Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА

Городецької Анни Олександрівни

на здобуття ступеня вищої освіти Бакалавра

Система захисту інформаційних ресурсів центру надання адміністративних послуг

Галузь знань 12 – Інформаційні технології
Спеціальність 125 – Кібербезпека
Освітня програма Кібербезпека

Шифр КРБКБ.220236.22.02.24 ПЗ

Виконала студенткам 4 курсу група КБ-22-2 Анна ГОРОДЕЦЬКА
Керівник д-р філософії Наталія ПЕТЛЯК
Нормоконтролер д-р філософії Наталія ПЕТЛЯК

До захисту допускаю:
Завідувач кафедри кібербезпеки Юрій КЛЬОЦ

3 06 2026 р.

Хмельницький 2026

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

9 січня 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Городецькій Анні Олександрівні

1 Тема роботи Система захисту інформаційних ресурсів центру надання адміністративних послуг
Керівник роботи доктор філософії Наталія ПЕТЛЯК
Затверджено наказом ректора університету від 8 січня 2026 р. № 7
2 Строк подання студентом кваліфікаційної роботи на кафедру 27 травня 2026 р.
3 Вихідні дані до роботи Проаналізувати особливості функціонування інформаційної системи центру надання адміністративних послуг, зокрема структуру інформаційних ресурсів, типи оброблюваних персональних даних, канали взаємодії із державними реєстрами та користувачами, а також організацію доступу співробітників до інформації. Дослідити нормативно-правові вимоги у сфері захисту інформації та персональних даних, визначити основні загрози й вразливості інформаційних ресурсів ЦНАП. Проаналізувати існуючі технічні та організаційні засоби захисту інформації, їх переваги та недоліки. Розробити модель загроз і порушника, спроєктувати архітектуру системи захисту, обґрунтувати вибір методів і засобів забезпечення інформаційної безпеки. - Реалізувати підсистеми контролю доступу, криптографічного захисту та аудиту подій.
4 Зміст пояснювальної записки (перелік питань, які потрібно розробити) Аналіз сучасного стану захисту інформаційних ресурсів центру надання адміністративних послуг. Особливості функціонування інформаційної системи центру надання адміністративних послуг. Ідентифікація загроз та вразливостей інформаційних ресурсів. Проєктування системи захисту інформаційних ресурсів. Формування вимог до системи захисту. Розроблення архітектури системи захисту. Реалізація підсистем захисту. Аналіз відповідності вимогам інформаційної безпеки.
5 Перелік графічного матеріалу (із зазначенням обов'язкових креслень) Структурна схема інформаційної системи Центру надання адміністративних послуг. Модель загроз інформаційній безпеці ЦНАП. Модель порушника. Структурна схема системи захисту інформації. Алгоритм функціонування системи захисту інформації.

6 Консультанти розділів кваліфікаційної роботи

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

7 Дата видачі завдання 12 січня 2026 р.

КАЛЕНДАРНИЙ ПЛАН

Назва етапів (розділів) кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
Вибір і затвердження теми кваліфікаційної роботи	Лютий	
Ознайомлення з предметною областю	Лютий	
Дослідження існуючих рішень	Лютий	
Визначення основних загроз інформаційним ресурсам	Лютий	
Формування вимог до системи захисту інформації	Березень	
Розроблення структури та архітектури системи захисту	Березень	
Розроблення алгоритму функціонування системи захисту	Квітень	
Реалізація підсистем захисту	Квітень	
Аналіз відповідності системи встановленим вимогам	Травень	
Оформлення пояснювальної записки та графічної частини	Травень	
Захист кваліфікаційної роботи	Червень	

Студент Анна ГОРОДЕЦЬКА

Керівник кваліфікаційної роботи Наталія ПЕТЛЯК

АНОТАЦІЯ

Тема кваліфікаційної роботи: Система захисту інформаційних ресурсів центру надання адміністративних послуг.

Автор роботи: Городецька Анна Олександрівна.

Керівник роботи: Петляк Наталія Сергіївна.

Пояснювальна записка: 60 с., 2 додатки, 5 рисунків, 14 таблиць, 46 джерел.

Графічна частина: 4 плаката.

Ключові слова: інформаційна безпека, захист інформаційних ресурсів, Центр надання адміністративних послуг, контроль доступу, аудит подій, рівень довіри.

Метою кваліфікаційної роботи є підвищення рівня захищеності інформаційних ресурсів у системах Центру надання адміністративних послуг шляхом впровадження механізмів контролю доступу та аналізу користувацької активності.

У межах роботи проаналізовано основні загрози інформаційній безпеці та розроблено програмний прототип системи, що реалізує автентифікацію користувачів, ведення журналу подій, оцінювання ризикової активності та автоматичне реагування на підозрілі дії. Додатково розглянуто сучасні підходи до побудови систем контролю доступу та методи виявлення аномальної поведінки користувачів. Особливу увагу приділено журналюванню подій як ключовому джерелу даних для аналізу безпеки інформаційних систем. Запропонований підхід дозволяє підвищити оперативність реагування на потенційні загрози та зменшити ризики витоку даних.

Отриманим результатом є програмна система захисту, яка забезпечує моніторинг дій користувачів, фіксацію змін у системі та зменшення ймовірності несанкціонованого доступу до персональних даних.

25.05.2026

ABSTRACT

Subject of qualification work: Information resources protection system of the administrative services center.

Author: Horodetska Anna Oleksandrivna.

Head of work: Petliak Nataliia Sergiivna.

Explanatory note: 60 p., 2 appendices, 5 figures, 14 tables, 46 sources

Graphic part: 1 posters, 10 presentation slides.

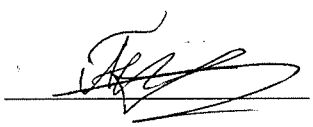
Keywords: information security, protection of information resources, Center for Administrative Services, access control, event auditing, trust level.

The aim of the qualification work is to improve the level of protection of information resources in Center for Administrative Services systems by implementing access control mechanisms and analyzing user activity.

The study analyzes the main threats to information security and develops a software prototype system that provides user authentication, event logging, risk activity assessment, and automatic response to suspicious actions. Additionally, modern approaches to access control system design and methods for detecting anomalous user behavior are considered. Special attention is paid to event logging as a key data source for security analysis. The proposed approach improves the responsiveness to potential threats and reduces the risk of data leakage.

The obtained result is a software security system that enables monitoring of user actions, recording system changes, and reducing the likelihood of unauthorized access to personal data.

25.05.2026



ЗМІСТ

ВСТУП

З розвитком електронних державних сервісів зростають і вимоги до захисту інформації. Центри надання адміністративних послуг є одним із основних місць, де громадяни взаємодіють із державою, подають документи та отримують необхідні довідки. Під час цього обробляється велика кількість персональних даних, тому порушення вимог захисту інформації можуть призвести до суттєвих наслідків. Під час аналізу роботи ЦНАП встановлено, що на постійній основі інформаційна система взаємодіє з державними реєстрами та користувачами. Це дозволяє швидко надавати послуги, але водночас створює додаткові ризики. Зокрема, можливий несанкціонований перегляд даних без службової необхідності або використані не за призначенням. Важливим фактором ризику залишається людський чинник. Помилки працівників трапляються досить часто. Крім того, обробка інформації в ЦНАП регламентується законодавством України. Законодавство встановлює вимоги до її захисту, але на практиці виконати всі ці вимоги не завжди просто. Система повинна залишатися зручною для користувачів і одночасно безпечною.

Обсяг оброблюваних даних зростає, а разом із цим збільшуються і ризики їх втрати або витоку. Тому питання побудови надійної системи захисту інформації для ЦНАП є важливим. Необхідно розробити системи захисту інформаційних ресурсів Центру надання адміністративних послуг. Для цього необхідно проаналізувати особливості роботи інформаційної системи та визначити основні загрози і вразливості. Також дослідити існуючі засоби захисту. На основі результатів запропонувати власні рішення для підвищення рівня безпеки. Отримані результати можуть бути використані для покращення роботи подібних систем та підвищення рівня інформаційної безпеки. Особлива увага приділяється автоматизації контролю за діями персоналу в режимі реального часу. Це дозволить вчасно виявляти підозрілу активність і мінімізувати ризики, пов’язані з людським фактором.

1 АНАЛІЗ СУЧАСНОГО СТАНУ ЗАХИСТУ ІНФОРМАЦІЙНИХ РЕСУРСІВ ЦЕНТРУ НАДАННЯ АДМІНІСТРАТИВНИХ ПОСЛУГ

1.1 Особливості функціонування інформаційної системи центру надання адміністративних послуг

Інформаційна система Центру надання адміністративних послуг (ЦНАП) є ключовим елементом його функціонування. Забезпечує виконання більшості процесів, пов’язаних із наданням послуг громадянам. Вона об’єднує локальні обчислювальні ресурси з державними інформаційними мережами та забезпечує взаємодію з державними реєстрами в межах єдиного інформаційного середовища. Основне призначення системи полягає в автоматизації збору, обробки, зберігання та передачі персональних даних. Ці дані необхідні для отримання адміністративних послуг. Особливістю функціонування такої системи є постійна взаємодія з великою кількістю зовнішніх реєстрів, зокрема Державним реєстром речових прав на нерухоме майно, Єдиним державним демографічним реєстром та іншими інформаційними ресурсами. У результаті робоче місце адміністратора фактично стає точкою, де зосереджується значний обсяг конфіденційної інформації. Це підвищує вимоги до організації захисту даних і потребує більш уважного підходу до забезпечення інформаційної безпеки [1].

Робота системи починається з моменту звернення заявника та введення його даних у систему електронної черги або в картку звернення. Система повинна забезпечувати високу швидкість відгуку при паралельному зверненні до зовнішніх шлюзів (наприклад, через систему «Трембіта»). Аналіз показує, що через різну пропускну здатність каналів зв’язку та неоднакові протоколи обміну даними між відомствами, інформаційна система часто перебуває у стані підвищеного навантаження [2]. Це змушує адміністраторів використовувати проміжні програмні засоби для копіювання чи тимчасового збереження даних. Ці дії призводять до розширення потенційної поверхні атаки та створюють додаткові ризики витоку інформації ще до її фіксації в основному реєстрі [3]. Структура інформаційних ресурсів ЦНАП є неоднорідною. Кожен тип даних має свій життєвий цикл, джерело

походження та ступінь критичності для національної безпеки чи приватного життя особи. Для детального розуміння необхідно провести класифікацію інформаційних ресурсів визначивши їхню роль у функціонуванні системи (наведено в таблиці 1.1).

Таблиця 1.1 Класифікація інформаційних ресурсів ІС ЦНАП

№	Категорія інформаційного ресурсу	Склад та приклади даних (об'єкти захисту)	Місце обробки та фізичного зберігання	Суб'єкти, що мають санкціонований доступ	Рівень критичності при порушенні конфіденційності
1	Біометричні ідентифікатори	Оцифровані відбитки пальців, зразки власноручного підпису, фотозображення обличчя	Сховища типу BLOB у централізованих державних реєстрах (МВС, ДМС)	Адміністратори з правом оформлення паспортних документів	Критичний
2	Персональні дані (ПД)	ПІБ, дата народження, серія та номер паспорта, ІПН, адреса реєстрації	Локальні бази СЕД ЦНАП, сервери територіальних громад	Всі адміністратори фронт-офісу, спеціалісти відділів	Високий
3	Відомості про майнові права	Номери свідоцтва про право власності, кадастрові номери, дані про обтяження (арешти)	Державна реєстраційна система (Мін'юст), земельний кадастр	Державні реєстратори, нотаріуси, уповноважені особи	Високий
4	Службово-технологічна інформація	Історія руху справи, електронні резолюції, маршрути документів між відділами	Внутрішня база даних системи документообігу (Back-office)	Керівництво ЦНАП, профільні спеціалісти	Середній
5	Системні журнали	Записи про автентифікацію, IP-адреси сесій, історія SQL-запитів до бази даних	Журнали аудиту на серверному обладнанні установи	Системні адміністратори, офіцери безпеки	Високий

На основі даних (таблиця 1.1) встановлено, що найбільш вразливими є системні журнали та біометричні дані, доступ до яких повинен бути максимально

обмеженим. Якщо у порушника зявилась можливість змінювати записи у журналах подій він може приховати факти копіювання персональних даних або несанкціонованого доступу до майнових реєстрів [4]. У реальних умовах роботи ЦНАП один адміністратор протягом робочого дня опрацьовує значну кількість звернень. Через це забезпечити постійний контроль за кожною дією практично неможливо. У результаті виникає ситуація, коли рівень безпеки залежить від людського фактора, а не лише від застосованих технічних засобів захисту.

