність як об'єкт управління і є критичними для виконання основних функцій підприємства [23]. У випадку інтернет-провайдерів ІР набувають статусу об'єктів підвищеного рівня ризику через те, що їх втрата чи модифікація безпосередньо впливає на доступність сервісів, правову відповідальність та репутацію компанії. Сукупність інформаційних ресурсів які зазначенні на рисунку 1.1.

Інформаційні ресурси, які обробляються або зберігаються в межах інфраструктури інтернет-провайдера, є об'єктами постійної загрози як з боку

зовнішніх кіберзловмисників, так і з боку потенційних внутрішніх порушників – працівників компанії з надлишковими правами доступу.


Рисунок 1.1 – Інформаційні ресурси інтернет-провайдера

Саме тому одним із основних принципів управління інформаційними ресурсами є контроль доступу, поділ прав та обов'язкове журналювання дій користувачів [8].

Особливу увагу слід приділяти клієнтським базам даних. Вони не лише зберігають ідентифікаційні дані, але й містять інформацію, яка відповідно до законодавства підлягає обов'язковому захисту згідно з Законом України «Про захист персональних даних». Порушення конфіденційності цієї категорії інформації тягне за собою не лише фінансову, а й кримінальну відповідальність [23].

Провайдерські інформаційні системи мають складну розподілену архітектуру, що включає кілька рівнів обробки інформації – від периферійного обладнання до центральних серверів обліку та аналітики. Це ускладнює забезпечення єдиної політики безпеки і вимагає застосування принципів сегментації мережі, багаторівневого шифрування та обов'язкового використання засобів аутентифікації з двома факторами (2FA).

У наукових джерелах вказується, що втрата, модифікація або блокування доступу до критичних інформаційних ресурсів є однією з основних причин дестабілізації роботи інтернет-провайдерів. За даними досліджень, понад 60% випадків тривалих відмов у наданні послуг пов’язані саме з порушенням цілісності інформаційних активів або результатами цілеспрямованих атак, пов’язаних із шкідливим ПЗ [36].

Таким чином, інформаційні ресурси інтернет-провайдера є не лише технічними артефактами, а важливими активами, які потребують системного обліку, класифікації, шифрування та захисту на рівні як політики компанії, так і державного регулювання. Побудова ефективної системи управління інформаційними ресурсами повинна базуватися на принципах безперервного моніторингу, відповідності стандартам безпеки, а також чіткої регламентації доступу до даних усіх рівнів.

У структурі інформаційних ресурсів інтернет-провайдера важливо виділити не лише їх типи, але й рівень критичності. Ресурси, які напряму пов’язані з обслуговуванням клієнтів (наприклад, DHCP-логи, бази обліку, сервіси аутентифікації), мають високий пріоритет захисту. Їх порушення може спричинити масові відмови у наданні послуг, порушення угод SLA (Service Level Agreement), втрату ліцензії на провайдерську діяльність або накладення штрафних санкцій з боку регуляторних органів.

У науковій літературі відзначено, що інформаційні ресурси в телекомунікаційному секторі класифікуються також за доступністю: відкриті, внутрішні (обмеженого доступу), конфіденційні та критично секретні [8, с. 73]. Наприклад, внутрішні технічні журнали моніторингу не призначені для зовнішнього доступу, але при витоку вони можуть надати зловмиснику ключову інформацію про конфігурацію мережі. У свою чергу, витік персональних даних клієнтів чи ключів шифрування може кваліфікуватися як інцидент критичного рівня. Важливо враховувати, що інформаційні ресурси провайдера постійно змінюються – збільшується обсяг логів, змінюються ключі, оновлюються бази

даних. Це породжує необхідність у впровадженні політик життєвого циклу інформації, що включає створення, використання, архівування, резервне копіювання, знищення. Автоматизація цих процесів дозволяє знизити ризики людського фактора та зменшити навантаження на IT-відділ.

Ще однією важливою характеристикою IP є їх взаємозв'язок. Наприклад, маршрутизатори використовують таблиці, сформовані з даних DHCP-серверів; сервер автентифікації перевіряє запити за допомогою бази облікових записів; система білінгу формує звіти на основі логів з фаєрвола. Тому порушення в одній частині системи може мати каскадний ефект на всю інфраструктуру.

Серед найбільш поширених вразливостей у керуванні інформаційними ресурсами інтернет-провайдерів можна виділити:

- недостатній захист паролів до ключових систем;
- відсутність розмежування прав доступу;
- зберігання резервних копій у незашифрованому вигляді;
- відсутність контролю змін у конфігураційних файлах;
- слабкий контроль зовнішніх підключень (VPN, API).

У випадках, коли інфраструктура інтернет-провайдера охоплює декілька філій чи підрядників, важливим стає питання централізованого управління політиками безпеки. У таких ситуаціях доцільно використовувати системи типу SIEM (Security Information and Event Management), які дозволяють об'єднувати всі лог-файли, сповіщення та дії користувачів в єдину аналітичну платформу [36].

Окремо варто зазначити роль інформаційних ресурсів у питаннях інцидент-менеджменту. Всі вхідні запити, підозрілі дії, помилки доступу мають фіксуватися у відповідних логах для подальшого аналізу. Без належного збереження цих ресурсів неможливо здійснити ретроспективне розслідування інцидентів чи сформувати доказову базу для юридичних дій.

У підсумку, інформаційні ресурси інтернет-провайдера – це не лише технічна основа надання послуг, але й стратегічний актив, від якого залежить стабільність, репутація та відповідність законодавчим вимогам. Їх захист має

реалізовуватись на основі принципів цілісного управління, адаптивної безпеки та технологічної гнучкості відповідно до змін зовнішнього середовища.

## 1.3 Загрози безпеці та види шкідливого програмного забезпечення

Загрози інформаційній безпеці (ІБ) в інфраструктурі інтернет-провайдерів характеризуються високою динамікою, зростаючою складністю і здатністю порушувати критичні бізнес-процеси. Центральне місце серед сучасних загроз посідає шкідливе програмне забезпечення (ШПЗ), яке цілеспрямовано атакує як мережеві вузли, так і сервери, бази даних, канали обміну та інші інформаційні ресурси провайдера.

ШПЗ (англ. malware) – це спеціально розроблене програмне забезпечення, що виконує дії, які завдають шкоди системі або її користувачеві, порушуючи конфіденційність, цілісність чи доступність даних [7]. ШПЗ здатне діяти приховано, модифікувати легітимні процеси, використовувати вразливості програмного забезпечення та залишатися непоміченим протягом тривалого часу, що особливо небезпечно в умовах роботи інтернет-провайдера.

До загроз інформаційній безпеці, що супроводжують використання шкідливого ПЗ, належать:

- крадіжка або підміна персональних та технічних даних;
- блокування доступу до ресурсів (DoS, ransomware);
- порушення коректної роботи серверів авторизації, білінгу, маршрутизаторів;
- проникнення у внутрішні системи з метою шпигунства чи саботажу;
- компрометація ключів та сертифікатів безпеки.

Особливої актуальності набуває класифікація шкідливого програмного забезпечення за типами дії та загрозами для інформаційних ресурсів інтернет-провайдера. У таблиці 1.1 узагальнено основні види ШПЗ, характерні для цієї

галузі. Серед особливо небезпечних тенденцій слід виділити появу шкідливого ПЗ як послуги (Malware-as-a-Service, MaaS).

Таблиця 1.1 – Основні види шкідливого програмного забезпечення, небезпечні для IP інтернет-провайдера

| Вид ШПЗ | Опис і механізм дії | Загроза для провайдера |
|---|---|---|
| Трояни | Маскуються під легальне ПЗ, дають зловмиснику прихований доступ. | Витік даних клієнтів, перехоплення логінів. |
| Черв'яки | Самостійно розмножуються мережею, поширюючись без участі користувача. | Перевантаження каналів, дестабілізація вузлів. |
| Руткіти | Впроваджуються в ядро ОС, приховуючи активність зловмисника. | Утруднення виявлення інфекції, тривала присутність. |
| Spyware Keyloggers | Фіксують дії користувачів, передають дані ініціатору атаки. | Витік внутрішньої інформації з адміністративних систем. |
| Backdoors | Створюють приховані шляхи доступу в систему. | Обхід автентифікації, непомітний контроль над системою. |
| Ransomware | Шифрують дані і вимагають викуп за їх розблокування. | Зупинка білінгу, втрати в репутації, порушення SLA. |
| Botnets | Заражені вузли виконують зовнішні атаки під контролем зловмисника. | Участь в DDoS, санкції через підозрілу активність мережі. |
| Exploit kits | Автоматично знаходять вразливості в системі та впроваджують інше ШПЗ. | Початковий етап багатьох атак, зараження ключових систем. |

Ця модель передбачає оренду або придбання готових інструментів атаки, що спрощує процес кіберзлочину навіть для технічно непідготовлених осіб [35]. Платформи MaaS включають панелі керування, генератори обфускації, автоматичні модулі шифрування тощо.

Іншою складністю є використання поліморфних вірусів, які змінюють власну структуру при кожному запуску, та використання штучного інтелекту зловмисниками для обходу систем поведінкового аналізу. Ці чинники суттєво ускладнюють застосування традиційних сигнатурних методів виявлення загроз.

Як підкреслюється у сучасних дослідженнях, ефективний захист інформаційних ресурсів інтернет-провайдерів від шкідливого ПЗ має ґрунтуватися на таких підходах:

- інтеграція моделей поведінкового аналізу;
- застосування аналітичних платформ SIEM;
- регулярне оновлення сигнатур антивірусного ПЗ;
- реалізація механізмів сегментації мережі;
- обмеження прав доступу за принципом найменших привілеїв [36].

У контексті захисту критичної інфраструктури важливою є не лише технологічна складова, а й політика безперервного моніторингу загроз, проведення аудиту безпеки, навчання персоналу та реалізація планів реагування на інциденти.

Крім класифікацій за технічними ознаками, в сучасному інформаційному середовищі важливим є контекст застосування шкідливого ПЗ. Зокрема, дедалі частіше атаки на інтернет-провайдерів є складовою гібридних загроз – поєднання технічного шкідливого впливу з інформаційно-психологічними кампаніями, що спрямовані на дестабілізацію регіонів, порушення зв’язку в критичних ситуаціях або спричинення економічних збитків.

Однією з поширених тактик є ланцюгові атаки (attack chain), коли зловмисник послідовно проходить кілька етапів:

1. Первинне проникнення через фішинг або незахищений вебінтерфейс.

2. Використання вразливостей (exploit) для підняття привілеїв.

3. Встановлення бекдорів, закріплення в системі.

4. Вилучення даних або шифрування серверів з подальшими вимогами викупу (ransomware).

Такі багатоступеневі загрози складно виявити традиційними антивірусами, адже кожен компонент атаки може бути не шкідливим сам по собі, але в комбінації – становити суттєву загрозу. Прикладом подібної тактики є атака на найбільші європейські телеком-холдинги в 2022–2023 роках, коли через зараження центрального білінгового сервера було скомпрометовано десятки тисяч облікових записів користувачів [37].

Особливу небезпеку становить використання легітимного програмного забезпечення в злочинних цілях – так званий Living off the Land (LotL). У цьому випадку зловмисник використовує наявні в системі інструменти адміністрування (наприклад, PowerShell, WMI або PHP-розширення) для прихованого запуску скриптів, обходу контролю доступу та прихованого зв’язку із зовнішніми C2-серверами (Command and Control). Внаслідок цього виявлення таких атак ускладнюється, а журналювання часто не спрацьовує – адже команди формально виконуються авторизованими процесами.

Крім технічних векторів, істотною загрозою є внутрішні порушення (insider threats), коли доступ до ресурсів отримують недобросовісні співробітники або колишні працівники, що зберегли облікові дані. Такі дії часто непомітні, бо виконуються в рамках легітимних сесій і можуть призводити до витоку баз даних, викрадення конфігурацій чи підключення до «дзеркальних» DNS-серверів.

Окрему категорію становлять загрози, пов’язані з використанням IoT-пристроїв (маршрутизаторів, модемів, камер спостереження). Через слабкий захист (заводські паролі, відсутність оновлень) ці пристрої часто стають частиною бот-мереж, які здійснюють DDoS-атаки на зовнішні сервіси або створюють бекдори у внутрішню мережу провайдера [36].

У межах протидії шкідливому ПЗ та іншим загрозам інтернет-провайдери повинні:

– здійснювати співпрацю з державними органами, зокрема CERT-UA, НКЕК, Держспецзв'язком, для обміну даними про вразливості, участі у тренуваннях і кібернавчаннях;

– застосовувати індикатори компрометації (IoC) – шаблони ознак атак, що дозволяють системам SIEM автоматично виявляти нетипові події;

– реалізовувати набір політик відновлення (Disaster Recovery Plans, DRP), які передбачають покрокове відновлення критичних сервісів після успішного втручання.

Таким чином, ефективна протидія загрозам інформаційній безпеці в умовах інфраструктури інтернет-провайдера потребує багатошарового захисту, використання як технологічних засобів, так і адміністративних практик. Ключову роль у цьому відіграє поєднання аналітичного моніторингу, кіберосвіти персоналу, безперервного оновлення захисних систем і тісної взаємодії з кіберінституціями на національному рівні. Це дозволяє не лише запобігати атакам, а й оперативно реагувати на нові виклики, що постійно виникають у цифровому середовищі.

## 1.4 Основні методи та засоби захисту від шкідливого ПЗ

Захист від шкідливого програмного забезпечення є ключовим елементом системи інформаційної безпеки інтернет-провайдера. З огляду на масштаб, розгалуженість інфраструктури, відкритість каналів комунікації та взаємодію з великою кількістю клієнтів, провайдери потребують комплексної, багаторівневої системи протидії ШПЗ, що поєднує організаційні, програмні та технічні засоби.

Сучасна стратегія захисту інформаційного середовища від шкідливого ПЗ ґрунтується на принципах проактивності, багаторівневості та адаптивності. Це

означає, що ефективна система безпеки повинна не лише виявляти вже відомі загрози, але й запобігати появі нових, аналізувати поведінкові аномалії, а також оперативно реагувати на інциденти.

Для систематизації засобів захисту доцільно класифікувати їх за принципом типу впливу та рівня функціонування в системі (табл. 1.2).

Таблиця 1.2 – Класифікація методів і засобів захисту від шкідливого програмного забезпечення

| Тип засобів | Приклади реалізації | Рівень застосування |
|---|---|---|
| Превентивні (проактивні) | Актуалізація ПЗ, патч-менеджмент, обмеження прав доступу | Системне адміністрування, політики безпеки |
| Детекційні (виявлення загроз) | Антивірусне ПЗ, IDS/IPS-системи, SIEM | Сервери, мережеве обладнання, бази даних |
| Реактивні (відповідь на інциденти) | Ізоляція заражених вузлів, відновлення з резервних копій | Системи аварійного реагування (IR) |
| Криптографічні засоби | Шифрування даних, електронні підписи, SSL/TLS | Передача даних, бази клієнтів |
| Аналітичні засоби | Моніторинг активності, логування, аналіз поведінки користувачів | SOC/CSIRT, SIEM, машинне навчання |
| Організаційні заходи | Політика безпеки, аудит, навчання персоналу | Управління, служба ІБ |

Превентивні заходи передбачають мінімізацію потенційних вразливостей до їх експлуатації. Зокрема, регулярне оновлення програмного забезпечення, обмеження привілеїв доступу за принципом найменшої необхідності, сегментація

мережі та багатофакторна автентифікація є базовими елементами проактивного захисту [7].