Зазвичай системи ЦНАП має віддалені робочі місця (ВРМ) у територіальних громадах. Вони підключаються до центрального сервера через VPN-канали. Рівень захищеності системи визначається найменш захищеним елементом а саме віддаленим персональним комп'ютером, що може не мати належного рівня захисту або оновленого програмного забезпечення. Інформаційна система працює в умовах постійного конфлікту між вимогою територіальної доступності послуг та необхідністю суворої ізоляції даних. ІС ЦНАП є динамічним середовищем, яке постійно змінюється внаслідок впровадження нових адміністративних послуг та оновлення державних регламентів. Кожна така зміна впливає на систему захисту та потребує адаптації. У результаті традиційні статичні методи стають не ефективними. Висока концентрація персональних даних, людський фактор під час масового обслуговування, а також складність міжвідомчої взаємодії призводять до необхідності розроблення активної підсистеми захисту, яка зможе автоматично ідентифікувати аномальну активність користувачів [5].

1.2 Аналіз нормативно-правових вимог щодо захисту інформації

Під час аналізу нормативно-правових аспектів діяльності ЦНАП насамперед привертає увагу значний обсяг паперового та електронного документообігу, який фактично є основою забезпечення інформаційної безпеки. Державні установи в Україні функціонують в умовах жорсткого нормативного регулювання, що є обґрунтованим, оскільки обробляються персональні дані великої кількості

громадян. Базовим нормативним актом у цій сфері є Закон України «Про захист персональних даних», який визначає основні принципи обробки інформації, зокрема порядок її збору, зберігання та використання. Разом із тим на практиці реалізація вимог цього закону пов’язана з рядом організаційних складнощів. Зокрема, кожен заявник повинен надати згоду на обробку своїх даних, а система має забезпечити її належну фіксацію. У таких умовах формується правове середовище, у якому навіть незначні помилки в роботі програмного забезпечення або діях персоналу можуть призвести до юридичних наслідків [6].

Законодавчі вимоги становлять лише частину загальної системи захисту інформації, адже разом із ними діють і технічні вимоги до створення комплексної системи захисту інформації (КСЗІ). Фактично йдеться про сукупність установлених технічних вимог, без яких жодна державна інформаційна система не може повноцінно функціонувати. Основну роль у регулюванні цих процесів відіграє Держспецзв’язок. Зазначений орган формує відповідні нормативні вимоги, зокрема через документи типу НД ТЗІ 2.5-004-99 [7]. Попри відносну складність нормативного формулювання, зазначені вимоги фактично орієнтовані на реалізацію контролю доступу до інформації та фіксацію всіх операцій, що виконуються в системі. Окрему увагу приділено доступу до публічної інформації та регламентації прав громадян на отримання відомостей [8]. Також існують галузеві нормативи, що визначають перелік обов’язкових завдань із захисту інформації від несанкціонованого доступу [9]. Для ЦНАП це обумовлює необхідність підтвердження відповідності системи захисту встановленим вимогам інформаційної безпеки. Без цього підключення до державних реєстрів, таких як реєстр речових прав або демографічний реєстр, є неможливим [10]. Саме тому дотримання технічних вимог є реальною умовою функціонування всієї системи, що має відповідати міжнародним та державним стандартам управління безпекою [11]. Від дотримання цих правил залежить, чи зможе ЦНАП безпечно працювати щодня без збоїв та витоку даних. Щоб краще зрозуміти, як саме нормативні вимоги впливають на технічну реалізацію системи захисту, доцільно розглянути їх у вигляді таблиці 1.2.

Таблиця 1.2. Порівняльний аналіз нормативних вимог та реалізації

№	Назва нормативного акту	Основна вимога до безпеки інформації	Технічний спосіб виконання в проєкті	Наслідки ігнорування вимоги
1	ЗУ «Про захист персональних даних»	Мінімізація доступу та цілісність інформації.	Впровадження динамічного маскування полів (Data Masking).	Витік ПД та кримінальна відповідальність персоналу.
2	НД ТЗІ 2.5-004-99 (КСЗІ)	Створення захищених журналів аудиту дій користувача.	Побудова ланцюгів хешування для логів (Immutable Logs).	Анулювання атестата відповідності системи.
3	ЗУ «Про електронні довірчі послуги»	Автентифікація через кваліфікований підпис (КЕП).	Інтеграція шлюзів перевірки токенів та сертифікатів.	Неможливість юридичного підтвердження операцій.
4	Постанови КМУ (Регламент ЦНАП)	Розмежування прав доступу за посадовими обов'язками.	Рольова модель доступу з контролем аномалій поведінки.	Хаос у правах доступу та внутрішні зловживання.
5	Стандарт ISO/IEC 27001	Постійний моніторинг та оцінка ризиків у реальному часі.	Алгоритм розрахунку Trust Score для кожної сесії.	Запізніла реакція на інциденти та злам системи.

Аналіз практичного застосування нормативних вимог у діяльності установ дозволяє відзначити наявність певного протиріччя між забезпеченням інформаційної безпеки та швидкістю обслуговування громадян. Зокрема, використання кваліфікованого електронного підпису для підтвердження кожної операції потребує виконання додаткових дій користувачем, що може впливати на тривалість обробки звернень [12]. Водночас відповідно до законодавчих вимог у сфері електронних довірчих послуг, застосування таких механізмів є необхідною

умовою забезпечення ідентифікації користувача та підтвердження правомірності виконаних дій. У зв’язку з цим актуальним є завдання розроблення технічних рішень, спрямованих на автоматизацію виконання вимог безпеки з мінімальним впливом на робочий процес персоналу [13]. Окремої уваги заслуговують європейські підходи до захисту персональних даних, зокрема принципи privacy by default та privacy by design [14]. У контексті євроінтеграційних процесів підвищується актуальність впровадження відповідних підходів [15]. Незважаючи на те, що вони ще не повністю інтегровані в наше законодавство, але відповідні принципи поступово інтегруються в практику функціонування інформаційних систем [16]. Це означає, що система не повинна показувати адміністратору більше, ніж йому треба для роботи. У процесі обробки звернень громадян має забезпечуватися принцип мінімізації даних, відповідно до якого користувач отримує доступ лише до тієї інформації, яка є необхідною для виконання конкретного завдання [17,18]. Це означає, що навіть за наявності технічної можливості доступу до різних державних реєстрів, обсяг відображуваних даних повинен бути обмежений функціональними обов’язками працівника. Такий підхід відповідає сучасним вимогам захисту інформації та сприяє зниженню ризиків несанкціонованого використання персональних даних. Аналіз нормативно-правової бази свідчить про те, що система захисту інформації формується на основі чітко визначених вимог, які регламентують порядок доступу, обробки та зберігання даних. Неврахування хоча б однієї з встановлених норм може призвести до невідповідності системи вимогам інформаційної безпеки. У зв’язку з цим розроблення модулів захисту доцільно здійснювати з урахуванням конкретних нормативних положень, що регламентують відповідні аспекти функціонування інформаційних систем [19]. Це дозволяє реалізувати програмні рішення, а й створити юридично обґрунтовану систему, результати функціонування якої можуть бути використані під час перевірок або розгляду спірних ситуацій. Поєднання правових вимог і технічних рішень є важливою умовою забезпечення належного рівня захисту інформаційних ресурсів у державних інформаційних системах [20].

1.3. Ідентифікація загроз та вразливостей інформаційних ресурсів

Після аналізу вимог до функціонування системи відповідно до законодавства доцільно розглянути практичні умови її використання, у межах яких можливе виникнення помилок або навмисних порушень [21]. Загрози для ЦНАП пов’язані не лише із зовнішніми атаками, але й з внутрішніми факторами, які часто зумовлені людським фактором, зокрема помилками або зниженням уваги користувачів [22]. Однією з ключових вразливостей об’єкта дослідження є надмірна довіра до користувача. В рамках концепції Zero Trust Architecture, довіра до будь-якого суб'єкта системи має постійно перевірятися [23]. Традиційно внутрішні користувачі розглядаються як довірені суб’єкти системи. Разом з тим результати досліджень свідчать про інше. Більшість витоків стаються не через злам ззовні, а через те, що хтось вирішив здійснити несанкціонований доступ до даних третіх осіб або без завершення активної сесії користувача [24]. Це створює суттєву вразливість, яку не закрити жодним антивірусом, бо з точки зору сервера - це виглядає як правомірна активність користувача. Тут варто згадати про технічний бік питання, а саме про вразливості, які створюємо, намагаючись зробити роботу зручнішою [25]. Наприклад, використання віддаленого доступу для системних адміністраторів або робота через VPN-тунелі з віддалених точок у громадах. Такий підхід забезпечує зручність, однак це створює додаткову точку потенційного несанкціонованого доступу [26]. Якщо хоча б один домашній комп’ютер адміністратора, який підключився до мережі центру, заражений шкідливим програмним забезпеченням для викрадення облікових даних, рівень захищеності системи значно знижується. Поширеною практикою є збереження паролів у браузері з метою економії часу, а це створює суттєвий ризик для безпеки системи. З метою систематизації виявлених загроз, сформовано матрицю, яка відображає джерела загроз, сценарії їх реалізації та можливі наслідки в межах інформаційної інфраструктури. При розробці таких моделей оцінки доцільно використовувати методики виявлення аномалій поведінки користувачів [27]. Основні результати наведено у таблиці 1.3, яка розміщена на наступній сторінці.

Таблиця 1.3. Матриця загроз та вразливостей системи

№	Джерело загрози	Конкретний сценарій (вразливість)	Об'єкт атаки	Ймовірність виникнення	Ступінь руйнівного впливу
1	Внутрішній інсайдер	Несанкціонований перегляд ПД без реєстрації справи.	Реєстр територіальної громади.	Висока	Середній (витік приватної інформації)
2	Системний адміністратор	Видалення записів у логах для приховування дій.	Журнали аудиту (Database Logs).	Низька	Критичний (втрата підзвітності)
3	Зовнішній хакер	Брутфорс паролів через вразливість VPN-шлюзу.	Мережевий периметр установи.	Середня	Високий (повний контроль над АРМ)
4	Шкідливе ПЗ	Шифрувальник, занесений через особисту пошту.	Локальні файлові сервери, СЕД.	Середня	Критичний (зупинка роботи центру)
5	Людська недбалість	Залишений КЕП (токен) у роз’ємі розблокованого ПК.	Державні реєстри (РРП, ЄДР).	Висока	Високий (юридичні маніпуляції)

Особливо увагу слід приділити загрозі так званого «тихого витоку» інформації. Це коли дані крадуть не гігабайтами за один раз, а по одній анкеті щодня. Така активність майже не помітна для стандартних систем моніторингу, бо вона вписується в норми трафіку. Але за рік можна накопичити значний обсяг даних. Це свідчить про те, що традиційні підходи до захисту, які орієнтовані переважно на виявлення масових або різких відхилень, є недостатніми [28]. Необхідним є впровадження механізмів, здатних аналізувати поведінку користувачів, зокрема виявляти нетипові дії, такі як часте відкриття записів із подібними характеристиками або систематичний доступ до інформації, що не пов’язана з поточними завданнями.

Не можна забувати про загрозу втрати цілісності. У ЦНАП це виглядає як «випадкова» зміна адреси в реєстрі чи виправлення дати народження. Якщо система

не вміє перевіряти цілісність кожного запису у режимі реального часу, у результаті може сформуватися база, де значна частина інформації є недостовірною [29]. Така ситуація може мати більш критичні наслідки, ніж витік інформації, тоді як підміну даних часто виявляють лише тоді, коли людина приходить за довідкою і бачить там нісенітницю. Це руйнує довіру до державних інституцій, а в умовах війни та цифровізації - це серйозний удар по іміджу всієї системи.

Проведений аналіз показав, що значна частина ризиків пов’язана не лише з технічними засобами, а й з діями працівників, які безпосередньо працюють з інформацією. Кожну виявлену вразливість можна розглядати як окремий напрям для вдосконалення системи захисту. Тобто мова йде не просто про усунення проблем, а про побудову більш продуманої системи, яка здатна реагувати на підозрілу активність ще на ранніх етапах. У результаті такий підхід дозволяє перейти від звичайного контролю до більш активного управління ризиками, що є важливим для забезпечення належного рівня інформаційної безпеки [30].

1.4 Аналіз існуючих засобів і методів захисту інформації

Під час переходу від аналізу загроз до оцінювання наявних засобів захисту до того, що реально пропонує ринок програмного забезпечення для наших ЦНАПів, виявляються як переваги, так і обмеження. З одного боку, існують потужні комплекси типу «Вулик» або різні модифікації систем електронного документообігу, які формально забезпечують належний рівень регламентованого захисту [31]. Вони мають сертифікати, пройшли значну кількість експертиз і нібито гарантують повну безпеку. Проте, попрацювавши з такими системами на практиці, виявлено одну фундаментальну проблему: вони занадто статичні. Більшість існуючих засобів захисту зосереджені на тому, щоб запобігти несанкціонованому доступу, спираючись на загальні положення захисту від несанкціонованого доступу [32]. Але їх функціональні можливості є обмеженими у разі виявлення аномальної поведінки внутрішніх користувачів. Більшість стандартних систем захисту (КСЗІ)

фокусуються на антивірусному захисті, фаєрволах та базовому логуванні подій [33]. Але проблема в тому, що журнали подій, як правило, не аналізуються в режимі реального часу. Вони просто накопичуються гігабайтами на серверах, і факт витоку даних виявляється із запізненням, часто вже після появи бази у відкритому доступі на чорному ринку. Це реактивний підхід до забезпечення безпеки, який не запобігає злочину, а лише констатує його [34]. З метою виявлення невідповідностей існуючих засобів сучасним вимогам безпеки проведено їх порівняльний аналіз, що передбачає оцінювання функціональних можливостей із урахуванням запропонованих рішень (таблиця 1.4). Водночас важливим є дотримання мінімальних вимог захисту інформаційних систем [35].

Таблиця 1.4. Порівняння ефективності засобів захисту

№	Критерій оцінки ефективності	Стандартні засоби (напр. «Вулик», СЕД)	Проектована підсистема моніторингу	Вплив на загальну безпеку
1	Метод контролю доступу	Статична рольова модель (RBAC).	Динамічна адаптивна модель (ABAC).	Запобігання використанню надлишкових прав.
2	Виявлення внутрішніх загроз	Тільки ручний аналіз журналів подій.	Автоматичний аналіз аномалій (Real-time).	Миттєва ідентифікація підозрілих дій персоналу.
3	Захист від масового копіювання	Практично відсутній (ліміти не діють).	Технологія Throttling (штучне гальмування).	Унеможливлення швидкого вивантаження бази ПД.
4	Стійкість до модифікації логів	Низька (адміністратор може редагувати записи).	Висока (Immutable Hashed Chain).	Гарантія цілісності доказової бази інцидентів.
5	Реакція на інцидент	Повідомлення адміністратору через певний час.	Автоматичне обмеження прав або блок сесії.	Мінімальний час між атакою та її зупинкою.

Варто звернути увагу на те, що розробники ігнорують такі прості речі, як швидкість роботи користувача. В існуючих системах, якщо адміністратор відкриє 200 карток мешканців за 5 хвилин, система навіть не зверне на це уваги, бо технічно

у нього є на це право. Такі показники активності перевищують фізично можливі межі виконання операцій користувачем. Відсутність такого елементарного логічного фільтра - це є суттєвим обмеженням функціональності системи [36]. Крім того, існуючі засоби захисту часто створюють значне навантаження на апаратні ресурси. Вони так навантажують робочі станції своїми перевірками, що адміністратори починають шукати способи їх вимкнути або обійти, що лише погіршує ситуацію, створюючи додаткові ризики для безпеки системи. Це створює конфлікт між зручністю роботи та безпекою, у якому пріоритет часто надається зручності використання [37].

Ще один нюанс - це робота з привілейованими користувачами. У стандартних системах системний адміністратор може зайти в базу через консоль і видалити будь-який запис про свої дії. Існуючі засоби захисту майже ніяк не контролюють тих, хто ці засоби налаштовує. Наша ж концепція передбачає, що навіть той, хто має найвищі права, не може бути невидимкою. Використання криптографічних ланцюжків у логах, де кожен запис прив’язаний до попереднього, робить будь-яке втручання помітним.

У результаті аналізу можна зробити висновок, що існуючі засоби захисту інформації в ЦНАП на сьогодні виконують більше формальну функцію. Вони добре підходять для проходження перевірок і отримання атестатів відповідності КСЗІ, однак їх ефективність у протидії внутрішнім загрозам або цілеспрямованим атакам залишається обмеженою. Саме цей розрив між формальною безпекою та реальною захищеністю і став поштовхом для моєї розробки. У межах дослідження не передбачається заміна існуючих систем, натомість пропонується створення інтелектуальної надбудови, яка дозволить усунути виявлені обмеження та забезпечити реалізацію проактивних механізмів захисту інформації. Це дозволить вийти на новий рівень безпеки, де система сама стає помічником, а не просто пасивним сховищем текстових журналів подій. Це допоможе автоматично відсіювати рутинний інформаційний шум і звертати увагу адміністратора безпеки лише на дійсно критичні аномалії в роботі реєстрів.

1.5 Постановка задачі розроблення системи захисту

Завершуючи аналітичний огляд першого розділу, можна зазначити, що виявлені недоліки в законодавстві, технічні вразливості та обмеження існуючих програм мають бути враховані при формуванні подальшого плану дій. При цьому методологія розробки ТЗ на створення КСЗІ має відповідати державним методичним вказівкам [38]. Основною метою розробки є створення інтелектуальної підсистеми, яка функціонуватиме між базою даних ЦНАП та користувачем, здійснюючи постійний контроль за його діями [39]. При цьому не передбачається повне переосмислення існуючої системи документообігу, оскільки це є недоцільним. Основна увага приділяється розробленню підсистеми моніторингу, здатної у автоматизованому режимі аналізувати активність користувача [40].

Ключовим є відмова від жорстких методів блокування доступу, оскільки це дозволяє зберегти стабільність роботи системи. Замість цього доцільно застосовувати підходи, які дозволяють обмежувати підозрілу активність поступово та без явного втручання в робочий процес. Це можуть бути алгоритми, що частково приховують дані або уповільнюють обробку запитів у разі виявлення нетипової поведінки. Такий підхід дає можливість своєчасно реагувати на потенційні загрози, не порушуючи звичний режим роботи системи. З метою реалізації зазначених підходів доцільно сформувати вимоги до функціональних модулів підсистеми захисту, які будуть розглянуті далі (таблиця 1.5). Також слід врахувати вимоги щодо організації служби захисту інформації в автоматизованій системі. Для систематизації технічних аспектів розробки необхідно деталізувати функціональний склад майбутньої підсистеми. Кожен із запропонованих модулів має вирішувати конкретну групу вразливостей, що були виявлені на етапі аналізу ІТ-інфраструктури установи [41]. Окрім базових функцій безпеки, до переліку включено інструменти динамічного впливу на сесію користувача, що дозволяють реалізувати концепцію адаптивного захисту. Основні вимоги до програмних модулів та очікувані результати від їх впровадження представлено в таблиці 1.5.

Таблиця 1.5 Функціональні вимоги до підсистеми захисту

№	Назва вимоги (модуля)	Технічна сутність та функціональне призначення	Пріоритет реалізації	Очікуваний результат для безпеки ЦНАП
1	Аналізатор Trust Score	Обчислення рейтингу довіри на основі часу входу, IP-адреси та історії дій.	Критичний	Динамічне управління правами доступу в межах однієї сесії.
2	Модуль Active Throttling	Алгоритм затримки пакетів даних при виявленні ознак автоматичного парсингу.	Високий	Технічне уповільнення витоку бази до швидкості, що не має сенсу для атаки.
3	Екранне маскування (Masking)	Динамічна заміна конфіденційних полів на хешовані значення (напр. XXX-XX-00).	Середній	Захист від візуального копіювання чи фотографування екрана монітора.
4	Immutable Logging Unit	Блок фіксації подій з використанням ланцюжків цілісності (Hashed Chain).	Високий	Створення доказової бази, яку неможливо підробити навіть з правами Root.
5	Контроль контексту (Context API)	Перевірка наявності відкритої справи в СЕД при запиті до персональних даних.	Критичний	Заборона «цікавості» адміністратора щодо даних осіб, які не є в черзі.

Сформований набір вимог охоплює основні виявлені ризики, але він відображає саме ті загрози, які були визначені на етапі аналізу. Особлива увага в постановці задачі приділяється питанню автономності. Підсистема має бути захищеною від несанкціонованого вимкнення або модифікації навіть з боку привілейованих користувачів установи. Це означає, що адміністратор не повинен мати можливості відключити її через стандартні засоби керування. Такий підхід створює додатковий рівень контролю саме для привілейованих дій, які часто залишаються поза увагою у звичайних системах. Також важливо, щоб система була

гнучкою у налаштуваннях. У разі зміни регламенту роботи ЦНАП або впровадження нових послуг має бути передбачена можливість коригування алгоритмів функціонування без необхідності повного перепроєктування програмного забезпечення. Це забезпечує довговічність запропонованого рішення та дозволяє адаптувати його до змін законодавства і організаційних процесів [42].

Задача полягає у переході до проактивного захисту. Необхідно створити середовище, у якому будь-яке відхилення від нормальної поведінки підлягає аналізу та відповідному реагуванню [43]. Результатом роботи має стати програмний комплекс, який не лише здійснює фіксацію подій у журналах, а й забезпечує реальний захист інформаційних ресурсів за рахунок швидкого та обґрунтованого прийняття рішень [44]. Існуючі підходи не дозволяють у повному обсязі виявляти внутрішні загрози, що зумовлює необхідність розроблення підсистеми аналізу поведінки користувача. Впровадження такого підходу дозволяє підвищити ефективність системи захисту за рахунок переходу від реакції на інциденти до їх попередження. Це, у свою чергу, формує основу для подальшого проєктування архітектури системи захисту та вибору методів її реалізації [45-46].

У наступному розділі здійснюється перехід від сформульованих вимог до побудови конкретних моделей та алгоритмів функціонування системи, що дозволяє реалізувати запропоновані підходи у вигляді цілісного програмного рішення. На цьому етапі визначені вимоги набувають більш практичного змісту та використовуються для побудови структури системи і опису її роботи. Увага приділяється визначенню логіки обробки запитів, контролю доступу та оцінки дій користувачів. Це дозволяє перейти від загального опису до конкретних механізмів, які забезпечують захист інформації під час роботи системи. Наступний розділ є логічним продовженням проведеного аналізу і спрямований на практичну реалізацію запропонованих рішень.

2. ПРОЄКТУВАННЯ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЙНИХ РЕСУРСІВ

2.1 Формування вимог до системи захисту

Вимоги до системи захисту формуються як конкретні правила її роботи під час обробки інформації. Вони визначають, які дії дозволені користувачам, як система реагує на різні ситуації та які обмеження застосовуються для запобігання порушенням. У системі передбачено роботу з кількома типами користувачів. Розподіл ролей закладається на базовому рівні, оскільки саме від нього залежить доступ до функціональних можливостей. Для кожної ролі визначається перелік дозволених дій. Наприклад, оператор працює з записами та заявами, але не має можливості змінювати системні налаштування. Адміністратор має ширші повноваження, але його дії також підлягають контролю. Доступ до функцій системи перевіряється не лише під час входу, а перед кожною дією. Це означає, що навіть при наявності активної сесії користувач не може виконати операцію, яка не відповідає його ролі. Такий підхід дозволяє уникнути виконання дій поза межами наданих прав. Вхід у систему організований таким чином, щоб зменшити ризик несанкціонованого доступу. Пароль не зберігається у відкритому вигляді, а використовується його захищене представлення. Під час перевірки система порівнює не сам пароль, а результат його обробки. Додатково враховується кількість невдалих спроб входу. У разі їх повторення доступ може бути тимчасово обмежитись. Це дозволяє ускладнити підбір паролів. Усі дії користувачів підлягають обов’язковій фіксації. Записується інформація про те, хто виконав дію, коли вона була виконана і до яких даних відбувався доступ. Це стосується як перегляду, так і зміни або видалення інформації. Наявність таких записів дозволяє відновити послідовність подій у разі виникнення проблемних ситуацій.