Детекційні засоби включають традиційне антивірусне ПЗ з сигнатурним аналізом, а також сучасні системи виявлення вторгнень (IDS/IPS), які працюють на основі поведінкових алгоритмів. Поширеними є також гібридні системи з використанням машинного навчання для виявлення аномальної активності, характерної для новітнього ШПЗ [36].

Реактивні засоби орієнтовані на обмеження поширення загрози та швидке відновлення нормальної роботи системи. Це може бути автоматична ізоляція зараженого вузла, переведення інфраструктури у резервний режим, активація Disaster Recovery Plan (DRP), або повна заміна пошкоджених компонентів.

Криптографічні методи дозволяють захистити інформацію навіть у випадку її перехоплення або несанкціонованого доступу. Шифрування даних у базах користувачів, SSL-захист веб-інтерфейсів, а також застосування електронних цифрових підписів є обов'язковими компонентами сучасного захисту інтернет-провайдерів [25].

Аналітичні засоби, як-от системи централізованого моніторингу (SIEM), дозволяють акумулювати дані з різних джерел, проводити їх кореляцію та формувати сповіщення про потенційні атаки. Такі засоби часто поєднують логування, контроль відповідності, поведінкову аналітику та підтримку інцидент-менеджменту.

Крім технічних засобів, вкрай важливою залишається організаційна складова: чітка політика безпеки, інструкції користувачів, регулярний аудит та навчання персоналу. Згідно з дослідженнями, понад 60% інцидентів виникають через людський фактор – помилки конфігурації, використання слабких паролів, або нехтування елементарними правилами гігієни інформації [36].

Таким чином, ефективний захист від шкідливого програмного забезпечення не може бути забезпечений лише за рахунок однієї технології. Він має бути системним, адаптивним, і персоналізованим під особливості інфраструктури

провайдера. Поєднання превентивних, детекційних, реактивних, криптографічних, аналітичних і організаційних заходів дозволяє формувати стійку до загроз систему захисту, яка відповідає вимогам часу.

Сучасні тенденції у сфері протидії шкідливому ПЗ свідчать про зростання ролі інтегрованих підходів – тобто таких, що поєднують технічні, аналітичні та організаційні компоненти в єдину функціональну систему. Це особливо актуально для інтернет-провайдерів, які одночасно обслуговують інфраструктурні ресурси, кінцевих користувачів і корпоративних клієнтів. У таких умовах захисні механізми повинні охоплювати не лише точки входу в систему, але й внутрішній трафік, API, протоколи зв'язку та адміністрування.

Значну увагу також приділяють поведінковому аналізу, що здійснюється не лише в межах SIEM, а й за допомогою систем UEBA (User and Entity Behavior Analytics). UEBA дозволяє фіксувати нетипові дії користувачів, наприклад: вхід у незвичний час, перевищення кількості запитів до БД, багаторазові помилки входу. Це дозволяє виявляти шкідливі дії навіть у випадку, якщо вони виконуються з легітимного акаунта, але із зміненим патерном поведінки.

Окрему нішу у захисті інфраструктури займають засоби мікросегментації, які реалізуються через програмно-окреслену мережу (SDN) або через мережеві політики на рівні фаєрволів та VLAN. Така стратегія дозволяє обмежити переміщення шкідливого ПЗ у разі компрометації окремого вузла – замість того, щоб ізольовувати всю мережу, система обмежує лише доступ між модулями.

У практиці провайдерів зростає використання sandbox-систем – ізольованих середовищ для запуску підозрілих скриптів і файлів. Наприклад, надходження вкладень через внутрішню пошту може спочатку оброблюватися в sandbox, де проводиться симуляція виконання без шкоди для основної інфраструктури. Це дозволяє виявити zero-day атаки або змінені версії наявного шкідливого ПЗ.

Доцільним є впровадження засобів обфускації вихідного коду веб-інтерфейсів, шифрування внутрішніх API-запитів, а також контрольованого ротаційного збереження логів. Це мінімізує ризик витоку структурної інформації

про систему (наприклад, через витік логів), а також унеможливлює зворотну інженерію клієнтських додатків.

Іншим важливим напрямом є застосування Threat Intelligence – сервісів, які надають зведення про поточні загрози, уразливості та активні кампанії з розповсюдження шкідливого ПЗ. Провайдери можуть інтегрувати ці дані в антивірусні платформи, налаштування фаєрволів, фільтри DNS та системи контролю пошти, таким чином підвищуючи реактивність системи безпеки.

Серед рекомендованих практик захисту в реальному середовищі провайдера:

- використання honeypot-серверів для виявлення зовнішніх сканувань;
- застосування бази доменів із репутаційними оцінками (DNSBL, RPZ) для фільтрації фішингових або заражених сайтів;
- блокування типових протоколів атаки на рівні шлюзу (наприклад, SMB, Telnet, які використовуються для поширення черв'яків);
- впровадження систем активного реагування, які в режимі реального часу блокують IP-адреси при виявленні підозрілої активності.

Особливої уваги потребує контроль безпеки віддаленого адміністрування, адже у разі компрометації системи керування доступом зловмисник може отримати повний контроль над вузлами. Використання VPN, двофакторної автентифікації, обмеження доступу за IP-адресами та ведення окремого журналу дій адміністратора – це мінімальні вимоги до безпечного керування мережею провайдера.

Також актуальним є створення стратегій безпечного оновлення ПЗ – перевірка цифрових підписів оновлень, використання внутрішніх репозиторіїв і попереднє тестування змін у віртуалізованих середовищах. Це дозволяє уникнути несанкціонованих оновлень, через які може бути впроваджене модифіковане або заражене ПЗ.

Таким чином, захист від шкідливого ПЗ в інтернет-провайдерському середовищі потребує комплексного підходу, що охоплює не лише класичні

засоби, а й сучасні інструменти моніторингу, поведінкового аналізу, ізоляції середовища, а також кіберрозвідки. Побудова ефективної системи ІБ можлива лише за умов безперервного оновлення, високого рівня автоматизації та участі людського ресурсу в аналізі критичних інцидентів. Усе це формує основу стійкої цифрової інфраструктури, здатної протистояти сучасним кіберзагрозам.

## 1.5 Постановка задачі

У першому розділі було системно досліджено теоретичні основи забезпечення інформаційної безпеки (ІБ) інтернет-провайдерів, визначено структуру їхніх інформаційних ресурсів, класифіковано загрози безпеці та шкідливе програмне забезпечення, а також проаналізовано сучасні підходи й засоби захисту.

Розглянуто поняття інформаційної безпеки в контексті національної, комерційної та інфраструктурної стійкості. Встановлено, що ІБ є багатовимірним явищем, яке охоплює правові, організаційні та технічні аспекти. Особливу увагу приділено специфіці безпеки в інтернет-провайдерських структурах, де середовище характеризується високим ступенем відкритості, розподіленою архітектурою та великою кількістю взаємодіючих компонентів. Виявлено, що захист інформаційних активів провайдерів має бути адаптивним, багаторівневим і відповідати міжнародним стандартам, зокрема ISO/IEC 27001.

У результаті аналізу інформаційних ресурсів інтернет-провайдера встановлено, що вони охоплюють як технічні (логи, таблиці маршрутизації, конфігураційні файли), так і персональні дані клієнтів, комерційну аналітику, облікову та білінгову інформацію. Така широка і критична структура вимагає чіткого управління доступом, шифрування, резервного копіювання та створення централізованої системи безпеки. Доведено, що порушення цілісності або

доступності цих ресурсів може спричинити збої в наданні послуг, втрату ліцензії або юридичну відповідальність.

Окремо охарактеризовано загрози інформаційній безпеці, пов’язані зі шкідливим програмним забезпеченням. Встановлено, що інфраструктура інтернет-провайдера є привабливою мішенню для таких типів ШПЗ, як руткіти, трояни, шпигунські програми, ransomware, botnets, та exploit kits. Було виявлено, що основними цілями таких атак є несанкціонований доступ до даних, підміна інформації, дестабілізація роботи мережі та участь у зовнішніх атаках (наприклад, DDoS). Особливу небезпеку становлять сучасні тенденції, як-от використання MaaS (Malware-as-a-Service) та обфускованих або поліморфних загроз, що значно ускладнюють виявлення за допомогою традиційних сигнатурних методів.

У підрозділі 1.4 систематизовано основні методи та засоби захисту від шкідливого ПЗ, поділені на превентивні, детекційні, реактивні, криптографічні, аналітичні та організаційні. Обґрунтовано необхідність багаторівневої моделі захисту, де кожен етап – від оновлення ПЗ до впровадження SIEM-систем – формує єдину систему протидії загрозам. Підкреслено важливість людського чинника та організаційної складової (аудит, навчання, політики безпеки), які мають доповнювати технічні інструменти.

Крім того, у роботі враховано актуальні нормативні вимоги та рекомендації міжнародних стандартів у сфері захисту персональних даних, що дозволяє адаптувати систему до реальних умов експлуатації. Особливу увагу приділено забезпеченню сумісності із сучасними програмними платформами, розширюваності системи та можливості її масштабування під потреби різних типів інтернет-провайдерів. Передбачено можливість подальшої інтеграції з аналітичними модулями, що дозволяє підвищити ефективність виявлення загроз і оптимізувати процес реагування на інциденти безпеки.

У кваліфікаційній роботі проводиться розробка системи захисту інформаційних ресурсів інтернет провайдера. Опрацьовується підвищення рівня

інформаційної безпеки в інфраструктурі провайдера шляхом виявлення, запобігання та нейтралізації програмних загроз.

Щоб досягнути поставленної мети в роботі потрібно виконати наступне:

– провести проєктування системи захисту інформаційних ресурсів за вимогами захисної системи;

– розписати структуру бази даних користувачів послуг інтернет-провайдера та розбір архітектури системи;

– обрати методи шифрування персональних даних у БД та середовище розробки з інструментами;

– провести реалізацію бази даних користувачів з шифруванням даних;

– провести тестування системи та зробити оцінку ефективності.

В програмі зберігаються всі дані клієнтів у безпечному вигляді, виявляє небезпечні запити, журнал активності.Програма створена у вигляді бази даних. Створено sql-таблиці кожна має свої дані і функції, це вже детально розписано в самій роботі. Таким чином, теоретичне підґрунтя роботи дозволило сформувати уявлення про актуальні загрози, вимоги до безпеки та архітектурні особливості провайдерських інформаційних систем.

# 2 ПРОЄКТУВАННЯ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЙНИХ РЕСУРСІВ

## 2.1 Вимоги до захисної системи

Інтернет-провайдери обслуговують значну кількість абонентів, забезпечують критичні комунікаційні сервіси та накопичують великі обсяги чутливої інформації – від персональних даних користувачів до технічних конфігурацій мережі. Такий обсяг даних, поєднаний із розгалуженою архітектурою та відкритим характером взаємодії, створює широке поле для кібератак, зокрема із застосуванням шкідливого ПЗ, фішингових інструментів і зловмисного трафіку.

Зважаючи на це, виникає необхідність розробки універсальної захисної системи, що інтегрується в інформаційне середовище провайдера без порушення його архітектури, проте забезпечує надійний захист на ключових рівнях – баз даних, веб-запитів, обліку та моніторингу , які зазначені в таблиці 2.1

Таблиця 2.1 – Формулювання задачі захисної системи

| Компонент | Суть реалізації |
|---|---|
| Мета | Створення модульної системи захисту від шкідливого ПЗ для провайдерського середовища |
| Об’єкти захисту | Персональні дані користувачів, веб-запити, база даних, сервер авторизації |
| Ключові функції | Шифрування, контроль доступу, фільтрація запитів, журналювання, сповіщення |
| Сумісність | Робота в середовищі Apache (XAMPP), взаємодія з MySQL, PHP |
| Результат | Демонстрація захисту даних і запитів, виявлення аномалій, запобігання витокам |

У рамках цього завдання система має працювати як внутрішній програмний модуль або серверний фільтр, який виконує попередню перевірку трафіку, контролює дії користувача та забезпечує цілісність і конфіденційність даних у БД. Ключовою ознакою є її незалежність від основної логіки сайту – система має взаємодіяти з API або лог-файлами, не потребуючи радикальних змін у серверах авторизації чи білінгу та мати свою класифікацію , яка наведена в таблиці 2.2.

Таблиця 2.2 – Класифікація вимог до системи захисту

| Категорія | Основні вимоги |
|---|---|
| Функціональні | Шифрування персональних даних (AES), фільтрація запитів (SQLi, XSS), обмеження прав доступу |
| Безпекові | TLS для трафіку, збереження паролів у хеш-форматі (SHA-2), логування з IP і часом |
| Системні | Сумісність з XAMPP, робота з MySQL, безперебійність, мінімальні ресурси |
| Реактивні | Ізоляція підозрілих запитів, збереження логів для аналізу, повідомлення адміністратора |
| Аналітичні | Централізоване журналювання, можливість аудиту дій, спостереження за шаблонами активності |

Оцінка реалізації базується на тестових сценаріях, які передбачають атаки через запити (SQL-ін’єкції, підміна даних), спроби несанкціонованого доступу, перевірку захищеності шифрування та стабільність логування подій.

Окрім технічних характеристик, проєктована система має відповідати сучасним вимогам до зручності використання та ефективного адміністрування. Це передбачає наявність інтерфейсу для виведення основних параметрів безпеки, журналів подій, сповіщень про спроби порушень, а також механізмів зворотного зв’язку з боку адміністратора системи. Не менш важливим є забезпечення гнучкості в налаштуваннях: можливість додавання нових фільтрів, розширення

правил блокування запитів, а також адаптації до змін у структурі бази даних чи веб-інтерфейсу.

Варто зазначити, що особливу небезпеку для інтернет-провайдерів становлять складні комбіновані атаки, коли зловмисники використовують декілька векторів проникнення одночасно. Наприклад, через веб-форму може здійснюватися SQL-ін’єкція, яка дає доступ до БД, після чого – ініціюється шифрування таблиць за допомогою завантаженого скрипта. У таких випадках система повинна не лише фіксувати підозрілу активність, а й забезпечити оперативну реакцію – блокування запиту, сповіщення адміністратора, тимчасову ізоляцію сервісу або переведення його в захищений режим.

Крім внутрішніх вимог, проєктована система має враховувати зовнішні нормативні обмеження, які регламентують обробку персональних даних, зокрема Закон України «Про захист персональних даних» та міжнародні стандарти, як-от GDPR або ISO/IEC 27001. Це означає, що всі операції зі зберігання, передачі й обробки чутливої інформації мають бути обґрунтовані, логовані й технічно захищені.

Під час формування вимог важливо також враховувати особливості навантаження на систему. У типових умовах інтернет-провайдер може обробляти тисячі запитів на годину. Система захисту має функціонувати у реальному часі, без затримок, що впливають на швидкість відгуку веб-інтерфейсу або доступ до ресурсів користувача. У зв’язку з цим доцільним є застосування асинхронної обробки логів, попередньо згенерованих правил безпеки та кешування часто використовуваних шаблонів запитів.