Система повинна забезпечувати можливість збереження попередніх станів даних. Це дозволяє відновити інформацію у випадку помилкових або небажаних змін. Така функція є важливою не лише для захисту від помилок користувачів, але і для протидії навмисному втручанню. Резервне копіювання розглядається як обов’язковий елемент роботи системи. Копії створюються регулярно та

зберігаються окремо від основної бази даних. Важливо, щоб процес відновлення був швидким і не призводив до втрати інформації. Окрім фіксації дій, система повинна реагувати на нетипову поведінку користувачів. Наприклад, різке збільшення кількості запитів або часті звернення до різних записів за короткий час можуть розглядатися як відхилення від нормальної роботи. У таких випадках система не обов'язково одразу блокує користувача, але виділяє події для подальшого аналізу. Для спрощення оцінки поведінки використовується показник довіри. Його значення змінюється залежно від дій користувача. Якщо робота відбувається без відхилень, показник залишається стабільним. У разі появи підозрілих дій він знижується і може бути використаний як підстава для додаткової перевірки.

Для оцінювання рівня довіри до поточної сесії користувача в системі використовується механізм динамічного розрахунку показника Trust Score. Значення показника змінюється залежно від характеру та інтенсивності виконуваних дій користувача. Формалізовано процес зміни рівня довіри можна подати таким співвідношенням:

$$TS = TS_{prev} - \sum_{i=1}^{n} w_i \cdot r_i \tag{2.1}$$

де TS_{prev} - попереднє значення показника довіри;

w_i - ваговий коефіцієнт критичності і-го порушення;

r_i - параметр, що характеризує факт або інтенсивність ризикової дії користувача.

Запропонований підхід дозволяє враховувати сукупність поведінкових ознак користувача та виконувати адаптивне оцінювання рівня безпеки поточної сесії. У разі виявлення нетипової активності значення показника Trust Score поступово зменшується, що дозволяє системі своєчасно реагувати на потенційні загрози. Такий механізм забезпечує можливість автоматичного контролю дій користувача без необхідності постійного втручання адміністратора системи. Прийняття рішення

стосовно обмеження доступу відбувається на основі порівняння поточного значення показника довіри з критичним порогом:

$$\text{if } TS < T_{crit} \Rightarrow \text{block} \tag{2.2}$$

де T_{crit} - критичне порогове значення рівня довіри, при досягненні якого система автоматично ініціює процедуру блокування або обмеження поточної сесії користувача.

Такий підхід дозволяє системі реагувати не тільки на вже виконані порушення, а й на підозрілі дії користувача під час роботи. Якщо система фіксує нетипову активність, то рівень довіри поступово знижується, а доступ до окремих функцій може бути обмежений. Це дає змогу своєчасно виявляти потенційно небезпечні ситуації та зменшувати ризик витоку або неправомірного використання інформації.

Доступ до критичних операцій обмежується навіть для авторизованих користувачів. Виконання таких дій, як зміна або видалення важливих даних, може вимагати додаткового підтвердження. Це дозволяє зменшити ризик як випадкових помилок, так і навмисного втручання. Передача даних між компонентами системи здійснюється у захищеному вигляді. Це стосується як взаємодії між клієнтом і сервером, так і внутрішніх процесів. Облікові дані не передаються і не зберігаються у відкритому вигляді. Інтерфейс системи не повинен ускладнювати роботу користувача. Надмірна складність призводить до помилок, тому логіка взаємодії має залишатися зрозумілою. Захисні механізми реалізуються таким чином, щоб не заважати виконанню основних задач. Система повинна бути гнучкою та придатною до подальшого розвитку. Передбачається можливість додавання нових ролей, функцій і правил доступу без значних змін у структурі. Це дозволяє адаптувати систему до змін у вимогах та умовах роботи. Забезпечує довгострокову стійкість архітектури та спрощує масштабування системи в майбутньому.

2.2 Розроблення моделі загроз та порушника

Після формування вимог до системи необхідно визначити загрози, які можуть впливати на її роботу. Для цього аналізуються реальні ситуації, що можуть виникати під час експлуатації інформаційної системи ЦНАП. Працівники протягом дня взаємодіють із даними: переглядають записи, здійснюють пошук та вносять зміни. У більшості випадків ці дії є нормальними і не викликають підозри. Проте при зміні їх інтенсивності або характеру можуть виникати ризики. Частина з них формується поступово. Окремі дії самі по собі виглядають звичайними, але їх повторення з часом створює небезпеку. Наприклад, користувач може працювати з великою кількістю записів. Якщо це відбувається одноразово, це не є проблемою. Але регулярне виконання таких дій може свідчити про спробу збору інформації. Схожа ситуація виникає при копіюванні даних. Якщо інформація копіюється великим обсягом, це легко помітити. Якщо ж копіювання відбувається невеликими частинами протягом тривалого часу, система може не одразу зафіксувати відхилення. Крім поведінкових загроз, необхідно враховувати і технічні. Вони пов’язані зі спробами отримати несанкціонований доступ або вплинути на роботу системи. Такі ситуації виникають рідше, але можуть мати значні наслідки. Для систематизації загроз доцільно подати їх у структурованому вигляді (таблиця 2.1).

Таблиця 2.1 - Модель загроз інформаційній системі ЦНАП

№	Загроза	Тип впливу	Джерело	Опис реалізації	Ознаки
1	2	3	4	5	6
1	Несанкціонований доступ	Доступ	Зовнішній	Підбір пароля	Часті невдалі входи
2	Використання чужого акаунта	Доступ	Внутрішній	Робота під іншим користувачем	Нетипова активність
3	Перевищення прав доступу	Доступ	Внутрішній	Доступ до функцій поза роллю	Запити до закритих даних
4	Масовий перегляд записів	Доступ	Внутрішній	Велика кількість переглядів	Висока інтенсивність

Продовження таблиці 2.1

1	2	3	4	5	6
5	Часті запити до системи	Доступ	Внутрішній	Постійне звернення	Короткі інтервали
6	Нетипові вибірки	Доступ	Внутрішній	Незвичні запити	Нестандартні параметри
7	Зміна даних	Цілісність	Внутрішній	Редагування записів	Часті зміни
8	Помилкова зміна	Цілісність	Внутрішній	Ненавмисне редагування	Некоректні значення
9	Видалення записів	Цілісність	Внутрішній	Видалення інформації	Зникнення даних
10	Масове видалення	Цілісність	Адміністратор	Групове видалення	Різке зменшення
11	Копіювання даних	Конфіденційність	Внутрішній	Винесення інформації	Часті звернення
12	Поступовий витік	Конфіденційність	Внутрішній	Копіювання частинами	Регулярність
13	Формування великих вибірок	Конфіденційність	Внутрішній	Збір масиву даних	Нетиповий обсяг
14	Брутфорс	Доступ	Зовнішній	Підбір пароля	Серія спроб
15	SQL-ін’єкція	Доступ	Зовнішній	Вставка шкідливих запитів	Помилки БД
16	Перехоплення даних	Конфіденційність	Зовнішній	Незахищена передача	Підозрілий трафік
17	Підміна інформації	Цілісність	Зовнішній	Зміна даних	Некоректні записи
18	DDoS-атака	Доступність	Зовнішній	Перевантаження системи	Затримки
19	Збій системи	Доступність	Внутрішній	Відмова обладнання	Недоступність
20	Втрата резервних копій	Доступність	Внутрішній	Відсутність backup	Неможливість відновлення
21	Зловживання правами	Усі	Адміністратор	Повний доступ	Критичні зміни
22	Відсутність логування	Усі	Внутрішній	Дії без запису	Відсутність слідів
23	Маніпуляція журналом	Цілісність	Адміністратор	Видалення логів	Втрата історії
24	Нетиповий час роботи	Усі	Внутрішній	Робота поза графіком	Нічна активність
25	Доступ з нового місця	Доступ	Внутрішній	Інша локація	Зміна IP

Кінець таблиці 2.1

1	2	3	4	5	6
26	Повторювані дії	Доступ	Внутрішній	Шаблонні запити	Однакові операції
27	Помилки програмного забезпечення	Усі	Система	Некоректна робота	Збої
28	Людський фактор	Усі	Внутрішній	Неуважність	Помилки
29	Витік через носії	Конфіденційність	Внутрішній	Копіювання на пристрій	Відсутність контролю
30	Комбіновані загрози	Усі	Будь-який	Поєднання дій	Складні сценарії

Проведений аналіз показує, що значна частина загроз пов’язана не з прямим порушенням правил, а зі зміною характеру звичайної роботи. Одна і та сама дія може бути як нормальною, так і небезпечною залежно від умов виконання. Наприклад, перегляд записів є базовою функцією системи. Однак надмірна інтенсивність або виконання таких дій у нетиповий час може свідчити про відхилення. Тому оцінювати необхідно не лише сам факт дії, а й контекст її виконання. Увагу слід приділити загрозам накопичувального характеру. Вони не проявляються одразу і можуть тривалий час залишатися непоміченими. Саме такі ситуації становлять підвищений ризик, оскільки їх складніше виявити стандартними засобами контролю. Варто враховувати можливість комбінування загроз. У реальних умовах вони рідко виникають ізольовано. Частіше спостерігається послідовність дій, яка поступово призводить до порушення безпеки.

Після визначення загроз необхідно встановити, хто саме може їх реалізувати. Для цього формується модель порушника, яка описує можливі типи користувачів, їх доступ, рівень знань та характер дій. Зовнішній порушник не має прямого доступу до системи, тому його основна мета - отримати цей доступ. Для цього використовуються технічні засоби, зокрема автоматизовані спроби підбору паролів або атаки на систему. Такі дії зазвичай відрізняються від звичайної роботи і можуть бути виявлені. Внутрішній користувач перебуває в іншій ситуації. Він уже має доступ до системи та добре орієнтується у її структурі. Йому не потрібно долати

захист, тому саме це робить його більш небезпечним у певних випадках. Його дії можуть виглядати як звичайна робота, що ускладнює їх виявлення. Адміністратор має повний доступ до системи, що значно розширює його можливості. Це створює окремий рівень ризику, оскільки навіть одна помилка або навмисна дія може призвести до серйозних наслідків. Потрібно врахувати ситуації, коли обліковий запис є скомпрометованим. У такому випадку система не може відрізнити дії реального користувача від дій сторонньої особи. Це ускладнює процес виявлення загроз, тому наступним кроком є формування моделі порушника (таблиця 2.2).

Таблиця 2.2 - Модель порушника

№	Тип порушника	Рівень доступу	Знання системи	Поведінка	Можливі дії	Рівень ризику
1	Зовнішній випадковий	Відсутній	Низький	Непослідовна	Спроби входу	Низький
2	Зовнішній підготовлений	Відсутній	Середній	Цілеспрямована	Атаки	Середній
3	Хакер	Відсутній	Високий	Системна	Злам	Високий
4	Оператор	Обмежений	Високий	Робоча	Перегляд даних	Середній
5	Недобросовісний оператор	Обмежений	Високий	Цілеспрямована	Копіювання	Високий
6	Досвідчений користувач	Обмежений	Високий	Аналізуюча	Обхід обмежень	Дуже високий
7	Адміністратор	Повний	Високий	Контрольна	Усі операції	Критичний
8	Скомпрометований акаунт	Будь-який	Невідомий	Непередбачувана	Будь-які дії	Високий

Аналіз моделі порушника показує, що найбільшу небезпеку становлять внутрішні загрози. Вони менш помітні, можуть тривалий час залишатися без уваги та не викликати підозри на початковому етапі. Особливість таких загроз полягає в тому, що вони маскуються під звичайну роботу. Саме тому їх виявлення потребує аналізу не окремих дій, а їх сукупності та динаміки. Побудована модель загроз і порушника дозволяє визначити основні ризики, які необхідно врахувати під час проєктування системи захисту інформаційних ресурсів.

2.3 Розроблення архітектури системи захисту

У процесі розроблення системи захисту інформаційних ресурсів одним із ключових етапів є формування її архітектури. Саме архітектурне рішення визначає, яким чином здійснюється обробка інформації, контроль доступу, взаємодія між компонентами системи та реагування на дії користувачів. У межах даної роботи архітектура розглядається не лише як технічна структура, а як основа для реалізації механізмів захисту, що функціонують у реальному середовищі. Взаємодія користувача із системою починається з веб-інтерфейсу, який реалізовано на базі фреймворку Flask. Користувач вводить свої облікові дані у форму авторизації, після чого ці дані перетворюються у HTTP-запит і передаються на сервер. При цьому важливо відзначити, що сам інтерфейс не виконує жодних функцій, пов’язаних із прийняттям рішень щодо доступу або обробки інформації. Його основне призначення полягає у передачі даних на серверну частину системи, де зосереджена вся критична логіка.

Після надходження запиту на сервер розпочинається етап перевірки користувача. Для цього використовуються дані, що зберігаються у базі даних, але не у відкритому вигляді. Паролі представлені у вигляді хешованих значень, що виключає можливість їх безпосереднього використання у разі несанкціонованого доступу до бази. Під час автентифікації система обчислює хеш введеного пароля та порівнює його з наявним значенням. У разі невідповідності доступ до системи не надається. Додатково враховується кількість невдалих спроб входу. Якщо таких спроб стає надмірна кількість, система автоматично реагує шляхом тимчасового обмеження можливості авторизації. Такий підхід дозволяє протидіяти типовим атакам підбору паролів та підвищує загальний рівень захисту системи без ускладнення її використання для легітимних користувачів.

Після успішної автентифікації користувача процес контролю не припиняється. Навпаки, кожна подальша дія підлягає перевірці. При надходженні нового запиту система ідентифікує користувача та визначає, чи має він право виконувати відповідну операцію. Це реалізовано безпосередньо у логіці серверних

маршрутів, що дозволяє здійснювати контроль ще до виконання будь-яких дій. Такий підхід забезпечує чітке розмежування доступу та унеможливлює виконання операцій поза встановленими правилами. Кожна дія користувача фіксується у журналі подій. Записи містять інформацію про користувача, тип виконаної операції, час її здійснення, а також додаткові параметри, що можуть бути використані для аналізу. Ці дані зберігаються окремо від основної інформації, що дозволяє забезпечити їх цілісність та використання для подальшого аналізу. Журнал подій використовується не лише для відновлення історії дій, але й як основа для аналізу поведінки користувачів. Система враховує не окремі дії, а їх сукупність. Аналізується кількість запитів, частота звернення до даних, характер виконуваних операцій. У випадку виявлення різких змін у поведінці або нетипових дій система ідентифікує такі ситуації як потенційно ризикові. Особлива увага приділяється контексту виконання дій. Наприклад, авторизація у нетиповий час або використання нової IP-адреси не завжди є ознакою порушення, однак у поєднанні з іншими факторами може свідчити про підозрілу активність. Система фіксує такі параметри та враховує їх при подальшій оцінці поведінки користувача.

На основі зібраної інформації формується показник довіри до користувача. Він не є статичним і змінюється у процесі роботи. При нормальній поведінці значення показника залишається стабільним. У випадку відхилень він поступово знижується. Такий підхід дозволяє виявляти потенційні загрози ще до їх переходу у критичну фазу. При досягненні певного порогового значення система застосовує відповідні заходи реагування. Це може бути обмеження доступу до окремих функцій, запит повторної автентифікації або повне блокування облікового запису. Важливо, що такі дії виконуються автоматично, без необхідності втручання адміністратора, що забезпечує оперативність реагування. Взаємодія з базою даних здійснюється виключно через сервер. Користувач не має прямого доступу до даних, що виключає можливість їх несанкціонованого використання. Усі запити проходять централізовану обробку, що дозволяє контролювати кожну операцію. Варто звернути увагу на те, що описані механізми працюють не ізольовано, а у взаємозв’язку між собою. Наприклад, перевірка доступу та аналіз поведінки

виконуються одночасно і доповнюють один одного. Якщо доступ до дії формально дозволений, але сама поведінка користувача виглядає нетипово, система може врахувати це при прийнятті рішення. Контроль здійснюється не лише на рівні правил, а й на рівні фактичної роботи користувача.

Важливо, що система не обмежується одноразовою перевіркою. Контроль продовжується протягом усього часу роботи користувача із системою. Кожна наступна дія може впливати на загальну оцінку поведінки, що дозволяє виявляти поступові зміни, які складно помітити одразу. Це особливо актуально для ситуацій, коли порушення відбувається не різко, а частинами. Такий підхід дозволяє зменшити кількість помилкових спрацювань. Якщо система реагувала б лише на окремі дії, це могло б призводити до необґрунтованих обмежень. У даному випадку рішення приймається з урахуванням загальної картини, що робить реакцію більш обґрунтованою. У підсумку система функціонує як єдина узгоджена структура, у якій перевірка доступу, фіксація дій та аналіз поведінки виконуються одночасно під час обробки кожного запиту. Це дозволяє виявляти не лише явні порушення, але й приховані загрози, які формуються поступово. Схема, наведена на рисунку, відображає загальну організацію інформаційної системи та взаємодію її основних компонентів (рисунок 2.1).

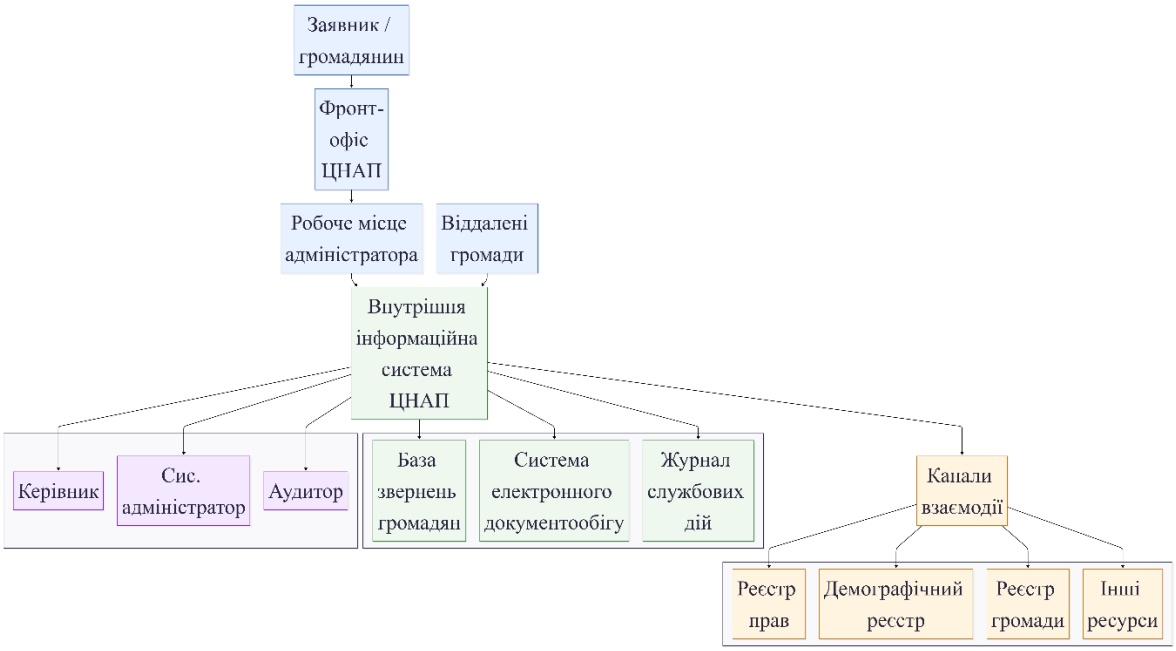


Рисунок 2.1 - Структурна схема інформаційної системи ЦНАП

Вона дає можливість зрозуміти не лише структуру системи, а й те, як саме відбувається обробка запитів, передача даних і виконання перевірок. За рахунок цього можна чітко побачити, на якому етапі і як реалізується контроль доступу та інші механізми захисту. На початковому рівні знаходиться користувач, який працює через відповідний інтерфейс. Саме через нього здійснюється вся взаємодія із системою. Користувач не має прямого доступу до внутрішніх компонентів, а всі його дії проходять через визначений канал. Це дозволяє контролювати процес ще на етапі формування запиту і не допускати виконання операцій поза межами системи. Після формування запиту він передається на сервер, який виступає центральним елементом системи. Саме тут зосереджена вся основна логіка. На серверному рівні відбувається перевірка облікових даних користувача, визначення його прав доступу та прийняття рішення щодо виконання запиту. Такий підхід дозволяє уникнути ситуацій, коли частина перевірок виконується окремо або не виконується взагалі. Централізація обробки має ще одну перевагу - всі правила застосовуються однаково. Це означає, що незалежно від того, хто саме працює із системою, перевірки виконуються за єдиною логікою. За рахунок цього зменшується ймовірність помилок і спрощується контроль.

Окрему роль у структурі відіграє база даних. Вона використовується не лише для зберігання основної інформації, але й для накопичення службових даних. Зокрема, у ній зберігаються журнали подій, які містять інформацію про всі виконані дії. Це дозволяє не просто зберігати результати роботи, а відстежувати, що саме відбувалося в системі. За рахунок цього з’являється можливість аналізувати поведінку користувачів. Наприклад, можна визначити, як часто виконується певна дія, чи змінюється характер роботи, чи не з’являються нетипові операції. Такий аналіз дає більше інформації, ніж просто перевірка прав доступу. Механізми захисту у даній архітектурі не винесені в окремий блок. Вони працюють разом із основною логікою обробки запитів. Це означає, що перевірка доступу, фіксація дій і оцінка поведінки виконуються одночасно з самою операцією. За рахунок цього система може реагувати одразу, а не після того, як дія вже виконана.

Кожен запит проходить кілька послідовних етапів. Спочатку визначається

користувач, далі перевіряються його права доступу, після цього оцінюється сама дія. Якщо все відповідає нормальній роботі, запит виконується. Якщо ж з'являються відхилення, система може або обмежити дію, або зафіксувати її для подальшого аналізу. При такому підході система контролює не тільки сам факт виконання операції, а й те, як вона виконується. Це дозволяє помічати ситуації, які не виглядають як порушення окремо, але у сукупності можуть створювати ризик. Ще один момент – система працює не "разово", а постійно. Дані, які накопичуються у журналі подій, використовуються далі. За рахунок цього система поступово "розуміє", що є нормальною поведінкою, а що - відхиленням. Це дозволяє підвищити точність виявлення підозрілих ситуацій. Інтеграція механізмів захисту у процес обробки запитів дозволяє уникнути ситуацій, коли потенційно небезпечна дія вже виконана і лише після цього аналізується. У даному випадку система має можливість впливати на процес виконання ще до завершення операції, що значно підвищує ефективність захисту. Загалом така побудова відповідає централізованій моделі, у якій усі ключові процеси контролюються на серверному рівні. Це спрощує керування системою, забезпечує узгодженість роботи її компонентів та дозволяє швидко реагувати на підозрілу активність. Централізація також забезпечує однакові правила обробки запитів для всіх користувачів. Це особливо важливо у системах, де обробляється персональна інформація, оскільки будь-які відхилення у правилах доступу можуть призвести до порушення безпеки.

У підсумку така архітектура дозволяє поєднати контроль доступу, фіксацію дій і аналіз поведінки в межах єдиної системи. Це означає, що система не лише обмежує доступ до інформації, а й контролює, як саме вона використовується у процесі роботи. Кожна дія користувача розглядається, як частина загальної картини, що дозволяє виявляти нетипові. За рахунок цього підвищується не тільки рівень захисту, а й загальна керованість системи. Усі основні механізми працюють узгоджено, що зменшує ймовірність помилок або неконтрольованих дій. Це дозволяє своєчасно реагувати на підозрілу активність та запобігати розвитку потенційних загроз ще на ранніх етапах. Структурна схема системи захисту інформації наведена в додатку Б.