Варто також передбачити функціональність резервного копіювання: регулярне створення бекапів бази даних і логів дозволяє мінімізувати втрати у разі успішної атаки або збою в роботі серверного середовища. У межах цієї системи резервні копії мають зберігатися в зашифрованому вигляді на окремому захищеному носії, що відповідає вимогам до побудови стійкої інформаційної інфраструктури.

Таким чином, запропонована система має охоплювати не лише виявлення та протидію загрозам, але й забезпечення загальної стійкості інформаційного середовища провайдера, контроль відповідності нормативним вимогам, а також підтримку ключових принципів управління безпекою: конфіденційності, цілісності, доступності та відповідальності.

## 2.2 Архітектура пропонованої системи

Проєктована система захисту інформаційних ресурсів інтернет-провайдера базується на принципах модульності, масштабованості та ізольованості функцій. Її архітектура має забезпечити цілісне функціонування в умовах обмеженого серверного середовища без порушення наявної інфраструктури. Основною метою архітектурного підходу є досягнення рівноваги між продуктивністю, безпекою та простотою інтеграції.

У структурному плані система передбачає три основні рівні: вхідний обробник запитів (Apache + PHP), рівень захисту та перевірки, а також рівень зберігання даних (MySQL). Такий підхід дозволяє відокремити функції прийому, перевірки та збереження, що полегшує тестування та модифікацію окремих модулів.

Функціональна логіка системи побудована за принципом наскрізної перевірки запиту: від моменту його надходження до остаточного збереження даних у базі. На початковому етапі вхідний трафік передається до серверу Apache, де обробляється PHP-скриптом. На цьому етапі проводиться валідація вхідних даних, початкове логування, фільтрація небезпечних конструкцій та ключових фраз, характерних для атак типу SQL-ін’єкція чи XSS. У разі виявлення загрозливого вмісту запит блокується, а інформація про інцидент фіксується у відповідному лог-файлі або таблиці подій у базі даних.

Якщо перевірка пройдена, скрипт переходить до етапу шифрування критичних полів (наприклад, ідентифікаційної інформації користувача). Зашифровані дані передаються на рівень зберігання, де у структурованій формі записуються до таблиці MySQL. Для реалізації шифрування використовується вбудована бібліотека OpenSSL із застосуванням алгоритмів, стійких до криптоаналізу, зокрема AES-256.

Журналювання вбудоване в логіку системи: усі дії з обробки запитів, доступу до чутливих даних або спроб обійти фільтрацію фіксуються разом з IP-адресою, міткою часу та параметрами запиту. Це дозволяє в подальшому здійснювати аналіз поведінки користувачів, виявляти аномалії та підозрілі активності, а також реагувати на інциденти відповідно до вимог політики безпеки.

Система не передбачає глибокого втручання в інфраструктуру провайдера: її компоненти легко інтегруються в існуюче середовище, не вимагають зміни API або структури баз даних.

Для зручності подання взаємозв'язків між компонентами систему доцільно подати у вигляді таблиці, яка може слугувати основою для побудови графічної моделі яка заначена на рисунку 2.1.

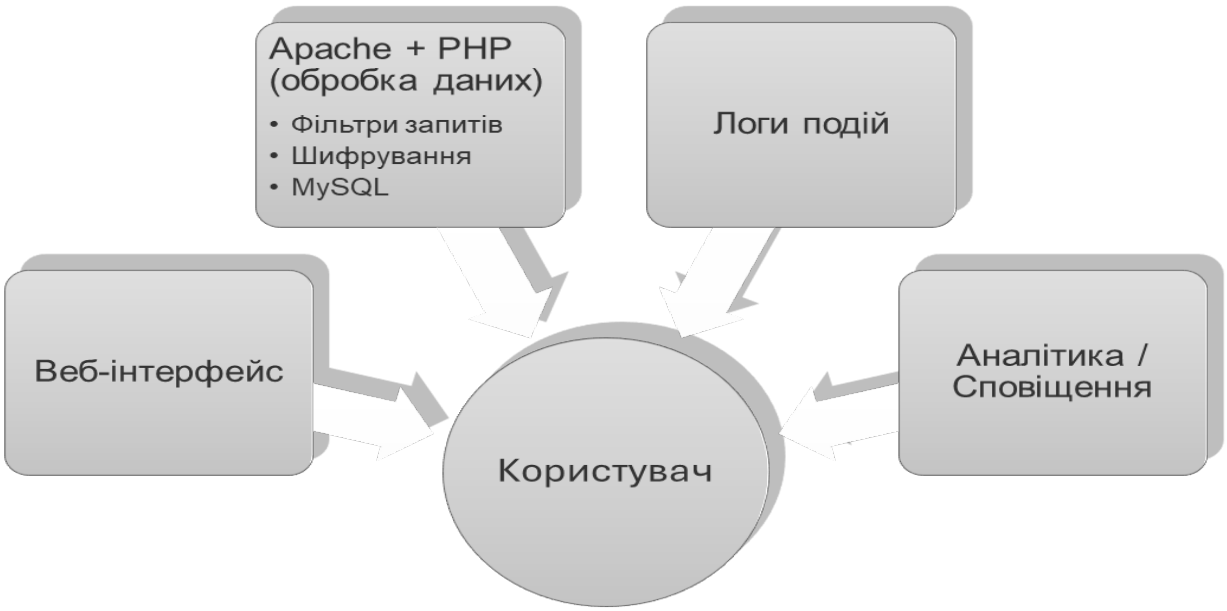



Рисунок 2.1. – Візуалізація архітектури

Більше того, передбачена гнучкість дозволяє масштабувати рішення – як у напрямі підвищення навантаження, так і з точки зору розширення функцій (наприклад, підключення систем аналітики чи зовнішніх систем виявлення вторгнень).

Візуалізація цієї архітектури може бути представлена у вигляді горизонтальної або каскадної схеми, де кожен компонент логічно пов’язаний із попереднім та наступним. Таке подання дозволяє чітко відобразити напрям руху даних та розмежувати зони контролю на кожному рівні.

У результаті запропонована архітектура дозволяє досягнути оптимального співвідношення між захищеністю системи, її функціональністю, а також простотою впровадження й подальшого супроводу. Вона відповідає сучасним вимогам до систем безпеки в умовах постійно зростаючих кіберзагроз, характерних для інтернет-провайдерів.

## 2.3 Структура бази даних користувачів послуг інтернет-провайдера

Зберігання облікових записів користувачів є важливою частиною будь-якої інформаційної системи, особливо якщо мова йде про захист даних інтернет-провайдера. Створення таблиці користувачів (users) забезпечує можливість автентифікації, збереження контактної інформації, фіксації дій у системі та застосування засобів безпеки.

Базу даних було спроєктовано з урахуванням вимог до конфіденційності, цілісності та надійності зберігання інформації. Основну роль у системі виконує таблиця users, структура якої наведена нижче.

Ця структура дає змогу надійно зберігати основні атрибути користувачів. Застосування хешування для паролів гарантує, що навіть у разі несанкціонованого доступу до бази, зловмисник не отримає відкритий пароль. Поле full_name за потреби може шифруватися (наприклад, за допомогою AES-алгоритму), що

дозволяє приховати персональні дані. Опис структури таблиці users показано в таблиці 2.3.

Таблиця 2.3 – Опис структури таблиці users

| Назва поля | Тип даних | Призначення |
|---|---|---|
| id | INT | Унікальний ідентифікатор користувача, первинний ключ |
| full_name | VARCHAR(255) | Ім’я та прізвище користувача (може зберігатися у зашифрованому вигляді) |
| email | VARCHAR(255) | Адреса електронної пошти або логін (унікальне значення) |
| password_hash | VARCHAR(255) | Хеш пароля користувача (наприклад, з використанням алгоритму SHA-256) |
| ip_address | VARCHAR(45) | IP-адреса користувача під час реєстрації або входу |
| created_at | TIMESTAMP | Дата й час створення облікового запису, автоматично генерується системою |

У таблиці users реалізовано унікальність для поля email, що унеможливлює повторну реєстрацію одного й того самого користувача. Для зручності моніторингу зберігається IP-адреса та час створення облікового запису [8; 15; 25].

Як показано на рисунку 2.2, структура таблиці users, створеної у середовищі phpMyAdmin, включає шість основних полів.

| # | Ім'я | Тип | Зіставлення | Атрибути | Нуль | За замовчуванням | Коментарі | Додатково | Дія |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(11) | | | Ні | Немає | | AUTO_INCREMENT | Змінити Знищити Більше |
| 2 | full_name | varchar(255) | utf8mb4_general_ci | | Ні | Немає | | | Змінити Знищити Більше |
| 3 | email | varchar(255) | utf8mb4_general_ci | | Ні | Немає | | | Змінити Знищити Більше |
| 4 | password_hash | varchar(255) | utf8mb4_general_ci | | Ні | Немає | | | Змінити Знищити Більше |
| 5 | ip_address | varchar(45) | utf8mb4_general_ci | | Так | NULL | | | Змінити Знищити Більше |
| 6 | created_at | timestamp | | | Ні | current_timestamp() | | | Змінити Знищити Більше |

Рисунок 2.2 – Структура таблиці users у середовищі phpMyAdmin

Поле id має ознаку первинного ключа (PRIMARY KEY) і автоматично збільшується при додаванні нового запису (AUTO_INCREMENT). Поля full_name, email, password_hash та ip_address зберігають відповідну текстову інформацію, тоді як поле created_at автоматично фіксує момент створення запису завдяки значенню current_timestamp().

Запропонована структура таблиці users дозволяє реалізувати основні вимоги до безпеки та функціональності у системі обліку користувачів послуг інтернет-провайдера:

- надійна ідентифікація – поле id забезпечує унікальність кожного користувача, виключаючи дублювання;
- конфіденційність – використання поля password_hash замість відкритого пароля гарантує захист у разі компрометації бази даних;
- контроль активності – поле ip_address фіксує джерело підключення, що дозволяє виявляти підозрілу активність;
- автоматичний облік часу – поле created_at автоматично зберігає дату й час створення облікового запису, що важливо для моніторингу подій;
- запобігання дублюванню – обмеження UNIQUE на полі email не дозволяє зареєструвати двох користувачів з однаковими електронними адресами.

Загалом така структура є оптимальною для реалізації системи автентифікації, що відповідає сучасним вимогам до збереження персональних даних та інформаційної безпеки

## 2.4 Вибір методу шифрування персональних даних у БД

Захист персональних даних у базах даних – один із найважливіших аспектів побудови інформаційно-безпекових систем. Для інтернет-провайдера, який зберігає облікові записи користувачів, зокрема їхні імена, електронні адреси та

паролі, є критично важливо забезпечити конфіденційність та недоступність цієї інформації для сторонніх осіб.

Персональні дані, які потребують захисту, умовно можна поділити на дві групи:

- паролі (секретні, односторонньо захищені дані);
- особисті ідентифікатори (ім’я та прізвище, IP-адреса тощо).

Паролі не підлягають зворотному відновленню, тому для їх збереження доцільно застосовувати хешування. У цій роботі використовується криптографічна функція SHA-256, що забезпечує незворотну трансформацію тексту у хеш-рядок фіксованої довжини.

У таблиці 2.4 наведено порівняння основних методів захисту, які розглядаються для застосування в системі.

Таблиця 2.4 – Порівняння методів захисту персональних даних

| Метод | Призначення | Можливість розшифрування | Переваги | Недоліки |
|---|---|---|---|---|
| SHA-256 | Хешування паролів | Немає | Незворотне, надійне | Не можна відновити пароль |
| AES-256 | Шифрування імен | Є | Можна розшифрувати при потребі | Потрібен захист ключа шифрування |
| Base64(не використовується) | Кодування, не шифрування | Так | Простота реалізації | Не забезпечує реального захисту |

Важливо, що однакові введення завжди дають однаковий результат, але відновити початковий текст за хешем неможливо без брутфорс-методів або словників, які ускладнюються застосуванням "солі" (random salt) [15; 25].

Особисті ідентифікатори, наприклад full_name, можуть бути збережені у відкритому вигляді, але для підвищення рівня конфіденційності варто застосовувати симетричне шифрування.

У системі пропонується використання алгоритму AES-256 (Advanced Encryption Standard). Він дозволяє як зашифрувати дані при збереженні, так і розшифрувати їх при зчитуванні – за наявності ключа [16].

Поняття криптографічного захисту охоплює цілий комплекс заходів, що спрямовані на запобігання несанкціонованому доступу до даних, їх підміні або викраденню. Найбільш ефективним напрямом сучасної криптографії є поєднання хешування (для незворотного зберігання) та симетричного шифрування (для конфіденційного, але контрольованого доступу до інформації).

У практиці застосування таких підходів варто згадати, що:

– державні органи, включаючи Міністерство цифрової трансформації України, використовують SHA-2 як стандарт для хешування паролів у реєстрах;

– банківські інформаційні системи застосовують симетричне шифрування для зберігання ПІБ клієнтів, історії операцій та адресних даних;

– хмарні сервіси, такі як AWS, Google Cloud або Azure, реалізують шифрування «на боці клієнта» саме за допомогою AES, а доступ до ключа захищено окремими механізмами.

Крім технічного аспекту, обрана модель шифрування відповідає Законодавству України, зокрема:

«Про захист інформації в інформаційно-телекомунікаційних системах» (Закон № 80/94-ВР), який передбачає обов’язкове впровадження технічних і програмних засобів для захисту персональних даних у базах, що мають обмежений доступ [23].

Однак, навіть правильний вибір алгоритмів не гарантує захищеність, якщо неправильно організоване зберігання ключа або відсутній контроль доступу до коду. Відомі випадки, коли шифрування виконувалось, але ключ знаходився у відкритому вигляді в коді PHP-файлів, що нівелювало весь захист.

З цієї причини у межах даної системи пропонується реалізувати окремий файл конфігурації, доступний лише серверу, з якого система зчитує ключ для AES. Крім того, доцільно обмежити права доступу до таблиць лише авторизованим користувачам з роллю адміністратора, що стане додатковим рівнем захисту.

З міркувань безпеки у цій системі SHA-256 застосовується до пароля, тоді як AES-256 використовується для імен користувачів. Алгоритми реалізуються засобами мови PHP: для хешування використовується функція hash(), а для шифрування – openssl_encrypt().

Перевагою такого підходу є гнучкість: при компрометації бази зловмисник не зможе відновити паролі, а особисті дані залишаться зашифрованими, якщо ключ зберігається окремо.

Таким чином, обрані методи шифрування дозволяють дотримуватись принципів інформаційної безпеки – конфіденційності, цілісності та автентичності, відповідно до стандартів сучасного захисту даних

## 2.5 Висновки до другого розділу

У межах другого розділу було здійснено постановку задачі, обґрунтування вимог, проєктування архітектури та вибір структурних елементів системи захисту інформаційних ресурсів інтернет-провайдера. Проведений аналіз, моделювання та логічне структурування підсистем забезпечили основу для подальшої практичної реалізації рішення, здатного протидіяти сучасним кіберзагрозам, зокрема шкідливому програмному забезпеченню.