Побудова системи захисту відображає, яким чином реалізуються основні механізми контролю в межах системи. На відміну від загальної архітектури, тут акцент зроблено саме на тих елементах, які відповідають за безпеку. Усі перевірки виконуються під час обробки запитів, що дозволяє контролювати дії користувача не після їх завершення, а безпосередньо в процесі роботи. Система не обмежується лише перевіркою прав доступу. Додатково враховується поведінка користувача, що дозволяє оцінювати не тільки сам факт виконання дії, а й те, як вона виконується. За рахунок цього з'являється можливість виявляти ситуації, які не виглядають як порушення окремо, але у сукупності можуть становити ризик. Усі дії проходять через єдину точку контролю, що забезпечує узгодженість перевірок і виключає можливість обходу системи. При цьому результати кожної операції фіксуються, що дозволяє відслідковувати подальший розвиток подій та використовувати ці дані для аналізу. Це дозволяє забезпечити не лише обмеження доступу до інформації, а й контроль за її використанням. У результаті система реагує не тільки на вже здійснені порушення, а й на потенційні загрози, які формуються поступово.

Показана структура програмного проєкту відображає фактичну організацію системи та розподіл функцій між її основними компонентами. У даному випадку вона не є формальною схемою, а безпосередньо відповідає тому, як система працює у процесі виконання запитів. Кожен елемент структури використовується на конкретному етапі обробки даних і виконує визначену роль у забезпеченні контролю та захисту інформації. Центральним компонентом системи є файл app.py, у якому реалізовано серверну логіку. Саме через нього проходить повний цикл обробки запитів користувачів. При зверненні до системи запит не передається одразу до бази даних або іншого елемента, а спочатку потрапляє у відповідний маршрут, де виконується перевірка авторизації, статусу користувача та його прав доступу. На цьому етапі система визначає, чи може користувач виконувати запитувану дію. Якщо перевірка не пройдена, подальша обробка не виконується, що дозволяє обмежити доступ ще до взаємодії з даними.

Після первинної перевірки система переходить до аналізу самої дії. Визначається її тип - перегляд, зміна, видалення або інша операція. Для кожного

типу встановлено окремі правила доступу. Наприклад, перегляд інформації може бути дозволений ширшому колу користувачів, тоді як редагування або видалення потребують додаткових умов. Такий підхід дозволяє реалізувати більш точний контроль без введення надмірних обмежень. На наступному етапі враховується контекст виконання дії. Система аналізує не лише сам запит, а й умови, за яких він виконується. До уваги береться частота звернень, послідовність дій, час роботи та попередня активність користувача. Якщо поточна дія суттєво відрізняється від звичайної поведінки, це впливає на оцінку ризику та рівень довіри. Такий механізм дозволяє виявляти ситуації, які не можуть бути визначені лише на основі прав доступу.

Після завершення перевірок система приймає рішення щодо виконання запиту. У стандартній ситуації дія виконується без обмежень. Якщо ж виявлено відхилення, можливі різні варіанти реагування: дія може бути повністю заблокована, обмежена або виконана із додатковою фіксацією як підозріла. Важливо, що рішення формується на основі сукупності факторів, а не одного параметра. Ключовим елементом системи є фіксація подій. Після виконання або навіть спроби виконання дії відповідна інформація записується у журнал. Усе це відбувається автоматично і не залежить від користувача. У записі зберігаються дані про тип операції, час виконання, користувача та рівень ризику. Це дозволяє формувати повну історію роботи системи та використовувати її для подальшого аналізу. База даних у даній системі виконує не лише функцію зберігання інформації. Вона використовується як джерело для прийняття рішень. Через неї визначається рівень довіри користувача, його статус, а ще накопичуються дані для аналізу поведінки. Журнал подій і таблиця загроз дозволяють відокремити звичайні операції від тих, які потребують. База даних виступає активним елементом системи, який впливає на її роботу.

Папка templates використовується на етапі формування відповіді користувачу. Вона містить HTML-шаблони, які відображають різні сторінки системи. Кожен шаблон пов’язаний із конкретною функцією, наприклад авторизацією, переглядом записів або аналізом журналу подій. При цьому шаблони не виконують перевірок і

не містять логіки прийняття рішень. Вони лише відображають вже оброблені дані. Це означає, що користувач бачить тільки ті результати, які дозволені системою.

Папка static відповідає за оформлення інтерфейсу. У ній зберігаються файли стилів, які визначають вигляд сторінок. Хоча ці файли не беруть участі у виконанні перевірок, вони впливають на зручність роботи. Зрозумілий інтерфейс зменшує кількість помилок користувачів, що, у свою чергу, знижує ризики, пов’язані з людським фактором.

Папка backups використовується для збереження резервних копій бази даних. Копії створюються під час роботи системи і містять інформацію про стан даних на певний момент часу. У випадку пошкодження або втрати даних система може бути відновлена до попереднього стану. Це забезпечує стабільність роботи і дозволяє уникнути критичних наслідків.

Важливим є те, що всі компоненти системи працюють як єдине ціле. Запит користувача проходить послідовний шлях: від інтерфейсу до серверної логіки, далі до бази даних і назад до відображення результату. На кожному етапі виконуються відповідні перевірки, що дозволяє забезпечити контроль за діями користувача. Важливо, що система не обмежується виконанням окремих перевірок, а накопичує інформацію про роботу користувачів. Журнал подій зберігає історію, яка використовується для виявлення довготривалих змін у поведінці. Це дозволяє виявляти загрози, які не проявляються одразу, а формуються поступово. У системі також передбачено обробку помилок. У випадку збою виконання окремої операції система не припиняє роботу повністю. Замість цього вона обмежує виконання конкретної дії і продовжує обробку інших запитів. Це дозволяє зберігати працездатність навіть у нестандартних ситуаціях.

Структура програмного проєкту визначає не лише розташування файлів, а й логіку функціонування всієї системи. Кожен компонент використовується на конкретному етапі обробки запиту, а їх взаємодія забезпечує перевірку, фіксацію та аналіз дій користувача. Саме за рахунок цього система здатна не лише виконувати операції, а й контролювати процес роботи з інформацією та забезпечувати її захист у реальних умовах експлуатації. У результаті система працює стабільніше та краще

контролює можливі загрози.

2.4 Вибір методів і засобів забезпечення конфіденційності, цілісності та доступності інформації

Після того як визначено, як саме побудована система, необхідно детальніше зрозуміти, за рахунок яких механізмів у ній забезпечується захист інформації. У даному випадку захист не реалізується одним окремим засобом, а формується за рахунок декількох взаємопов'язаних рішень. Почати доцільно з конфіденційності. У системі вона реалізується на кількох рівнях одночасно. Найперше - це робота з паролями. Вони не зберігаються у відкритому вигляді, а перетворюються у спеціальне значення ще на етапі введення. Тобто користувач вводить пароль, після чого система одразу його обробляє і зберігає вже у зміненому вигляді. За рахунок цього навіть у випадку доступу до бази даних отримані значення не дають можливості дізнатися реальні паролі. Під час входу виконується така сама операція. Система не порівнює введений пароль безпосередньо, а спочатку перетворює його, після чого співставляє з тим значенням, яке вже збережене. Якщо ці значення не співпадають, доступ не надається. Такий підхід дозволяє уникнути зберігання чутливої інформації у відкритому вигляді та значно підвищує рівень захисту. Враховується поведінка користувача під час авторизації. Якщо він кілька разів вводить неправильний пароль, система це фіксує. У певний момент подальші спроби просто блокуються. Це виглядає як проста перевірка, але на практиці значно ускладнює підбір пароля та знижує ефективність подібних атак.

Ще один важливий аспект - це доступ до даних. Користувач не працює з базою напряму і не має можливості взаємодіяти з нею безпосередньо. Усі дії проходять через сервер, який виконує роль посередника. Саме на цьому рівні визначається, що дозволено робити, а що ні. За рахунок цього дані не передаються користувачу без перевірки, а будь-яка операція контролюється. Детальніше ще продумано роботу з документами. Перегляд вважається стандартною дією, тому він

дозволяється. Однак скачування - це вже інша ситуація, оскільки саме на цьому етапі найчастіше відбувається витік інформації. Тому така дія обмежується. Якщо користувач намагається завантажити документ, система не виконує запит і додатково фіксує цю спробу. Це дозволяє не тільки запобігти витоку, а й відслідковувати подібні ситуації.

Цілісність інформації пов’язана з тим, щоб зміни не відбувалися непомітно. У системі це реалізовано через журнал подій. Кожна дія записується із зазначенням користувача, часу та характеру операції. Це означає, що жодна зміна не залишається без фіксації. У разі необхідності можна переглянути, що саме було змінено і ким. Передбачено збереження попередніх станів даних, якщо інформацію було змінено або видалено випадково, є можливість повернутися до попередньої версії. На перший погляд це виглядає як допоміжна функція, але на практиці вона дозволяє уникнути серйозних проблем. Також окремі дії не виконуються одразу. Наприклад, зміна або видалення важливих даних може вимагати підтвердження. Це дозволяє зменшити кількість випадкових помилок і забезпечити додатковий рівень контролю. Доступність системи залежить від її стабільної роботи. У даному випадку використовується резервне копіювання. Створюються копії бази даних, які можна використати у разі збою, якщо виникає проблема, систему можна відновити без втрати інформації. У системі враховується не лише сам факт виконання дій, а й їх характер. Наприклад, аналізується частота запитів, обсяг оброблюваних даних, час роботи. Якщо ці показники змінюються, це може свідчити про відхилення. На основі цього формується показник довіри. Він змінюється поступово і залежить від дій користувача. Якщо робота виглядає нормально - значення не змінюється. Якщо з’являються нетипові дії - воно знижується. Це дозволяє виявити проблему ще до того, як вона стане очевидною. Для узагальнення описаної логіки роботи системи доцільно представити послідовність обробки дій користувача під час взаємодії із системою. Така послідовність дозволяє чітко побачити, як саме виконуються перевірки і як система реагує на різні ситуації. Алгоритм функціонування системи захисту інформації наведено в додатках В та Г

Побудований алгоритм відображає послідовність дій системи під час обробки

запитів користувача та дозволяє більш детально зрозуміти, як саме реалізуються механізми захисту на практиці. Його особливістю є те, що всі перевірки виконуються поетапно, починаючи з автентифікації користувача і завершуючи прийняттям рішення щодо виконання або обмеження дії. При цьому кожен наступний крок залежить від результатів попереднього, що забезпечує логічну цілісність роботи системи. Окрему увагу в алгоритмі приділено аналізу поведінки користувача. На відміну від традиційних підходів, де контроль завершується після перевірки прав доступу, у даному випадку система продовжує оцінювати дії навіть після успішної авторизації. Це дозволяє виявляти ситуації, коли формально користувач має доступ, але його дії відрізняються від звичайної роботи. Саме на цьому етапі формується показник рівня довіри, який впливає на подальші рішення системи.

Ще однією важливою особливістю є те, що алгоритм не передбачає миттєвого блокування при одиничному відхиленні. Рішення приймається на основі накопиченої інформації, що дозволяє зменшити кількість помилкових спрацювань. У результаті система реагує більш гнучко, поступово змінюючи рівень доступу залежно від поведінки користувача. Алгоритм враховує необхідність забезпечення стабільності роботи системи. У разі виникнення помилок або перевищення допустимого навантаження передбачені механізми обмеження дій, які дозволяють зберегти працездатність системи без її повного зупинення. Це є важливим з точки зору забезпечення доступності інформаційних ресурсів.

Запропонований алгоритм поєднує у собі контроль доступу, фіксацію подій та аналіз поведінки користувачів у межах єдиного процесу. Такий підхід дозволяє забезпечити не лише захист від явних порушень, а й своєчасне виявлення прихованих загроз, що формуються поступово в процесі роботи системи. Система при цьому більш точно реагує на підозрілу активність та краще контролює можливі ризики. Алгоритм не блокує роботу через дрібниці, а реагує лише на дійсно небезпечні аномалії в системі.

2.5 Висновок до розділу 2

У процесі аналізу стає зрозуміло, що основні ризики пов’язані не лише з очевидними атаками, а з повсякденною діяльністю. Багато дій, які самі по собі є нормальними, можуть набувати іншого значення залежно від частоти, обсягу або часу виконання. Наприклад, перегляд записів не викликає підозри, однак його повторення у великій кількості або у нетиповий час може свідчити про потенційний ризик. Окремі загрози формуються поступово, що ускладнює їх виявлення. При розгляді порушника основна увага приділяється внутрішнім користувачам, оскільки саме вони мають доступ до системи та розуміють принципи її роботи. Їх дії можуть виглядати як звичайна робота, навіть якщо фактично містять ризик. Аналогічна ситуація виникає у разі використання облікового запису сторонньою особою. З урахуванням цього архітектура системи побудована таким чином, що всі дії виконуються через сервер, де здійснюється перевірка доступу та контроль операцій. Це виключає можливість обходу системи і забезпечує централізований контроль. При цьому перевірка не обмежується лише етапом входу - система постійно аналізує дії користувача, а всі операції фіксуються.

Методи захисту функціонують у взаємозв’язку і не використовуються окремо. Контроль доступу, журнал подій і резервне копіювання доповнюють один одного, забезпечуючи комплексний підхід до захисту інформації. Окрему роль відіграє аналіз поведінки користувачів, який враховує не лише сам факт виконання дій, а й їх контекст. Це дозволяє виявляти відхилення від звичайної роботи та реагувати ще до виникнення явного порушення. У результаті система захисту не обмежується набором правил, а фактично супроводжує роботу користувача, одночасно перевіряючи, фіксуючи та аналізуючи його дії.

Сформована структура та логіка функціонування системи створюють основу для подальшої реалізації програмного рішення, що забезпечує контроль доступу, аналіз поведінки користувачів і своєчасне реагування на потенційні загрози.

3. РЕАЛІЗАЦІЯ ТА ОЦІНКА ДОСТОВІРНОСТІ СИСТЕМИ ЗАХИСТУ

3.1 Обґрунтування вибору програмних і апаратних засобів

Вибір засобів для реалізації системи захисту здійснювався з урахуванням реальних умов її використання, а не лише теоретичних можливостей технологій. У більшості випадків інформаційні системи ЦНАП працюють на стандартних робочих станціях без складної серверної інфраструктури. Саме тому при розробленні було прийнято рішення орієнтуватися на прості рішення, які можна розгорнути без додаткових витрат часу та ресурсів. Програмна частина системи реалізована мовою Python:

```
def change_trust(user_id, delta):
    user = conn.execute("SELECT trust_score FROM users WHERE id = ?",
(user_id,)).fetchone()
    score = max(0, min(100, user["trust_score"] + delta))
```

Наведений фрагмент демонструє зміну рівня довіри користувача залежно від виконаних дій. Мова Python дозволяє досить швидко створювати прикладні рішення і при цьому зберігати контроль над логікою роботи. Це особливо важливо для системи захисту, оскільки кожна дія користувача повинна не просто виконуватися, а проходити через перевірку та фіксацію:

```
if score <= 25:
    conn.execute("UPDATE users SET status = 'blocked' WHERE id = ?",
(user_id,))
```

У разі досягнення критичного значення система автоматично блокує користувача. Використання Python дало можливість реалізувати ці механізми без ускладнення структури коду. Для побудови веб-частини використано фреймворк Flask. На відміну від більш складних рішень, він не нав’язує жорсткої архітектури і дозволяє самостійно визначати структуру системи. Це стало важливим при реалізації контролю доступу та аудиту, оскільки логіка перевірок інтегрується безпосередньо у обробку запитів. У результаті кожна дія користувача проходить через єдиний механізм контролю, що спрощує відстеження подій.

Зберігання даних організовано за допомогою SQLite. На перший погляд це

виглядає як спрощене рішення, однак у межах поставленого завдання воно повністю виправдане. База даних використовується не лише для зберігання основної інформації, а й для ведення журналів аудиту. Для реалізації серверної частини системи використано фреймворк Flask, який забезпечує обробку HTTP-запитів та взаємодію з базою даних:

```
app = Flask(__name__)
app.secret_key = "cnap-security-local-key"
DB_NAME = "security_system.db"
```

У системі реалізовано окремі таблиці для користувачів, записів громадян, службових документів, подій безпеки та повідомлень про загрози. Така структура дозволяє чітко розділити різні типи даних і забезпечити контроль за їх обробкою. У системі важливу роль відіграє таблиця журналу подій. Саме вона використовується для фіксації всіх дій користувачів:

```
conn.execute("""
INSERT INTO audit_log (user_id, username, action, risk_score,
description, created_at)
VALUES (?, ?, ?, ?, ?, ?)
""")
```

Кожна дія користувача записується із зазначенням рівня ризику. Кожен запис містить інформацію про тип дії, об’єкт, час виконання та оцінку ризику. Це дає змогу не лише зберігати історію, а й аналізувати поведінку користувачів у динаміці. База даних виконує не тільки функцію зберігання, а й стає частиною механізму безпеки. Наведений фрагмент демонструє процес фіксації дій користувача у журналі подій із зазначенням рівня ризику.

Інтерфейс системи реалізовано з використанням HTML-шаблонів і CSS. Вибір саме такого підходу пояснюється тим, що він дозволяє зосередитися на функціональності, а не на складних клієнтських технологіях. Для працівника ЦНАП важливі швидкий доступ до даних і зрозуміла структура сторінок. У системі передбачено окремі інтерфейси для роботи із записами, документами, користувачами, журналом подій та загрозами. Це дозволяє розділити функції і зменшити ймовірність помилок під час роботи. Було реалізовано механізм

резервного копіювання. Він базується на створенні копій файлу бази даних із фіксацією часу створення. Копії зберігаються у окремій директорії, що дозволяє відновити стан системи у разі збою або пошкодження даних (рисунок3.1).

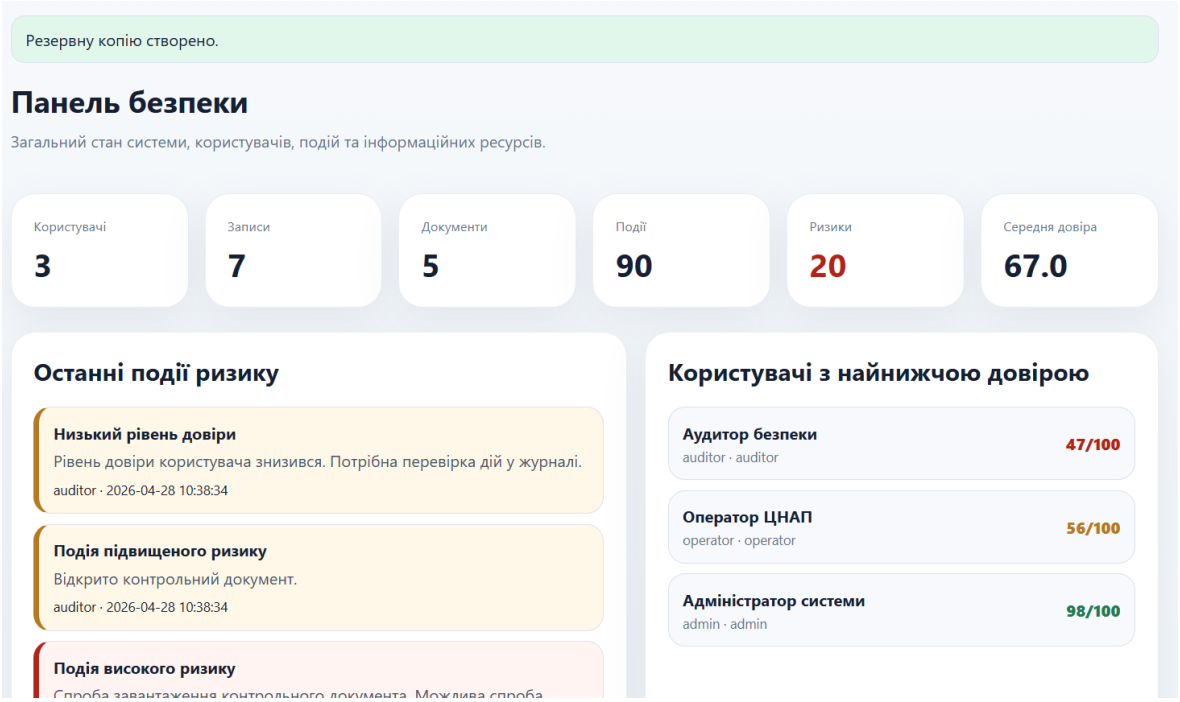


Рисунок 3.1 - Резервні копії бази даних системи

Це є простим, але ефективним, оскільки не потребує додаткових інструментів і може використовуватися навіть без спеціальної підготовки персоналу. З точки зору апаратного забезпечення система не має підвищених вимог. Вона може функціонувати на стандартному персональному комп'ютері з операційною системою Windows. Це дозволяє використовувати її у звичайних робочих умовах без необхідності придбання окремих серверів або спеціалізованого обладнання. При цьому продуктивності достатньо для обробки типових запитів і ведення журналів подій. Важливим моментом є те, що обрані засоби дозволяють легко модифікувати систему. За потреби можна змінити структуру бази даних, додати нові перевірки або розширити механізми контролю доступу без повної перебудови всієї системи. Це відповідає умовам роботи ЦНАП, де вимоги можуть змінюватися досить часто. Вибір програмних і апаратних засобів був зроблений з урахуванням не лише технічних можливостей, а й умов подальшої експлуатації. Реалізована

система не є перевантаженою, але при цьому забезпечує всі базові механізми захисту, необхідні для роботи з персональними даними.

3.2 Реалізація підсистем захисту (контроль доступу, криптографічний захист, аудит подій)

На цьому етапі система захисту розглядається вже не як набір вимог або теоретичних положень, а як конкретна реалізація, яка функціонує у програмному середовищі та взаємодіє з користувачем у процесі щоденної роботи. Основна ідея побудови полягає у тому, що жодна дія не виконується напряму. Кожен запит проходить через контрольоване середовище, де перевіряється, фіксується та аналізується. Реалізація виконана у вигляді веб-додатку (рисунок 3.2), що дозволяє централізувати всі процеси обробки даних. Це означає, що користувач не взаємодіє з базою даних безпосередньо. Усі операції виконуються через сервер, який виступає як єдина точка контролю. Такий підхід дозволяє не лише обмежити доступ, а й забезпечити повний контроль над усіма діями, що виконуються в системі. Навіть якщо користувач намагається виконати дію, яка не передбачена інтерфейсом, вона не буде виконана, оскільки сервер перевіряє кожен запит окремо.