На основі аналізу специфіки інформаційного середовища інтернет-провайдерів сформульовано комплексне технічне завдання щодо створення модульної, гнучкої та масштабованої системи захисту. Було визначено об’єкти захисту – персональні дані, веб-запити, бази даних, сервер авторизації – та сформовано перелік функціональних вимог, серед яких: застосування криптографічних засобів (SHA-256 і AES-256), контроль доступу, централізоване логування, фільтрація небезпечних запитів і оперативне сповіщення адміністратора.

Розроблена архітектура системи базується на принципах модульності та наскрізної перевірки запитів. Вона охоплює рівень приймання даних (Apache + PHP), рівень логіки безпеки (фільтрація, шифрування, автентифікація) та рівень зберігання даних (MySQL). Така модель дозволяє локалізувати функціональні зони відповідальності, полегшити діагностику та оновлення окремих компонентів системи без ризику для її загальної працездатності. Система передбачає механізми виявлення SQL-ін’єкцій, XSS-атак, аналіз шаблонів активності, автоматичне журналювання дій користувачів та створення повідомлень про інциденти.

У розділі також спроєктовано структуру бази даних, зокрема ключову таблицю users, яка включає поля з інформацією про ідентифікацію користувача, його IP-адресу, хеш пароля та мітку часу створення облікового запису. Було обґрунтовано вибір структури, що дозволяє досягти балансу між функціональністю, безпекою та масштабованістю.

Окрему увагу приділено вибору методів шифрування персональних даних. На основі аналізу криптографічних стандартів та практик безпеки в IT-сфері було запропоновано комбіноване застосування алгоритму SHA-256 для хешування паролів і AES-256 для шифрування відкритих полів, зокрема full_name. Здійснено порівняльну характеристику методів захисту, визначено їх переваги, недоліки та особливості застосування у межах реальної інфраструктури інтернет-провайдера.

Окрім зазначеного, важливо наголосити на здатності системи адаптуватися до змін у загрозовому ландшафті за рахунок гнучкої модульної структури, що дозволяє оперативно інтегрувати нові механізми захисту без потреби повного оновлення архітектури. Також передбачено можливість розширення функціоналу шляхом додавання нових підсистем, зокрема для аналізу трафіку в реальному часі та виявлення аномальної поведінки користувачів. Такий підхід посилює загальну стійкість системи до цілеспрямованих атак та забезпечує її актуальність у довгостроковій перспективі.

Розглянуто також питання відповідності проєктованої системи чинному законодавству України у сфері захисту інформації (Закон № 80/94-ВР, Закон «Про захист персональних даних»), а також міжнародним стандартам ISO/IEC 27001 та GDPR. Акцент зроблено на необхідності забезпечення не лише технічного, але й організаційного рівня захисту.

Загалом, результати розділу демонструють, що розроблена система:

- враховує сучасні вимоги до інформаційної безпеки;
- реалізована на основі відкритих технологій (PHP, MySQL, OpenSSL);
- забезпечує конфіденційність, цілісність, автентичність та доступність інформації;
- здатна масштабуватися та інтегруватися в типове серверне середовище інтернет-провайдера.

Отже, обґрунтована архітектура, правильно сформульовані вимоги та оптимальний вибір засобів захисту є підґрунтям для ефективного впровадження й подальшого тестування системи, що буде продемонстровано у наступному розділі.

# 3 РЕАЛІЗАЦІЯ ТА ДОСЛІДЖЕННЯ РОБОТИ СИСТЕМИ ЗАХИСТУ

## 3.1 Середовище розробки та використані інструменти (XAMPP, MySQL, PHP тощо)

Для реалізації системи захисту інформаційних ресурсів інтернет-провайдера було обрано локальне середовище розробки, яке забезпечує підтримку веб-технологій, мов серверного програмування, баз даних та зручного інтерфейсу для адміністрування. Розробка системи здійснювалася на персональному комп'ютері під керуванням операційної системи Windows 11, що дозволило швидко розгортати середовище без додаткових витрат на інфраструктуру.

У якості основної платформи для запуску веб-серверу та сервера баз даних було обрано програмний пакет XAMPP, який включає такі компоненти:

- Apache – веб-сервер, який забезпечує обробку HTTP-запитів;
- MySQL – система управління базами даних для зберігання інформації;
- PHP – мова серверного програмування для реалізації логіки обробки даних;
- phpMyAdmin – інструмент для візуального управління базами даних.

Установка XAMPP була виконана на локальний диск, що дозволило створити власне тестове середовище. При запуску програми XAMPP Control Panel було активовано служби Apache та MySQL, які працюють на портах 80, 443 (Apache) та 3306 (MySQL). Для адміністрування бази даних використовувався веб-інтерфейс phpMyAdmin, доступний за адресою http://localhost/phpmyadmin.

На рисунку 3.1 показано інтерфейс XAMPP Control Panel під час роботи системи. Активні модулі Apache та MySQL свідчать про те, що середовище успішно запущене та готове до обробки запитів та роботи з базами даних.


Рисунок 3.1 – Середовище запуску XAMPP Control Panel із запущеними модулями Apache та MySQL

XAMPP – це безкоштовний програмний пакет, який об’єднує веб-сервер Apache, систему керування базами даних MySQL, мову програмування PHP та інші корисні компоненти. У рамках цієї роботи XAMPP використовувався як основне середовище розробки, оскільки дозволяє швидко розгорнути локальний сервер без складної конфігурації. Усі служби запускалися через зручну XAMPP Control Panel, що значно спрощує керування модулями.

Apache HTTP Server – це програмний веб-сервер, який обробляє вхідні HTTP-запити та передає їх до обробника (PHP) або повертає файли клієнтам. У цьому проєкті Apache слугує серверною платформою, яка приймає запити з інтерфейсу користувача та передає їх для обробки у PHP-скриптах.

MySQL – реляційна система керування базами даних, яка була використана для створення бази даних isp_protection та збереження таблиць (users, logs, alerts тощо). MySQL дозволяє ефективно зберігати, фільтрувати й шукати інформацію,

а також забезпечує захист на рівні доступу до таблиць. Вибір цієї СКБД зумовлений її швидкістю, надійністю та підтримкою у XAMPP.

PHP – це серверна мова програмування, яка широко використовується для створення веб-додатків. У системі захисту PHP відповідає за логіку взаємодії між користувачем, базою даних та механізмами безпеки. Зокрема, на PHP реалізовано шифрування, хешування паролів, обробку форм авторизації, перевірку запитів на шкідливу активність.

phpMyAdmin – веб-інструмент для управління MySQL-базами даних. Він дозволяє створювати таблиці, змінювати структуру БД, переглядати вміст та виконувати SQL-запити. У даній роботі phpMyAdmin використовувався для створення таблиці users, її заповнення тестовими даними та перегляду результатів виконання скриптів.

Вибір середовища XAMPP та супутніх інструментів був зумовлений низкою переваг, які дозволили ефективно реалізувати систему захисту інформаційних ресурсів. По-перше, XAMPP забезпечує повноцінний стек для веб-розробки, включаючи веб-сервер, СКБД і засоби обробки серверної логіки. По-друге, його установка та налаштування не потребують спеціальних знань у галузі адміністрування серверів, що значно спрощує розробку та тестування.

Компоненти системи гармонійно взаємодіють між собою. PHP-скрипти, що відповідають за авторизацію та обробку запитів, безпосередньо звертаються до MySQL-бази через вбудовані засоби. Apache обробляє вхідні запити та формує відповідь, а phpMyAdmin дозволяє зручно переглядати та редагувати дані, що зберігаються в таблицях. Завдяки цій взаємодії розробник отримує повний контроль над усіма рівнями системи – від зберігання даних до логіки обробки запитів.

Таке середовище є не лише оптимальним для розробки у навчальних і дослідницьких цілях, але й широко застосовується у малих і середніх проєктах, що не потребують хмарної інфраструктури або складної масштабованості. В результаті воно дозволяє зосередитися на реалізації функціоналу захисту та

моделюванні загроз без зайвих витрат часу на налаштування системного оточення.

## 3.2 Реалізація бази даних користувачів з шифруванням даних

Для реалізації функціоналу автентифікації та зберігання інформації про користувачів у середовищі XAMPP було створено базу даних під назвою isp_protection. Вона містить таблицю users, структура якої наведена в підрозділі 2.3.

База даних створювалася за допомогою візуального інтерфейсу phpMyAdmin, який дозволяє працювати з MySQL без використання командного рядка. Після запуску локального сервера (Apache і MySQL), адміністратор переходить за адресою http://localhost/phpmyadmin, де створює нову базу даних та додає до неї таблицю users.

Для створення таблиці було використано SQL-запит, який наведено у Додатку Б (лістинг 1). У цьому запиті таблиця users містить шість полів. Поле id є первинним ключем і має властивість автоматичного збільшення значення (AUTO_INCREMENT). Поле email має обмеження UNIQUE, що унеможливлює реєстрацію кількох облікових записів з однаковими адресами електронної пошти.

Інші поля таблиці відповідають за збереження персональних даних (full_name), хешу пароля (password_hash), IP-адреси користувача (ip_address) та мітки часу (created_at), яка генерується автоматично.

Після створення таблиці users база даних isp_protection стала основою для реалізації подальших функцій захисту та автентифікації, які розглядатимуться у наступних підрозділах.

З метою забезпечення захисту даних у системі не допускається зберігання паролів користувачів у відкритому вигляді. Усі паролі проходять процес хешування – криптографічного перетворення, яке забезпечує незворотну зміну

значення. Це означає, що навіть у разі витоку бази даних зловмисник не зможе отримати справжні паролі.

У межах реалізованої системи було обрано алгоритм SHA-256, який належить до сімейства SHA-2 та широко використовується у сфері інформаційної безпеки. Цей алгоритм створює фіксований хеш-рядок довжиною 256 біт, незалежно від довжини вхідного тексту. Перевагою цього методу є швидкість обчислення та стійкість до колізій і атак типу «brute force».

Процедура хешування здійснюється на стороні сервера за допомогою мови PHP. При введенні пароля користувачем на формі реєстрації, система автоматично перетворює його у хеш перед збереженням у базу даних. Для цього використовується вбудована функція hash(), вказана на рисунку 3.2

```
$password = $_POST['password'];
$password_hash = hash('sha256', $password);
```

Рисунок 3.2 – PHP-код хешування пароля з використанням SHA-256

Отриманий хеш записується у поле password_hash таблиці users. Важливо відзначити, що функція hash() є незворотною, тому система не може відновити початковий пароль – лише порівнює хеш введеного значення з тим, що зберігається у базі.

Для підвищення рівня безпеки у подальших реалізаціях доцільно додати використання «солі» – випадкового значення, яке додається до пароля перед хешуванням. Це запобігає атакам із використанням таблиць підстановки (rainbow tables).

Таким чином, хешування паролів забезпечує базовий, але надійний рівень захисту автентифікаційних даних користувачів, що відповідає сучасним вимогам до безпеки веб-систем [15, 25].

На відміну від пароля, який зберігається у вигляді хешу, поле full_name може зберігатися у базі даних у відкритому вигляді. Однак для підвищення рівня конфіденційності персональної інформації у системі було реалізовано шифрування цього поля за допомогою алгоритму AES-256 (Advanced Encryption Standard).

Шифрування є процесом перетворення даних у зашифрований вигляд, який не можна прочитати без наявності ключа. У випадку з AES-256 використовується симетричний підхід – той самий ключ застосовується як для шифрування, так і для розшифрування.

У середовищі PHP реалізація шифрування виконується за допомогою функції openssl_encrypt(), яка дозволяє застосовувати різні алгоритми та режими шифрування. У рисунку 3.3 нижче наведено базовий варіант шифрування імені користувача перед збереженням у базу даних:

```
$key = "my_secure_key_256bit12345678901234567890123456"; // 32
символи (256 біт)
$iv                                                    =
openssl_random_pseudo_bytes(openssl_cipher_iv_lengt
h('AES-256-CBC'));
$encrypted_fullname  =  openssl_encrypt($full_name,
'AES-256-CBC', $key, 0, $iv);
```

Рисунок 3.3 – PHP-код шифрування повного імені користувача з використанням AES-256

У цьому прикладі:

- $key – ключ шифрування, який має зберігатися поза межами загальнодоступного коду, наприклад, у конфігураційному файлі з обмеженим доступом;

– $iv – вектор ініціалізації, який робить шифрування унікальним для кожного випадку (навіть якщо ім’я одне й те саме);

– openssl_encrypt() – функція, яка повертає зашифрований рядок.

Для коректної роботи системи, крім самого шифротексту, необхідно зберігати також $iv, або прикріпити його до зашифрованого тексту (наприклад, через base64_encode()), оскільки без нього розшифрування буде неможливим.

Шифрування особистих даних користувачів дозволяє гарантувати, що навіть у разі несанкціонованого доступу до бази, інформація, яка зберігається у полі full_name, залишиться нерозбірливою без наявності ключа.

Таким чином, використання AES-256 у поєднанні з правильним зберіганням ключа та вектора ініціалізації відповідає сучасним вимогам конфіденційності та забезпечує надійний рівень захисту персональних даних.

Для демонстрації роботи створеної бази даних isp_protection було згенеровано тестові записи у таблиці users. Ці записи мають умовні значення, включаючи зашифровані імена та хешовані паролі, що імітують реальне середовище функціонування системи. Така симуляція дозволяє протестувати функціонал автентифікації та перевірити ефективність збереження персональних даних, заповнено на рисунку 3.4.

| ←T→ | id | full_name | email | password_hash | ip_address | created_at |
|---|---|---|---|---|---|---|
| Редагувати Копіювати Видалити | 1 | R29yb2g= | ivan@ukr.net | 9a04c3e9e85c6a9e9e1d4b02afc6eb29b79e... | 192.168.1.2 | 2025-05-24 23:03:58 |
| Редагувати Копіювати Видалити | 2 | T2xlbmE= | olena@gmail.com | 07fd23ac99e3641b8f23f76c9dcf2b34fbe2... | 192.168.1.3 | 2025-05-24 23:03:58 |
| Редагувати Копіювати Видалити | 3 | U3RlcGFu | stepan@mail.ua | 1af14cbe01572832e7b12f7cc9cf9f33f67e... | 192.168.1.4 | 2025-05-24 23:03:58 |
| Редагувати Копіювати Видалити | 4 | SXJ5bmE= | iryna@meta.ua | 3a94fd004e28fcd4a3a4ee7ec5b5288446b2... | 192.168.1.5 | 2025-05-24 23:03:58 |
| Редагувати Копіювати Видалити | 5 | TWFrc3lt | maksym@gmail.com | 88b203e2e281a3c70c8fa5a17b964ea7725f... | 192.168.1.6 | 2025-05-24 23:03:58 |
| Редагувати Копіювати Видалити | 6 | VmlrdG9y | viktor@ukr.net | 62265df21aefef3ef88a2bb46cc7f84532a0... | 192.168.1.7 | 2025-05-24 23:03:58 |
| Редагувати Копіювати Видалити | 7 | T2tzYW5h | oksana@mail.com | 4edc9f9aa987c78db925d0b4512cd07fcb3a... | 192.168.1.8 | 2025-05-24 23:03:58 |
| Редагувати Копіювати Видалити | 8 | TmF0YWxrYQ== | natalka@meta.ua | 7f034aa8c28932f76b64b80db270a06df2a4... | 192.168.1.9 | 2025-05-24 23:03:58 |
| Редагувати Копіювати Видалити | 9 | WXVyaWk= | yurii@ukr.net | d1ce7a4f321e2dfe318ef4df9e10984a4933... | 192.168.1.10 | 2025-05-24 23:03:58 |
| Редагувати Копіювати Видалити | 10 | TWFyaWlh | mariia@gmail.com | bfb44dcac1eec22d66f7b704bc9c012f5ad7... | 192.168.1.11 | 2025-05-24 23:03:58 |

Рисунок 3.4 – Демонстраційне заповнення таблиці users

У таблиці users демонстраційно внесено 10 умовних записів, які імітують реальних користувачів. Імена збережено у зашифрованому вигляді (приклад – кодування base64, як спрощена ілюстрація шифрування AES). Паролі хешовані за допомогою SHA-256, що гарантує їхню незворотність та безпечне зберігання.

Кожен запис містить унікальну IP-адресу та дату створення, яка фіксується автоматично через механізм CURRENT_TIMESTAMP. Така модель дозволяє в подальшому ефективно реалізувати автентифікацію, аудит доступу та контроль підозрілих дій користувачів.

Крім основної таблиці users, для забезпечення фіксації подій, контролю доступу та виявлення загроз у межах тієї ж бази даних було реалізовано дві додаткові таблиці: logs та alerts.

Структура таблиці logs відображена в таблиці 3.1.

Таблиця 3.1 – Структура таблиці logs

| Поле | Тип даних | Призначення |
|---|---|---|
| log_id | INT, PK | Унікальний ідентифікатор запису |
| user_id | INT (FK) | Ідентифікатор користувача з таблиці users |
| ip_address | VARCHAR(45) | IP-адреса користувача під час дії |
| action | VARCHAR(50) | Тип дії (login, logout, error, access) |
| created_at | TIMESTAMP | Час фіксації події |

Таблиця logs. Ця таблиця використовується для збереження історії дій користувача, зокрема – входів до системи, вказуючи IP-адресу, тип дії та час події.

У таблиці logs продемонстровано приклади фіксації типових дій користувачів у системі: вхід (login), вихід (logout) та помилка (error), зазначено на структурі рисунку 3.5.

Структура таблиці | Вид відносин

| # | Ім'я | Тип | Зіставлення | Атрибути | Нуль | За замовчуванням | Коментарі | Додатково | Дія |
|---|---|---|---|---|---|---|---|---|---|
| 1 | log_id | int(11) | | | Ні | *Немає* | | AUTO_INCREMENT | Змінити Знищити Більше |
| 2 | user_id | int(11) | | | Так | *NULL* | | | Змінити Знищити Більше |
| 3 | ip_address | varchar(45) | utf8mb4_general_ci | | Так | *NULL* | | | Змінити Знищити Більше |
| 4 | action | varchar(50) | utf8mb4_general_ci | | Так | *NULL* | | | Змінити Знищити Більше |
| 5 | created_at | timestamp | | | Ні | current_timestamp() | | | Змінити Знищити Більше |

Рисунок 3.5 – Структура таблиці logs для журналювання дій користувачів

Запис здійснюється з урахуванням IP-адреси, часу події та ідентифікатора користувача (через зовнішній ключ на таблицю users). Такий підхід дозволяє реалізувати ефективний аудит доступу та контроль активності в системі, продемонстровано на рисунку 3.6 нижче.

| | log_id | user_id | ip_address | action | created_at |
|---|---|---|---|---|---|
| Редагувати Копіювати Видалити | 1 | 1 | 192.168.1.2 | login | 2025-05-24 23:11:26 |
| Редагувати Копіювати Видалити | 2 | 3 | 192.168.1.4 | logout | 2025-05-24 23:11:26 |
| Редагувати Копіювати Видалити | 3 | 5 | 192.168.1.6 | login | 2025-05-24 23:11:26 |
| Редагувати Копіювати Видалити | 4 | 2 | 192.168.1.3 | login | 2025-05-24 23:11:26 |
| Редагувати Копіювати Видалити | 5 | 4 | 192.168.1.5 | error | 2025-05-24 23:11:26 |

Рисунок 3.6 – Демонстраційне заповнення таблиці logs у середовищі phpMyAdmin

Потенційно шкідлива активність – це дії, які можуть призвести до порушення безпеки інформаційної системи. Вона включає несанкціонований доступ, виконання шкідливого коду, спроби обійти захист або отримати конфіденційні дані. Такі дії можуть бути частиною атаки або свідчити про наявність вразливостей у системі. До неї автоматично записуються дані у разі виявлення загроз.

Структура (таблиця 3.2) використовується для фіксації підозрілої або потенційно шкідливої активності. Таблиця alerts призначена для зберігання записів про загрози безпеці, які виявляються під час взаємодії користувачів із системою.

Таблиця 3.2 – Структура таблиці alerts

| Поле | Тип даних | Призначення |
|---|---|---|
| alert_id | INT, PK | Унікальний ідентифікатор події |
| type | VARCHAR(50) | Тип загрози (SQLi, XSS, тощо) |
| details | TEXT | Опис загрози або фрагмент підозрілого запиту |
| ip_address | VARCHAR(45) | IP-адреса джерела загрози |
| created_at | TIMESTAMP | Дата і час виявлення події |

Вона містить такі поля, як type (тип загрози), details (опис або фрагмент шкідливого запиту), ip_address (джерело) та created_at (час події). Важливість цієї таблиці полягає у тому, що вона дозволяє не лише зафіксувати потенційні атаки, але й надалі проводити аналіз інцидентів, виявляти закономірності та оновлювати політику фільтрації запитів, показано на риснунку 3.7.

| # | Ім'я | Тип | Зіставлення | Атрибути | Нуль | За замовчуванням | Коментарі | Додатково | Дія |
|---|---|---|---|---|---|---|---|---|---|
| 1 | alert_id | int(11) | | | Ні | *Немає* | | AUTO_INCREMENT | Змінити Знищити Більше |
| 2 | type | varchar(50) | utf8mb4_general_ci | | Ні | *Немає* | | | Змінити Знищити Більше |
| 3 | details | text | utf8mb4_general_ci | | Так | *NULL* | | | Змінити Знищити Більше |
| 4 | ip_address | varchar(45) | utf8mb4_general_ci | | Так | *NULL* | | | Змінити Знищити Більше |
| 5 | created_at | timestamp | | | Ні | current_timestamp() | | | Змінити Знищити Більше |

Рисунок 3.7 – Структура таблиці alerts для фіксації загроз та підозрілих дій

До кожної події зберігаються тип загрози (type), її опис (details), IP-адреса джерела та час виникнення. Це дозволяє оперативно реагувати на потенційно небезпечні ситуації та забезпечити журналювання інцидентів безпеки для подальшого аналізу й удосконалення системи захисту.

Далі наведемо реалізацію таблиці alerts. У таблиці alerts зафіксовано п’ять типових загроз, які можуть бути виявлені в процесі взаємодії користувача з системою, продемонстровано на рисунку 3.8.

| | alert_id | type | details | ip_address | created_at |
|---|---|---|---|---|---|
| Редагувати Копіювати Видалити | 1 | SQL Injection | Виявлено символи: "OR 1=1--" | 192.168.1.12 | 2025-05-24 23:18:17 |
| Редагувати Копіювати Видалити | 2 | XSS Attack | Виявлено фрагмент: <script>alert(1)</script> | 192.168.1.13 | 2025-05-24 23:18:17 |
| Редагувати Копіювати Видалити | 3 | Brute Force | Кілька спроб входу з невірним паролем | 192.168.1.14 | 2025-05-24 23:18:17 |
| Редагувати Копіювати Видалити | 4 | Invalid Token | Недійсний токен автентифікації | 192.168.1.15 | 2025-05-24 23:18:17 |
| Редагувати Копіювати Видалити | 5 | Unknown Method | Запит із забороненим HTTP-методом: TRACE | 192.168.1.16 | 2025-05-24 23:18:17 |

Рисунок 3.8 – Демонстраційне заповнення таблиці alerts у середовищі phpMyAdmin

Таким чином, обидві таблиці (logs та alerts) суттєво підвищують можливості системи у виявленні несанкціонованої активності та відстеженні дій користувачів, формуючи основу для журналювання та виявлення загроз.

## 3.3 Реалізація елементів захисту від шкідливого ПЗ (на прикладі фільтрації/перевірки запитів)

У рамках побудови системи захисту критично важливо забезпечити контроль авторизації користувачів та обмежити можливість повторного доступу сторонніх осіб. Для цього в базі даних реалізовано таблицю access_tokens, яка зберігає токени доступу – унікальні ідентифікатори, що видаються після успішної автентифікації.

Токени є аналогом сесій у веб-системах і можуть застосовуватись для:

- збереження стану входу користувача;
- обмеження доступу до захищених ресурсів;
- автоматичного завершення сесій після закінчення терміну дії;
- логування та аналізу дій користувачів.

У таблиці access_tokens (таблиця 3.3) зберігаються унікальні токени доступу, які прив’язані до користувачів за допомогою зовнішнього ключа. Поле expires_at визначає термін дії токена, після чого доступ до системи буде анульовано або вимагатиме повторної автентифікації.

Таблиця 3.3 – Структура таблиці access_tokens

| Поле | Тип даних | Призначення |
|---|---|---|
| token_id | INT, PK | Унікальний ідентифікатор токена |
| user_id | INT (FK) | Ідентифікатор користувача |
| token | VARCHAR(255) | Сам токен (унікальний рядок, наприклад, JWT або UUID) |
| expires_at | DATETIME | Час, коли токен втрачає чинність |
| created_at | TIMESTAMP | Дата створення токена |

Такий підхід підвищує рівень безпеки, дозволяє керувати активними сесіями та знижує ризик компрометації облікових записів у разі втрати токена.


Рисунок 3.9 – Структура таблиці access_tokens у phpMyAdmin

У системах з багаторівневим доступом важливо мати чітке розмежування прав користувачів. Наприклад, адміністратор має більше можливостей, ніж звичайний користувач. Для цього у базі даних створюється окрема таблиця-довідник roles, де зберігаються усі доступні ролі системи.

Кожен користувач може мати одну або декілька ролей, які визначають його можливості в системі: перегляд, редагування, адміністрування, управління безпекою тощо.

Система дозволяє гнучко керувати правами доступу, забезпечуючи принцип найменших привілеїв для підвищення рівня безпеки. У таблиці roles (таблиця 3.4) наведено перелік доступних ролей у системі, кожна з яких має власний рівень доступу та функціональні можливості.

Таблиця 3.4 – Структура таблиці roles

| Поле | Тип даних | Призначення |
|---|---|---|
| role_id | INT, PK | Унікальний ідентифікатор ролі |
| name | VARCHAR(50) | Назва ролі (наприклад: admin, user, audit) |
| description | TEXT | Опис функцій або прав, що надає ця роль |

Такий підхід дозволяє централізовано керувати правами користувачів, спрощує контроль доступу до критичних компонентів та забезпечує дотримання принципу найменших привілеїв. Для повноцінного управління правами доступу система повинна підтримувати призначення ролей користувачам, як показано на рисунку 3.10.

| | role_id | name | description |
|---|---|---|---|
| Копіювати Видалити | 1 | admin | Повний доступ до системи, керування користувачами ... |
| Копіювати Видалити | 2 | user | Звичайний зареєстрований користувач із базовими пр... |
| Копіювати Видалити | 3 | moderator | Може переглядати та блокувати шкідливі запити |
| Копіювати Видалити | 4 | auditor | Перегляд журналів, без можливості змін |

Рисунок 3.10 – Демонстраційне заповнення таблиці roles у phpMyAdmin

Це дозволяє задавати індивідуальні права або групувати користувачів за функціоналом. Така гнучка система є стандартною практикою у побудові безпечних інформаційних систем.

У таблиці user_roles реалізовано зв’язок між користувачами та їхніми ролями. Це дає змогу динамічно контролювати права доступу, розширювати повноваження окремих користувачів або обмежувати їх.

Таблиця user_roles (таблиця 3.5) реалізує зв’язок між таблицею users та roles, забезпечуючи можливість призначення однієї або кількох ролей кожному користувачеві.

Таблиця 3.5 – Структура таблиці user_roles

| Поле | Тип даних | Призначення |
|---|---|---|
| id | INT, PK | Унікальний ідентифікатор запису |
| user_id | INT (FK) | Зовнішній ключ на таблицю users |
| role_id | INT (FK) | Зовнішній ключ на таблицю roles |
| assigned_at | TIMESTAMP | Дата і час призначення ролі |

Такий підхід забезпечує масштабованість і безпечну роботу системи в умовах багатокористувацького середовища , заповнено на рисунку 3.11.

| | id | user_id | role_id | assigned_at |
|---|---|---|---|---|
| Копіювати Видалити | 1 | 1 | 1 | 2025-05-24 23:39:14 |
| Копіювати Видалити | 2 | 2 | 2 | 2025-05-24 23:39:14 |
| Копіювати Видалити | 3 | 3 | 2 | 2025-05-24 23:39:14 |
| Копіювати Видалити | 4 | 4 | 3 | 2025-05-24 23:39:14 |
| Копіювати Видалити | 5 | 5 | 4 | 2025-05-24 23:39:14 |

Рисунок 3.11 – Демонстраційне заповнення таблиці user_roles у phpMyAdmin

Для підтримки гнучкості та адаптивності системи захисту важливо винести критичні параметри безпеки в окрему таблицю. Це дозволяє адміністратору змінювати політику безпеки без потреби редагувати код. До таких параметрів можуть належати: максимальна кількість спроб входу, тайм-аут сесії, рівень захисту (низький, середній, високий), дозволені методи запитів тощо.

У таблиці security_settings зберігаються ключові параметри конфігурації системи захисту, які можна змінювати без втручання у програмний код. Структура таблиці security_settings відображена в таблиці 3.6.

Таблиця 3.6 – Структура таблиці security_settings

| Поле | Тип даних | Призначення |
|---|---|---|
| setting_id | INT, PK | Унікальний ідентифікатор налаштування |
| name | VARCHAR(100) | Назва параметра (ключ) |
| value | VARCHAR(255) | Значення параметра |
| description | TEXT | Пояснення або призначення параметра |
| updated_at | TIMESTAMP | Дата та час останнього оновлення |

Це дозволяє швидко адаптувати політику безпеки до нових умов, змінювати правила обробки запитів, таймаути сесій чи рівень загального захисту системи. Налаштування системи безпеки зазначено на рисунку 3.12.

| | setting_id | name | value | description | updated_at |
|---|---|---|---|---|---|
| Видалити | 1 | max_login_attempts | 5 | Максимальна кількість спроб входу перед блокування... | 2025-05-24 23:44:22 |
| Видалити | 2 | session_timeout_minutes | 30 | Час неактивності перед завершенням сесії | 2025-05-24 23:44:22 |
| Видалити | 3 | password_min_length | 8 | Мінімальна довжина пароля | 2025-05-24 23:44:22 |
| Видалити | 4 | security_level | high | Рівень захисту системи (low, medium, high) | 2025-05-24 23:44:22 |
| Видалити | 5 | allowed_methods | GET,POST | Дозволені HTTP-методи у запитах до сервера | 2025-05-24 23:44:22 |

Рисунок 3.12 – Налаштування безпеки системи у таблиці security_settings

Отже, у межах даного підрозділу було реалізовано ключові елементи захисту інформаційної системи інтернет-провайдера від шкідливого програмного забезпечення та загроз з боку користувачів. Основну увагу було зосереджено на побудові бази даних, що забезпечує контроль доступу, аудит дій, виявлення атак та гнучке управління політикою безпеки.

Було впроваджено таблицю access_tokens для зберігання сесій користувачів, що дозволяє відслідковувати активні токени доступу та обмежувати час їхньої дії. Механізм рольового доступу реалізовано за допомогою таблиць roles та user_roles, які забезпечують розмежування прав користувачів згідно з принципом найменших привілеїв.

Для обробки інцидентів безпеки реалізовано таблицю alerts, яка фіксує спроби SQL-ін’єкцій, XSS-атак та інші підозрілі запити, а також таблицю logs, що зберігає хронологію дій користувачів у системі (входи, виходи, помилки тощо).

Крім того, впроваджено таблицю security_settings, яка дозволяє централізовано зберігати та оновлювати критичні параметри безпеки – тайм-аути, кількість спроб входу, дозволені HTTP-методи та рівень захисту. Це підвищує гнучкість адміністрування безпеки та адаптивність системи до нових загроз.

Таким чином, побудована модель забезпечує конфіденційність, цілісність і доступність інформації відповідно до вимог сучасної кібербезпеки, а також створює основу для подальшої інтеграції засобів автоматичного виявлення та реагування на загрози.

## 3.4 Тестування системи та оцінка ефективності

Метою тестування є перевірка працездатності, надійності та ефективності реалізованої системи захисту інформаційних ресурсів інтернет-провайдера. Особлива увага приділяється відповідності системи ключовим вимогам безпеки: автентифікації, шифруванню персональних даних, контролю дій користувачів, фіксації загроз та управлінню доступом.

Тестування (таблиця 3.7) проводиться у симульованому середовищі з використанням phpMyAdmin, локального сервера XAMPP та засобів PHP. В рамках перевірки було змодельовано низку типових сценаріїв взаємодії користувачів і зловмисників із системою. Ці сценарії тестування є типовими перевірками для всієї системи інтернет-провайдера. Дотримуючись таких перевірок система буде завжди у безпеці , тому що вже проведенне тестування на різні засоби захисту.

Таблиця 3.7 – Тестові сценарії для перевірки системи захисту

| № | Сценарій тестування | Очікувана поведінка системи |
|---|---|---|
| 1 | Реєстрація користувача | Пароль зберігається у вигляді хешу (SHA-256); full_name – зашифрований |
| 2 | Вхід користувача | Створюється запис у таблиці logs та access_tokens |
| 3 | SQL-ін’єкція у полі логіна | Запис у таблиці alerts, дія блокується |
| 4 | XSS-атака у формі | Запис у alerts, система не виконує скрипт |
| 5 | Занадто багато спроб входу | Запит блокується; користувача можна додати до blocked_ips (за умовою) |
| 6 | Перевірка ролей користувача | Доступ дозволяється / забороняється згідно з роллю у user_roles |
| 7 | Перевірка обмеження токена за часом | Токен стає недійсним після expires_at |
| 8 | Перевірка записів у logs | Відображення історії входів, IP-адрес, дій |
| 9 | Оновлення параметрів у security_settings | Відразу впливають на правила фільтрації чи тайм-аути |

На рисунку 3.13 показано, як система реагує на спробу SQL-ін’єкції, зберігаючи відповідний запис у таблиці alerts. Фіксується тип загрози, її опис (наприклад, підозрілий фрагмент у полі логіна) та IP-адреса, з якої надійшов запит.

| lert_id | type | details | ip_address | created_at |
|---|---|---|---|---|
| 1 | SQL Injection | Виявлено символи: "OR 1=1--" | 192.168.1.12 | 2025-05-24 23:18:17 |
| 2 | XSS Attack | Виявлено фрагмент: <script>alert(1)</script> | 192.168.1.13 | 2025-05-24 23:18:17 |
| 3 | Brute Force | Кілька спроб входу з невірним паролем | 192.168.1.14 | 2025-05-24 23:18:17 |
| 4 | Invalid Token | Недійсний токен автентифікації | 192.168.1.15 | 2025-05-24 23:18:17 |
| 5 | Unknown Method | Запит із забороненим HTTP-методом: TRACE | 192.168.1.16 | 2025-05-24 23:18:17 |
| 6 | SQL Injection | Спроба SQL-ін’єкції: OR 1=1 -- у полі вводу логіна | 192.168.1.77 | 2025-05-24 23:57:12 |

Рисунок 3.13 – Приклад запису про SQL-ін’єкцію у таблиці alerts

Це дозволяє надалі аналізувати загрози, виявляти джерела атак і реагувати на них автоматично (наприклад, блокуванням IP-адреси або надсиланням сповіщення адміністратору).

У результаті експериментального тестування розробленої системи захисту інформаційних ресурсів інтернет-провайдера було підтверджено її відповідність заявленим функціональним та безпековим вимогам. Тестування охоплювало декілька напрямів: перевірка збереження персональних даних, обробка аномальної активності, ведення журналів дій, забезпечення авторизації з урахуванням ролей та керування активними сесіями.

Аналіз результатів показав, що паролі користувачів системи не зберігаються у відкритому вигляді, що відповідає принципам конфіденційності та вимогам безпечного зберігання облікових даних. Зашифроване збереження персональних даних (зокрема імені користувача) забезпечує додатковий рівень захисту у разі несанкціонованого доступу до бази даних.

Компоненти журналювання (logs) та оповіщення (alerts) виконують функції моніторингу та фіксації дій, що дозволяє виявляти спроби несанкціонованих дій, наприклад, SQL-ін’єкцій, або аномальну поведінку користувачів. Збереження детальної інформації про IP-адресу, час дії та її тип забезпечує передумови для аудиту безпеки.

Система авторизації на основі ролей продемонструвала можливість гнучкого розмежування прав доступу. Тестування підтвердило, що користувачі, залежно від призначеної їм ролі, мають або обмежений, або розширений доступ до функціоналу. Окремо було протестовано механізм контролю активності сесій за допомогою токенів (access_tokens), що дозволяє реалізувати обмеження часу сесії, автоматичне завершення дії токена, а також анулювання у випадку порушень.

Усі функціональні компоненти системи продемонстрували належний рівень працездатності, що підтверджено результатами тестових сценаріїв, наведених у таблиці 3.8.

Таблиця 3.8 – Результати тестування функціональних компонентів системи

| № | Функціональний елемент | Очікуваний результат | Результат |
|---|---|---|---|
| 1 | Хешування паролів | Пароль зберігається у вигляді SHA-256 хешу у таблиці users | Виконано |
| 2 | Шифрування персональних даних | Поле full_name зашифроване за допомогою AES-256 перед збереженням у БД | Виконано |
| 3 | Вхід користувача | Створюється запис у таблиці logs, генерується токен у access_tokens | Виконано |
| 4 | Реєстрація шкідливого запиту | Спроба SQL-ін’єкції фіксується у таблиці alerts | Виконано |
| 5 | Рольова авторизація | Користувач отримує доступ згідно з роллю, призначеною у user_roles | Виконано |
| 6 | Закінчення дії токена | Після expires_at токен втрачає чинність | Виконано |
| 7 | Оновлення параметрів безпеки | Зміна параметра в security_settings впливає на поведінку системи | Виконано |

Централізоване зберігання параметрів безпеки у таблиці security_settings забезпечує налаштовуваність системи без необхідності модифікації програмного коду. Зміна, наприклад, максимальної кількості спроб входу або дозволених HTTP-методів, миттєво впливає на поведінку системи.

У результаті тестування реалізованої системи захисту інформаційних ресурсів інтернет-провайдера підтверджено її функціональну цілісність та відповідність поставленим вимогам щодо безпеки. Усі ключові компоненти системи – включно з механізмами автентифікації, авторизації, шифрування, журналювання та реагування на загрози – продемонстрували стабільну та коректну роботу.

Хешування паролів за допомогою криптографічного алгоритму SHA-256 та шифрування персональних даних засобами AES-256 забезпечують належний рівень конфіденційності. Рольова модель доступу, побудована на зв’язку між таблицями users, roles та user_roles, дозволяє реалізувати гнучке розмежування прав доступу до системи.

Фіксація дій користувачів у таблиці logs та запис інцидентів безпеки у alerts створюють надійний механізм аудиту, що є важливою складовою виявлення підозрілої активності та ведення статистики безпечності. Застосування токенів (access_tokens) дозволяє керувати сесіями та забезпечити своєчасне завершення неактивних з’єднань.

Окрема увага була приділена збереженню налаштувань безпеки у таблиці security_settings, що надає змогу адміністратору оперативно змінювати політику захисту системи без необхідності модифікувати її логіку.

Таким чином, система підтвердила свою ефективність як у плані реалізації сучасних принципів інформаційної безпеки (конфіденційність, цілісність, автентичність), так і в контексті практичної придатності до впровадження у середовищі провайдерських інформаційних ресурсів.

## 3.5Висновки до третього розділу

У межах третього розділу кваліфікаційної роботи здійснено практичну реалізацію та тестування спроєктованої системи захисту інформаційних ресурсів інтернет-провайдера, спрямованої на виявлення та нейтралізацію загроз, пов’язаних із шкідливим програмним забезпеченням та іншими типами несанкціонованої активності. Було обрано відповідне середовище розробки – XAMPP, як інструмент локальної розгортки серверного середовища на основі Apache, PHP і MySQL. Це забезпечило гнучкість у налаштуванні, простоту тестування та сумісність з інфраструктурою типового провайдера.

У підрозділі 3.2 створено базу даних користувачів, до якої реалізовано механізми збереження інформації з урахуванням вимог інформаційної безпеки. Застосовано хешування паролів за допомогою SHA-256 та шифрування персональних ідентифікаторів з використанням алгоритму AES-256.

У межах підрозділу 3.3 реалізовано розширення функціоналу захисту шляхом створення додаткових таблиць:

- logs – для фіксації спроб входу;
- alerts – для виявлення і реєстрації інцидентів, зокрема SQL-ін’єкцій;
- access_tokens – для контролю дій автентифікованих сесій;
- roles та user_roles – для визначення рівня доступу;
- blocked_ips та security_settings – для реагування на аномалії та зберігання параметрів безпеки.

Ці компоненти забезпечують багаторівневий захист: фільтрацію запитів, логування, централізоване зберігання параметрів безпеки, керування ролями та контроль поведінки користувачів. Реалізовано взаємозв’язки між таблицями, що дозволяє системі адаптуватися до нових викликів та масштабуватися.

У підрозділі 3.4 проведено тестування системи на основі сценаріїв, що включали спроби SQL-ін’єкцій, входу з помилковими обліковими даними, перевірку надійності шифрування та ведення журналів. Система успішно виявляє загрози, блокує небезпечні запити, генерує відповідні повідомлення та зберігає інформацію для подальшого аналізу. Таблиці alerts і logs підтверджують фіксацію інцидентів, що є важливим чинником для забезпечення подальшої обробки, звітності та удосконалення засобів реагування.

Таким чином, реалізована система захисту продемонструвала функціональну повноту, надійність, здатність до адаптації та відповідність сучасним стандартам інформаційної безпеки в умовах інтернет-провайдерського середовища. Це дозволяє вважати її ефективним програмно-архітектурним рішенням, яке може бути основою для подальшого впровадження або масштабування в реальному середовищі.

# ВИСНОВКИ

Інформаційна безпека в умовах стрімкого розвитку цифрових технологій та зростаючих загроз у кіберпросторі є однією з пріоритетних задач для інтернет-провайдерів. Надійна система захисту інформаційних ресурсів має гарантувати безперервну роботу сервісів, захист персональних даних користувачів, а також здатність своєчасно виявляти та нейтралізовувати потенційні атаки.

В межах кваліфікаційної роботи на тему «Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення» було комплексно вирішено завдання проєктування, реалізації та тестування інформаційної системи, що здатна виявляти та протидіяти базовим видам кіберзагроз.

На першому етапі проведено ґрунтовний теоретичний аналіз стану проблеми, визначено основні види шкідливого програмного забезпечення (віруси, трояни, скриптові ін’єкції, програми-шпигуни тощо) та проаналізовано сучасні підходи до побудови систем захисту на рівні прикладного програмного забезпечення, баз даних і мережевої інфраструктури. Особливу увагу приділено аналізу відкритих вразливостей, таких як SQL-ін’єкції, XSS-атаки, обхід автентифікації та підміна запитів.

На другому етапі проєкту було спроєктовано структуру бази даних для обліку користувачів та супутніх дій, яка містить такі компоненти, як:

- основна таблиця users для зберігання облікових записів;
- logs для фіксації усіх дій користувачів (входів, помилок, спроб доступу);
- alerts для зберігання зафіксованих інцидентів безпеки;
- access_tokens для управління авторизаційними сесіями;
- roles і user_roles для реалізації рольової моделі доступу;
- security_settings як механізм конфігурації параметрів безпеки без втручання в код.

Реалізація бази даних здійснювалася за допомогою середовища phpMyAdmin, а логіка обробки даних – за допомогою мови PHP. Застосовано алгоритм хешування SHA-256 для паролів користувачів, що гарантує незворотність та захист від компрометації облікових даних. Для захисту персональної інформації реалізовано шифрування поля full_name з використанням симетричного алгоритму AES-256, що забезпечує конфіденційність у випадку несанкціонованого доступу до бази даних.

Окремо в роботі розглянуто реалізацію фільтрації вхідних даних користувача. Запити, які містять шкідливі патерни (наприклад, SQL-ін’єкцію або XSS), розпізнаються засобами регулярних виразів і не обробляються, а їхній вміст автоматично записується у таблицю alerts. Це дозволяє фіксувати всі підозрілі дії та потенційні загрози для подальшого аналізу.

Також реалізовано рівневе управління доступом: користувачам можуть бути призначені різні ролі, що обмежують або розширюють їхні можливості в системі. Такий підхід відповідає принципу мінімальних привілеїв та забезпечує сегментацію доступу до критично важливих компонентів.

Система успішно протестована за допомогою низки імітаційних сценаріїв, у тому числі – моделювання входу користувача, шкідливих запитів, перевищення дозволених спроб входу та завершення сесій. Під час тестування підтверджено:

- правильність обробки персональних даних;
- коректність дій системи при виявленні атак;
- стабільність при багаторазовому доступі;
- працездатність функціоналу автоматичного логування та шифрування;
- гнучкість системи налаштувань безпеки.

Результати показали, що реалізована система відповідає сучасним вимогам до інформаційної безпеки для веб-застосунків та може слугувати прототипом для впровадження у провайдерських середовищах, корпоративних мережах та малих IT-інфраструктурах.

У процесі виконання роботи досягнуто такі основні результати:

1. Досліджено характер сучасних кіберзагроз, актуальних для інтернет-провайдерів.

2. Побудовано базу даних користувачів із урахуванням вимог до безпечного зберігання та обробки інформації.

3. Реалізовано інструменти хешування, шифрування та контролю активності користувачів.

4. Запропоновано модель фільтрації запитів з фіксацією атак та системою автоматичного реагування.

5. Проведено тестування, яке підтвердило ефективність та надійність побудованої системи.

Таким чином, у роботі реалізовано повнофункціональну систему захисту, що включає не лише базові засоби контролю, але й розширені функції фіксації інцидентів, гнучкого управління доступом та масштабованості.

# ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ


1. Антоненко Н., Дігтяр Я., Крикун Н. Сучасні методи боротьби з комп’ютерними вірусами. Економіка та суспільство, 2022. №43.

2. Балакін С. В. Застосування штучних імунних систем при виявленні шкідливих програм в комп’ютерній мережі. Проблеми інформатизації та управління, 2017. №1-2(57-58). С. 7-11.

3. Бараннік В. В. Архітектурні особливості систем захисту веб-ресурсів від несанкціонованого доступу / В.В. Бараннік, О.В. Слободянюк, Н.В. Бараннік // НІСТ'2019. Наукоємні технології в інфокомунікація : матеріали ІІІ Міжнар. наук.-практ. конф. Харків. 2019. С.45-46.

4. Безверхня Ю. В. Хеджування як елемент управлінського обліку. Сучасні проблеми обліку, аналізу, аудиту й оподаткування суб’єктів господарської діяльності: теоретичні, практичні та освітянські аспекти : зб. наук. праць за матеріалами IV Всеукр. наук.-практ. конф. 30-31 березня 2020 р. С. 17-20.

5. Босак І. М., Данилович-Кропивницька М. Л. Аналіз світового досвіду формування системи інформаційної безпеки в контексті публічного управління. Науковий вісник Міжнародного гуманітарного університету. 2024. № 58. С. 72–78.

6. Вавіленкова А. Загрози від використання cloud-сервісів у сфері кібербезпеки // Інформ. безпека людини, сусп-ва, д-ви. 2023. № 2. URL: https://csecurity.kubg.edu.ua/index.php/journal/article/view/704 (дата звернення: 17.05.2025).

7. Гавриш Б. М., Тимченко О. В., Борзов Ю. О. Класифікація шкідливого програмного забезпечення та основні методи захисту. Комп’ютерні технології друкарства, 2022. №2 (48). С. 142-154.

8. Жилін А. В. Технології захисту інформації в інформаційно-телекомунікаційних системах : навч. посіб. / А. В. Жилін, О. М. Шаповал, О. А.

Успенський ; ІСЗЗІ КПІ ім. Ігоря Сікорського. Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2021. 213 с.

9. Зубок М., Мохор В. Кібербезпека топології Internet [Електронний ресурс].URL:https://ipme.kiev.ua/wpcontent/uploads/2022/07/Zubok_Mokhor_Kiberbezpeka_Topologii_Internet.pdf (дата звернення: 18.05.2025).

10. Інформаційна безпека та інформаційні технології: збірник тез доповідей VI Всеукраїнської науково-практичної конференції молодих учених, студентів і курсантів, м. Львів, 30 листопада 2023 року. Львів, ЛДУ БЖД, 2023, 489 с.

11. Інформаційна безпека держави. Конспект лекцій для здобувачів вищої освіти освітнього ступеню «бакалавр» спеціальності 262 «Правоохоронна діяльність»// Укл.: Ю.М.Ткач, С.М.Семендяй - Чернігів: НУ «Чернігівська політехніка», 2022. –133 с.

12. Карпович І. М., Гладка О .М., Наконечна Ю. А. Аналіз ризиків безпеки інформаційної системи іт-підприємства. Таврійський національний університет Імені в.І. Вернадського. Технічні науки, 2020. Том 31 (70) № 5. С. 69-74.

13. Керусов М. В. Політика інформаційної безпеки інтернет-провайдерів [Електронний ресурс] // Наук. вісн. Міжнар. гуманіт. ун-ту. 2021. № 50. URL: https://dspace.onua.edu.ua/bitstreams/f874b548-3390-4254-881f-b48dc79e7544/download (дата звернення: 17.05.2025).

14. Кібербезпека в інформаційному суспільстві : інформ.-аналіт. дайджест № 5 (трав. 2024) / Ін-т інформ., безпеки і права НАПрН України [Електронний ресурс]. URL: https://ippi.org.ua/sites/default/files/2024-5.pdf (дата звернення: 18.05.2025).

15. Козюра В. Д., Хорошко В. О., Шелест М. Є., Ткач Ю. М., Балюнов О.О. Захист інформації в комп'ютерних системах: підручник. Ніжин: ФОП Лук'яненко В.В., ТПК «Орхідея», 2020. 236с.

16. Костюк К. О. Дослідження криптографічних протоколів захисту інформації в мережі Інтернет. Тернопіль: ТНТУ, 2023. 63 с.

17. Мартинов С. М. Інформаційна безпека : підручник. Харків : ХНУРЕ, 2020. 280 с.

18. Мельник І. Розробка макросу та дослідження шифру Віженера та його модифікації. Вінницький національний аграрний університет. Журнал студентських наукових праць, 2022. №5. С. 256-253.

19. Методологічні аспекти забезпечення інформаційної безпеки підприємства / П. А. Фісуненко, О. І. Судакова, С. В. Деркач, Є. О. Притуляк // Східна Європа: економіка, бізнес та управління, 2019. № 23. С. 426-431.

20. Методологія захисту інформації. Аспекти кібербезпеки: підручник/ Г.М. Гулак – Київ: Видавництво НА СБ України, 2020. 256 с.

21. Моделі і методи захисту від загрозливих програм інформаційних систем / В. М. Джулій, В. О. Бойчук, В. Ю. Тітова, О. В. Сєлюков, О. В. Мірошніченко // Зб. наук. праць Військового інституту Київського національного університету імені Тараса Шевченка. Київ : ВІКНУ, 2020. Вип. 67. С. 72–84.

22. Нестеренко О., Поліщук В., Хижняк В., Шевченко В. Інформаційні технології підтримки прийняття рішень щодо визначення ресурсів для гасіння лісової пожежі засобами авіації. Екологічна безпека та природокористування, 46(2), 2023. С. 109–123.

23. Про захист інформації в інформаційно-телекомунікаційних системах : Закон України від 05.07.1994 № 80/94-ВР. URL: https://zakon.rada.gov.ua/laws/show/80/94-вр (дата звернення: 19.05.2025).

24. Програмне забезпечення для запобігання атакам з боку шкідливого ПЗ. Symantec Threat Report. URL: https://www.broadcom.com/company/newsroom (дата звернення: 21.05.2025).

25. Слободянюк В. В. Основи захисту інформації в комп’ютерних системах : навч. посіб. Київ : ВД "Професіонал", 2018. 312 с.

26. Топчій В. В., Бодунова О. М. Проблемні питання забезпечення кібербезпеки в Україні // Актуал. пробл. правознавства. 2024. № 1(37). С. 112–118.

URL: https://app-journal.in.ua/wp-content/uploads/2025/02/112.pdf (дата звернення: 25.05.2025)

27. Трофименко О., Дубовой Я., Логінова Н., Прокоп Ю., Задерейко О. Аналіз проблем забезпечення кібербезпеки медичних комп'ютерних систем. Захист інформації. Т. 23. 2021. № 1. Січень-березень. С. 30–39.

28. Тугарова О.К., Пономаренко І.С. Проблемні питання організації захисту інформації в медичних інформаційних системах. Актуальні проблеми управління інформаційною безпекою держави: матеріали XV Всеукраїнської науково-практичної конференції (м. Київ, 27 березня 2024 р.). м. Київ, 2024. Ч. 1. С. 685–689.

29. Харченко С. О. Наукові підходи до класифікації загроз інформаційній безпеці. Державне управління, 2019. № 2 (66). С. 191-197.

30. Шевчук М. О. До питання генези поняття інформаційної безпеки як складової національної безпеки. Науковий вісник Ужгородського національного університету. Серія «Право». 2023. Вип. 78. Ч. 2. С. 134–139.

31. Шемчук, В. В. Загрози інформаційній безпеці: проблеми визначення та подолання. Експерт: парадигми юридичних наук і державного управління, 2020. №(1(7), 285-296. (дата звернення: 24.05.2025).

32. Шемчук В. В. Інформаційна безпека та інформаційна оборона в контексті розвитку вітчизняної доктрини й законодавчої основи. Теорія та історія держави і права; історія політичних і правових учень, 2019.Том 30 (69) № 4. С. 31-37.

33. Cynet. Malware Protection: 6 Technologies to Protect Your Organization [Electronic resource]. 2025. URL: https://www.cynet.com/malware/malware-protection-6-technologies-to-protect-your-organization (дата звернення: 19.05.2025).

34. Free and Open Source Software : матеріали XV Міжнар. наук.-практ. конф., м. Харків, 13-14 лютого 2024 р. Харків: Харківський національний економічний університет імені Семена Кузнеця, 2024. 148 с.

35. Patsakis, C., Arroyo, D., & Casino, F. The Malware as a Service Ecosystem [Electronic resource] // arXiv preprint. 2024. URL: https://arxiv.org/abs/2405.04109 (дата звернення: 20.05.2025).

36. Salman, M., Khan, A., Usman, M., & Shahid, A. The Evolution of Cybersecurity Threats and Strategies for Effective Protection: A Review [Electronic resource] // ResearchGate. 2023. URL: https://www.researchgate.net/publication/385685835_The_Evolution_of_Cybersecurity_Threats_and_Strategies_for_Effective_Protection_A_review (дата звернення: 19.05.2025).

37. Schmitt, M. Securing the Digital World: Protecting Smart Infrastructures and Digital Industries with AI-enabled Malware and Intrusion Detection [Electronic resource] // arXiv preprint. 2023. URL: https://arxiv.org/abs/2401.01342 (дата звернення: 17.05.2025).

38. Tamrakar, A., & Patra, B. Cybersecurity Threats and Countermeasures: A Review [Electronic resource] // *ResearchGate*. 2018. URL: https://www.researchgate.net/publication/380335608_CYBERSECURITY_THREATS_AND_COUNTERMEASURES_A_REVIEW (дата звернення: 19.05.2025).

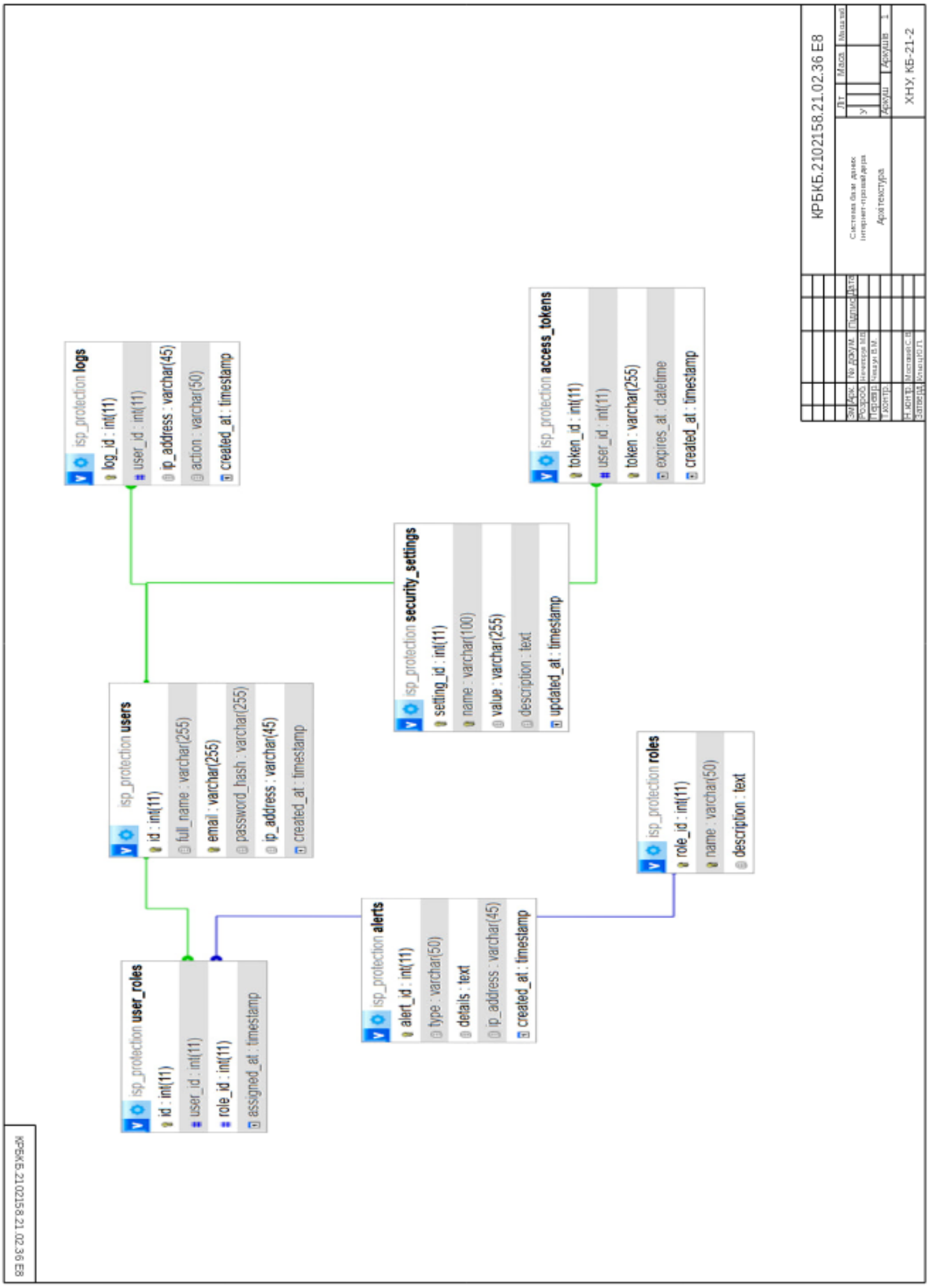
39. Virus Bulletin. APT vs Internet Service Providers – A Threat Hunter's Perspective [Electronic resource] // Virus Bulletin Conference Proceedings. 2020. URL: https://www.virusbulletin.com/virusbulletin/2020/10/apt-vs-internet-service-providers-threat-hunters-perspective (дата звернення: 18.05.2025).

40. World Economic Forum. Cybercrime Prevention Principles for Internet Service Providers [Electronic resource]. 2018. URL: https://www3.weforum.org/docs/WEF_Cybercrime_Prevention_ISP_Principles.pdf (дата звернення: 23.05.2025).

ДОДАТОК А

(обов’язковий)

Копія графічної частини



КРБКБ.2102158.21.02.36 ЕВ


| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | КРБКБ.2102158.21.02.36 ЕВ | | |
| Зм. | Арк. | № докум. | Підпис | Дата | Пропонування системи захисту інформаційних ресурсів | | Літ | Маса | Масштаб |
| Розроб. | | Нечесов М.В. | | | | | У | | |
| Перевір. | | Чекун В.М. | | | Візуалізація архітектури | | Аркуш | Аркушів 1 | |
| Н.контр. | | | | | | | ХНУ, КБ-21-2 | | |
| Н.контр. | | Мостовий С.В. | | | | | | | |
| Затверд. | | Кльоц Ю.П. | | | | | | | |

| Поле | Тип даних | Призначення |
|---|---|---|
| setting_id | INT, PK | Унікальний ідентифікатор налаштування |
| name | VARCHAR(100) | Назва параметра (ключ) |
| value | VARCHAR(255) | Значення параметра |
| description | TEXT | Пояснення або призначення параметра |
| updated_at | TIMESTAMP | Дата та час останнього оновлення |

## 5 security_settings

Створення: Тра 24 2025 р., 23:44
Останнє оновлення: Тра 24 2025 р., 23:44

| Стовпець | Тип | Атрибути | Нуль | За замовчуванням | Додатково | Посилання на | Коментарі | MIME |
|---|---|---|---|---|---|---|---|---|
| setting_id | int(11) | | Ні | | auto_increment | | | |
| name | varchar(100) | | Ні | | | | | |
| value | varchar(255) | | Ні | | | | | |
| description | text | | Так | NULL | | | | |
| updated_at | timestamp | | Ні | current_timestamp() | on update current_timestamp() | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | КРБКБ.2102158.21.02.36 Е8 | |
| Зм. | Арк. | № докум. | Підпис | Дата | Ключові параметри конфігурації системи захисту | Літ. / Маса / Масштаб |
| Розроб. | | Нечипорук М.В. | | | | У |
| Перевір. | | Чешун В. М. | | | | |
| Т.контр. | | | | | Структура таблиці security_settings | Аркуш 1 / Аркушів 1 |
| Н.контр. | | Мостовий С.Я. | | | | ХНУ, КБ-21-2 |
| Затверд. | | Кльоц Ю.Г. | | | | |

ДОДАТОК Б

(обов’язковий)

Лістинги Бази даних

Лістинг 1. - SQL-запит для створення таблиці users

```
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    full_name VARCHAR(255) NOT NULL,
    email VARCHAR(255) UNIQUE NOT NULL,
    password_hash VARCHAR(255) NOT NULL,
    ip_address VARCHAR(45),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Лістинг 2 – Створення таблиці logs

```
CREATE TABLE logs (
    log_id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT,
    ip_address VARCHAR(45),
    action VARCHAR(50),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id)
);
```

Лістинг 3 – Створення таблиці alerts

```
CREATE TABLE alerts (
    alert_id INT AUTO_INCREMENT PRIMARY KEY,
    type VARCHAR(50) NOT NULL,
    details TEXT,
    ip_address VARCHAR(45),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Лістинг 4 – Додавання тестових записів у таблицю users

```
INSERT INTO users (full_name, email, password_hash, ip_address)
VALUES
```

```
('R29yb2g=',                                          'ivan@ukr.net',
'9a04c3e9e85c6a9e9e1d4b02afc6eb29b79e...', '192.168.1.2'),
('T2xlbmE=',                                       'olena@gmail.com',
'07fd23ac99e3641b8f23f76c9dcf2b34fbe2...', '192.168.1.3'),
('U3RlcGFu',                                        'stepan@mail.ua',
'1af14cbe01572832e7b12f7cc9cf9f33f67e...', '192.168.1.4'),
('SXJ5bmE=',                                        'iryna@meta.ua',
'3a94fd004e28fcd4a3a4ee7ec5b5288446b2...', '192.168.1.5'),
('TWFrc3lt',                                     'maksym@gmail.com',
'88b203e2e281a3c70c8fa5a17b964ea7725f...', '192.168.1.6'),
('VmlrdG9y',                                        'viktor@ukr.net',
'62265df21aefef3ef88a2bb46cc7f84532a0...', '192.168.1.7'),
('T2tzYW5h',                                       'oksana@mail.com',
'4edc9f9aa987c78db925d0b4512cd07fcb3a...', '192.168.1.8'),
('TmF0YWxrYQ==',                                    'natalka@meta.ua',
'7f034aa8c28932f76b64b80db270a06df2a4...', '192.168.1.9'),
('WXVyaWk=',                                        'yurii@ukr.net',
'd1ce7a4f321e2dfe318ef4df9e10984a4933...', '192.168.1.10'),
('TWFyaWlh',                                     'mariia@gmail.com',
'bfb44dcac1eec22d66f7b704bc9c012f5ad7...', '192.168.1.11');
```

Лістинг 5 – Додавання записів до таблиці alerts

```
INSERT INTO alerts (type, details, ip_address)
VALUES
('SQL Injection', 'Виявлено символи: "OR 1=1--"', '192.168.1.12'),
('XSS Attack', 'Виявлено фрагмент: <script>alert(1)</script>',
'192.168.1.13'),
('Brute Force', 'Кілька спроб входу з невірним паролем',
'192.168.1.14'),
('Invalid Token', 'Недійсний токен автентифікації', '192.168.1.15'),
('Unknown Method', 'Запит із забороненим HTTP-методом: TRACE',
'192.168.1.16');
```

Примітка: Поле created_at додасться автоматично через DEFAULT CURRENT_TIMESTAMP

Лістинг 5 – Додавання записів до таблиці logs

```
INSERT INTO logs (user_id, ip_address, action)
VALUES
(1, '192.168.1.2', 'login'),
(3, '192.168.1.4', 'logout'),
(5, '192.168.1.6', 'login'),
(2, '192.168.1.3', 'login'),
(4, '192.168.1.5', 'error');
```

Примітка:Поле created_at додасться автоматично.

Лістинг 6 – SQL-запит для створення таблиці access_tokens

```
CREATE TABLE access_tokens (
    token_id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT,
    token VARCHAR(255) NOT NULL UNIQUE,
    expires_at DATETIME NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id)
);
```

Лістинг 7– Приклад заповнення таблиці access_tokens

```
INSERT INTO access_tokens (user_id, token, expires_at)
VALUES
(1, 'a1b2c3d4e5f6g7h8i9j0', '2025-05-25 23:59:00'),
(2, 'x9y8z7w6v5u4t3s2r1q0', '2025-05-26 00:15:00'),
(3, 'l0k9j8h7g6f5d4s3a2q1', '2025-05-26 01:00:00');
```

Лістинг 8 – SQL-запит для створення таблиці roles

```
CREATE TABLE roles (
    role_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(50) UNIQUE NOT NULL,
    description TEXT
);
```

Лістинг 9 – Приклад заповнення таблиці roles

```
INSERT INTO roles (name, description)
VALUES
('admin', 'Повний доступ до системи, керування користувачами та
безпекою'),
('user', 'Звичайний зареєстрований користувач із базовими правами
доступу'),
('moderator', 'Може переглядати та блокувати шкідливі запити'),
('auditor', 'Перегляд журналів, без можливості змін');
```

Лістинг 10 – SQL-запит для створення таблиці user_roles

```
CREATE TABLE user_roles (
    id INT AUTO_INCREMENT PRIMARY KEY,
```

```
    user_id INT,
    role_id INT,
    assigned_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id),
    FOREIGN KEY (role_id) REFERENCES roles(role_id)
);
```

Лістинг 11 – Приклад заповнення таблиці user_roles

```
INSERT INTO user_roles (user_id, role_id)
VALUES
(1, 1),  -- Ivan: admin
(2, 2),  -- Olena: user
(3, 2),  -- Stepan: user
(4, 3),  -- Iryna: moderator
(5, 4);  -- Maksym: auditor
```

Лістинг 12 – SQL-запит для створення таблиці security_settings

```
CREATE TABLE security_settings (
    setting_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) UNIQUE NOT NULL,
    value VARCHAR(255) NOT NULL,
    description TEXT,
    updated_at  TIMESTAMP  DEFAULT  CURRENT_TIMESTAMP  ON  UPDATE
CURRENT_TIMESTAMP
);
```

Лістинг 13 – Приклад заповнення таблиці security_settings

```
INSERT INTO security_settings (name, value, description)
VALUES
('max_login_attempts', '5', 'Максимальна кількість спроб входу перед
блокуванням'),
('session_timeout_minutes',   '30',   'Час   неактивності   перед
завершенням сесії'),
('password_min_length', '8', 'Мінімальна довжина пароля'),
('security_level', 'high', 'Рівень захисту системи (low, medium,
high)'),
('allowed_methods', 'GET,POST', 'Дозволені HTTP-методи у запитах до
сервера');
```

Завідувачу кафедри кібербезпеки
к.т.н., доц. Кльоцу Ю.П.
Нечипорука Михайла Вікторовича
Студента ФІТ, 4 курсу, групи КБ-21-2

ЗАЯВА

З правилами чинного Положення про систему забезпечення академічної доброчесності у Хмельницькому національному університеті, згідно з яким виявлення академічного плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту і застосування заходів дисциплінарної та академічної відповідальності, ознайомлений. Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність академічного плагіату оповіщений та надаю свою згоду на обробку й збереження університетом моєї роботи в інституційному репозитарії Хмельницького національного університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-обчислювального комплексу StrikePlagiarism та/або програмнотехнічного засобу Anti-Plagiarism) і використання роботи для виявлення академічного плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення текстових збігів в роботах.

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

25.05.25
дата

підпис

# Anti-Plagiarism (UA) v-15.281 Educational

**The maximum coincidence with one document 1.0%**

Dictionaries check: en_US, ru_RU, ua_UA. **Errors in the documents: 10%**

| | Document | | Sum coincidence on the DB | |
|---|---|---|---|---|
| | Symbols | Lexemes | Symbols | Lexemes |
| ID: 245264<br>Title: Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення<br>Added in a DB: 2025-06-12<br>Authors: Нечипорук Михайло Вікторович<br>Heads: Чешун В.М,<br>Consultants:<br>Opponents: | 77857 | 1176 | 1262 (2%) | 18 (2%) |

Plagiarism sources

| ID | Description | Plagiarism presence in the document | |
|---|---|---|---|
| | | Symbols | Lexemes |

**Протокол аналізу звіту подібності науковим керівником**

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

**Автор:** Нечипорук Михайло Вікторович

**Співавтор:**

**Назва:** Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення

**Науковий керівник:**

**Підрозділ:** Кафедра кібербезпеки

**Коефіцієнт подібності 1:**1.4%

**Коефіцієнт подібності 2:**0%

**Мікропробіли:** 0

**Заміна букв:** 0

**Інтервали:** 0
**Білі знаки:** 0 **Дата створення звіту:** 2025-06-12 09:30:55.0

**Після аналізу Звіту подібності констатую наступне:**

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

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

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

Обгрунтування:

11.06.2025р.

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КІБЕРБЕЗПЕКИ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення

Автор: Нечипорук Михайло Вікторович

Спеціальність: 125 – Кібербезпека

Освітня програма:Кібербезпека

Науковий керівник: Віктор ЧЕШУН, канд. техн. наук, доцент

Після аналізу звіту подібності зроблено такий висновок:

| № | Висновок | Позначка про відповідність |
|---|---|---|
| 1 | Запозичення, виявлені в роботі, є законними і не є плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних). Робота приймається до захисту. | відповідає |
| 2 | Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи. | |
| 3 | Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат. | |
| 4 | Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту. | |
| 5 | Інше: | |

Підтвердження:

Оригінальність тексту роботи за результатами перевірки системою StrikePlagiarism складає 99,1%, оригінальність тексту роботи за результатами перевірки системою Anti-Plagiarism складає 99%.

Згідно з правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 24.09.2024, авторська робота, обсяг оригінального тексту у відсотках до загального обсягу матеріалу в якій складає 90-100%, визначається роботою з високою унікальністю тексту і допускається до захисту.

Керівник роботи Віктор ЧЕШУН

Гарант ОП Віктор ЧЕШУН

Завідувач кафедри кібербезпеки Юрій КЛЬОЦ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

# РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

освітнього ступеня «бакалавр»

Студент Нечипорук Михайло Вікторович

Тема Система захисту інформаційних ресурсів інтернет-провайдера від шкідливого програмного забезпечення

Спеціальність 125 – Кібербезпека

**Обсяг кваліфікаційної роботи освітньо-кваліфікаційного рівня «бакалавр»:**

кількість листів креслень 3; кількість сторінок записки 70.

1. Короткий зміст роботи та прийнятих рішень У кваліфікаційній роботі досліджено актуальну проблему кібербезпеки інтернет-провайдерів та запропоновано систему захисту їхніх інформаційних ресурсів від шкідливого програмного забезпечення. Проведено аналіз видів шкідливого ПЗ, класифікацію загроз, досліджено особливості інфраструктури провайдера. Побудовано архітектуру захисної системи, яка включає засоби шифрування персональних даних, фільтрацію запитів, автентифікацію та журналювання. Реалізація системи виконана з використанням PHP та MySQL із детальною документацією.

2. Висновок про відповідність кваліфікаційної роботи завданню Кваліфікаційна робота повністю відповідає завданню, поставленому у технічному завданні та змісті проєкту. В роботі поєднано теоретичні засади захисту інформації з практичною реалізацією комплексної захисної системи, що відповідає сучасним вимогам кібербезпеки.

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У вступі обґрунтовано актуальність теми, визначено мету, об'єкт та предмет дослідження. У першому розділі досліджено теоретичні аспекти інформаційної безпеки, зокрема роль інтернет-провайдерів, загрози та типи шкідливого ПЗ. У другому розділі спроєктовано архітектуру системи, структуру бази даних, методи шифрування, що демонструє добру обізнаність автора у сучасних ІБ-практиках. Третій розділ присвячено реалізації системи: створено SQL-базу, застосовано засоби захисту, проведено тестування та оцінку ефективності. Робота базується на актуальних технологіях та враховує чинне законодавство України у сфері захисту інформації.

4. Позитивні сторони Робота має високу прикладну цінність. Автор запропонував цілісну систему, що охоплює як технічні, так і організаційні заходи безпеки. Значну увагу приділено забезпеченню конфіденційності персональних даних користувачів, веденню журналів активності, аналізу SQL-запитів. Проєкт реалізований із врахуванням реалій діяльності інтернет-провайдерів.

5. Негативні сторони роботи До недоліків можна віднести обмежене охоплення питань безпеки на рівні мережевої інфраструктури (наприклад, захист маршрутизаторів, DNS та міжмережевих екранів). Також відсутні детальні сценарії впровадження у масштабованому середовищі або з урахуванням хмарних сервісів.

6. Оцінка графічного оформлення та пояснювальної записки роботи Графічне оформлення кваліфікаційної роботи відповідає темі роботи та виконане з дотриманням стандартів. Загалом графічна частина виконана на належному рівні, а пояснювальна записка оформлена відповідно до встановлених вимог..

7. Відгук про роботу в цілому Кваліфікаційна робота заслуговує позитивної оцінки, оскільки весь матеріал роботи є структурованим, чітким та послідовним. Усі розділи роботи мають логічну послідовність, що сприяє зрозумінню викладеного матеріалу в рамках теми роботи. Графічний матеріал допомагає наочно продемонструвати доцільність та ефективність прийнятих рішень для досягнення мети.

8. Інші зауваження

9. Оцінка кваліфікаційної роботи Враховуючи всі позитивні та негативні сторони представленої кваліфікаційної роботи, можна зробити висновок, що вона заслуговує оцінки «задовільно».

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи)
Говорущенко Тетяна Олександрівна.
Доктор технічних наук, професор, декан факультету інформаційних технологій

« 16 » червня 2025