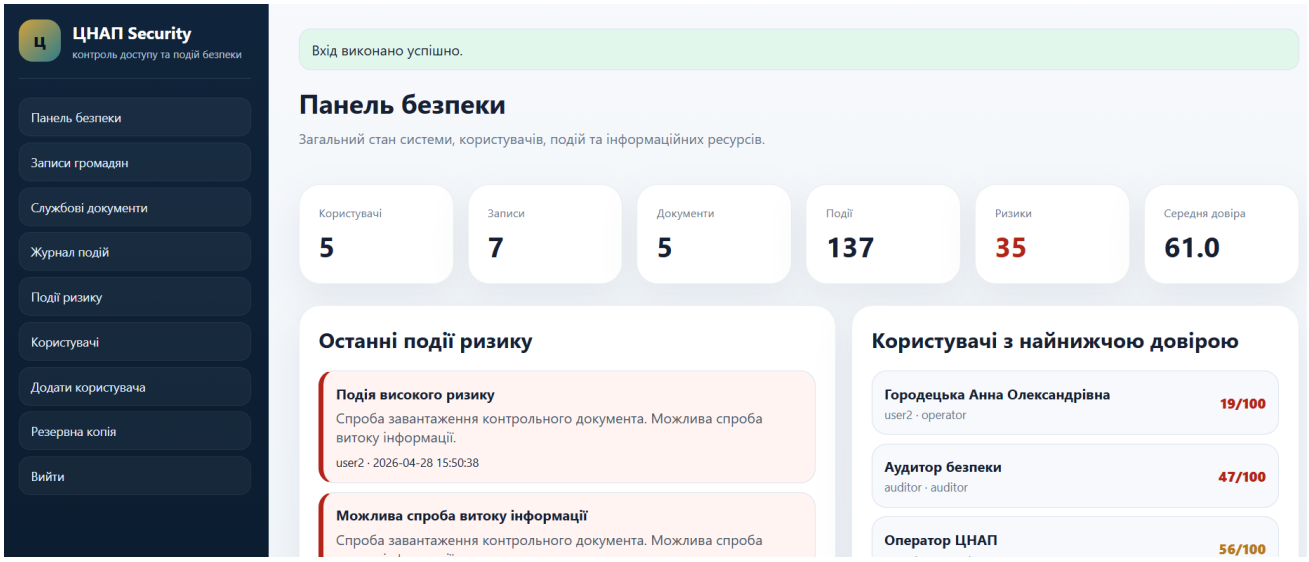


Рисунок 3.2 - Головна панель (Dashboard)

Першим етапом взаємодії є автентифікація користувача (рисунок 3.3). На цьому рівні реалізується базовий захист облікових даних. Пароль не зберігається у відкритому вигляді, що виключає можливість його використання у разі доступу до бази. Під час введення пароль одразу обробляється і перетворюється у захищене значення. Під час входу виконується аналогічна операція, після чого результати порівнюються. Такий підхід дозволяє перевіряти правильність даних без збереження їх у початковому вигляді. Окрім перевірки правильності введених даних, система аналізує сам процес входу. Якщо користувач декілька разів вводить неправильний пароль, це фіксується як підозріла активність. При цьому важливо, що система не реагує на одну помилку, а враховує їх повторюваність. Якщо кількість невдалих спроб перевищує допустиме значення, система обмежує подальші дії. Це дозволяє зменшити ефективність атак, пов’язаних із підбором паролів.

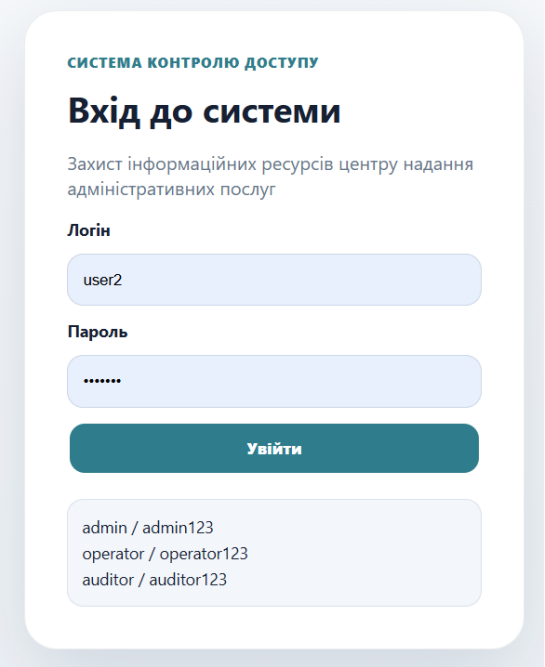


Рисунок 3.3 - Автентифікація користувача

Після успішного проходження автентифікації користувач потрапляє до системи, де доступ одразу визначається відповідно до його ролі. Ролі заздалегідь налаштовані та пов’язані з конкретними наборами дозволених дій, тому система

автоматично обмежує функціонал залежно від того, які повноваження має користувач. Перевірка прав доступу відбувається не одноразово під час входу, а супроводжує кожну дію протягом усього сеансу роботи. Завдяки цьому навіть авторизований користувач не може вийти за межі своїх дозволів. Користувач працює лише з тим функціоналом, який прямо відповідає його посадовим обов'язкам. Це дозволяє реалізувати принцип мінімально необхідних привілеїв, коли кожному надається тільки той рівень доступу, який справді потрібен для виконання робочих задач. Такий підхід підвищує загальний рівень безпеки, оскільки кожна дія додатково звіряється з політиками доступу перед її виконанням.

Система враховує не лише факт дозволу або заборони, а й умови, за яких виконується дія. Аналізується контекст операції: поточний стан даних, середовище виконання та попередня активність користувача. Одна й та сама дія може оцінюватися по-різному залежно від частоти її виконання, часу або послідовності інших операцій. Перевірка прав доступу поєднується з аналізом поведінкових характеристик користувача. Це дає змогу виявляти нетипові сценарії навіть тоді, коли формально дії не виходять за межі дозволених. У підсумку система стає більш гнучкою та здатною реагувати на потенційні ризики в режимі реального часу.

```
def roles_required(*roles):
    def decorator(func):
        @wraps(func)
        def wrapper(*args, **kwargs):
            user = current_user()
            if not user or user["role"] not in roles:
                score = change_trust(session.get("user_id"), -10)
                log_event("access_denied", "page", None, 80,
                          "Спроба доступу без прав")
                flash("Доступ заборонено", "danger")
                return redirect(url_for("dashboard"))
            return func(*args, **kwargs)
        return wrapper
    return decorator
```

Усі дії користувачів фіксуються у журналі подій. Журнал містить інформацію про користувача, тип виконаної операції, час та параметри виконання. Це дозволяє не лише відстежувати зміни у системі, а й аналізувати поведінку користувачів у динаміці. Важливо, що журнал не передбачає редагування, що забезпечує його

достовірність і виключає можливість приховування небажаних дій. Журнал подій використовується не лише як засіб фіксації, а як джерело даних для подальшого аналізу. Система враховує не окремі дії, а їх сукупність. Аналізується частота звернень, інтенсивність роботи та відхилення від звичного режиму. Наприклад, відкриття одного запису є нормальною дією, але якщо таких запитів стає значно більше, це вже розглядається як потенційний ризик роботи. Це дозволяє враховувати накопичувальний ефект дій. Якщо кожна окрема дія виглядає допустимою, система не реагує. Однак при повторенні або поєднанні таких дій формується загальна оцінка ризику. Для реалізації механізму фіксації подій у системі використовується централізована функція логування, яка записує всі дії користувачів до журналу аудиту. Фрагмент її реалізації наведено нижче:

```
def log_event(action, object_type=None, object_id=None,
              risk_score=0, description=""):
    user_id = session.get("user_id")
    username = session.get("username", "anonymous")

    conn = get_db()
    conn.execute("""
        INSERT INTO audit_log
        (user_id, username, action, object_type, object_id,
         risk_score, description, created_at)
        VALUES (?, ?, ?, ?, ?, ?, ?, ?)
    """, (user_id, username, action, object_type,
          object_id, risk_score, description, now()))
    conn.commit()
    conn.close()
```

Наведений фрагмент демонструє базовий механізм фіксації подій у системі. Кожен виклик функції супроводжується автоматичним записом інформації про користувача, тип виконаної операції та рівень ризику. Це дозволяє забезпечити централізований облік усіх дій у системі без необхідності ручного втручання. Завдяки використанню параметризованих запитів виключається можливість SQL-ін’єкцій, що додатково підвищує рівень безпеки. Таким чином, журнал аудиту стає не лише засобом зберігання інформації, а й активним елементом системи контролю безпеки. Важливим моментом є те, що різні дії мають різний вплив на рівень довіри. Це пов’язано з тим, що не всі операції однаково небезпечні. Наприклад, помилка

при введенні пароля не є критичною, тоді як спроба отримати інформацію у вигляді файлів має значно більший ризик. Саме тому для кожного типу дії визначається коефіцієнт впливу. Для узагальнення цього підходу використовується таблиця 3.1, яка відображає вплив різних дій на рівень довіри.

Таблиця 3.1 - Оцінка впливу дій користувача на рівень довіри

Дія користувача	Опис дії	Рівень впливу	Характер впливу
Невдалий вхід	Введення неправильного пароля	Низький	Тимчасове зниження довіри
Повторні помилки входу	Кілька невдалих спроб підряд	Середній	Стабільне зниження довіри
Часті запити	Велика кількість звернень за короткий час	Середній	Поступове накопичення ризику
Нетиповий час роботи	Виконання дій поза звичним робочим графіком	Середній	Контекстне зниження довіри
Масовий перегляд даних	Відкриття великої кількості записів	Високий	Значне зниження довіри
Спроба отримання даних	Завантаження або копіювання інформації	Високий	Різке зниження довіри
Повторювані дії	Багаторазове виконання однакових операцій	Середній	Накопичувальний ефект
Підозріла активність	Комбінація декількох нетипових дій	Високий	Комплексне зниження довіри

Оцінка рівня довіри дозволяє перейти від простого контролю доступу до більш гнучкого механізму аналізу поведінки. Система не лише перевіряє, чи має користувач право на дію, а й визначає, наскільки ця дія відповідає його звичному режиму роботи. На основі отриманої оцінки система приймає рішення щодо подальших дій.

Реакція системи залежить від ступеня відхилення. Якщо зміни незначні, система лише фіксує їх або формує попередження. У разі суттєвого зниження рівня довіри застосовуються обмеження. Якщо ж ризик досягає критичного значення,

доступ блокується повністю. Такий підхід дозволяє уникнути ситуацій, коли користувач блокується через одну дію. Рішення приймається на основі накопиченої інформації, що підвищує точність роботи системи. Для узагальнення реакції системи використовується таблиця 3.2.

Таблиця 3.2 - Реакція системи на зміну рівня довіри користувача

Рівень довіри (TS)	Стан користувача	Характер поведінки	Реакція системи	Мета реагування
80–100	Нормальний	Типова робота	Без обмежень	Забезпечення зручної роботи
60–79	Незначні відхилення	Поодинокі нетипові дії	Фіксація дій у журналі	Моніторинг поведінки
40–59	Підозріла активність	Часті або повторювані дії	Попередження користувача	Запобігання ризику
20–39	Високий рівень ризику	Нетипова або інтенсивна активність	Обмеження функцій (частковий доступ)	Обмеження можливих загроз
0–19	Критичний стан	Потенційно небезпечна поведінка	Блокування доступу	Запобігання порушенню

Усі підсистеми працюють у взаємозв’язку. Контроль доступу визначає можливості користувача, аудит фіксує його дії, а механізм оцінки довіри аналізує їх у динаміці. У результаті система не просто обмежує доступ, а контролює процес роботи. Реалізована система захисту дозволяє перейти від статичної моделі, у якій контроль виконується лише під час входу, до динамічної моделі, де аналіз здійснюється протягом усього часу роботи. Це особливо важливо для систем, у яких основні загрози пов’язані з діями авторизованих користувачів.

3.3 Оцінка достовірності системи захисту інформаційних ресурсів ЦНАП

Оцінка достовірності запропонованих заходів захисту виконується не шляхом формальної перевірки окремих функцій, а через аналіз поведінки системи у реальних умовах роботи. Основна увага приділяється тому, чи здатна система не

лише обмежувати доступ, а й виявляти відхилення у діях користувачів до моменту виникнення явного порушення. Такий підхід дозволяє оцінити систему не як набір механізмів, а як цілісну структуру, що реагує на зміну поведінки. Перевірка здійснювалась шляхом моделювання типових сценаріїв взаємодії користувача із системою. У нормальному режимі користувач виконує стандартні операції, такі як перегляд записів або обробка звернень. У таких умовах система не створює обмежень і не впливає на виконання дій. Це підтверджує, що реалізовані механізми не заважають звичайній роботі та не знижують зручність використання. Такий результат є важливим, оскільки надмірні обмеження можуть призводити до помилок користувачів і, відповідно, до нових ризиків. При зміні характеру роботи система починає реагувати. Наприклад, при багаторазових спробах входу або повторенні однотипних дій фіксується відхилення від нормального режиму. У цьому випадку важливо, що реакція не є миттєвою. Система накопичує інформацію і лише після цього формує оцінку ризику. Це дозволяє уникнути помилкових спрацювань та зменшує кількість необґрунтованих обмежень, які могли б впливати на роботу користувачів. Завдяки цьому захист стає гнучким і не турбує адміністраторів через кожну випадкову помилку. Система втручається лише тоді, коли поведінка користувача дійсно стає загрозливою. Результати перевірки роботи системи у різних сценаріях наведено у таблиці 3.3.

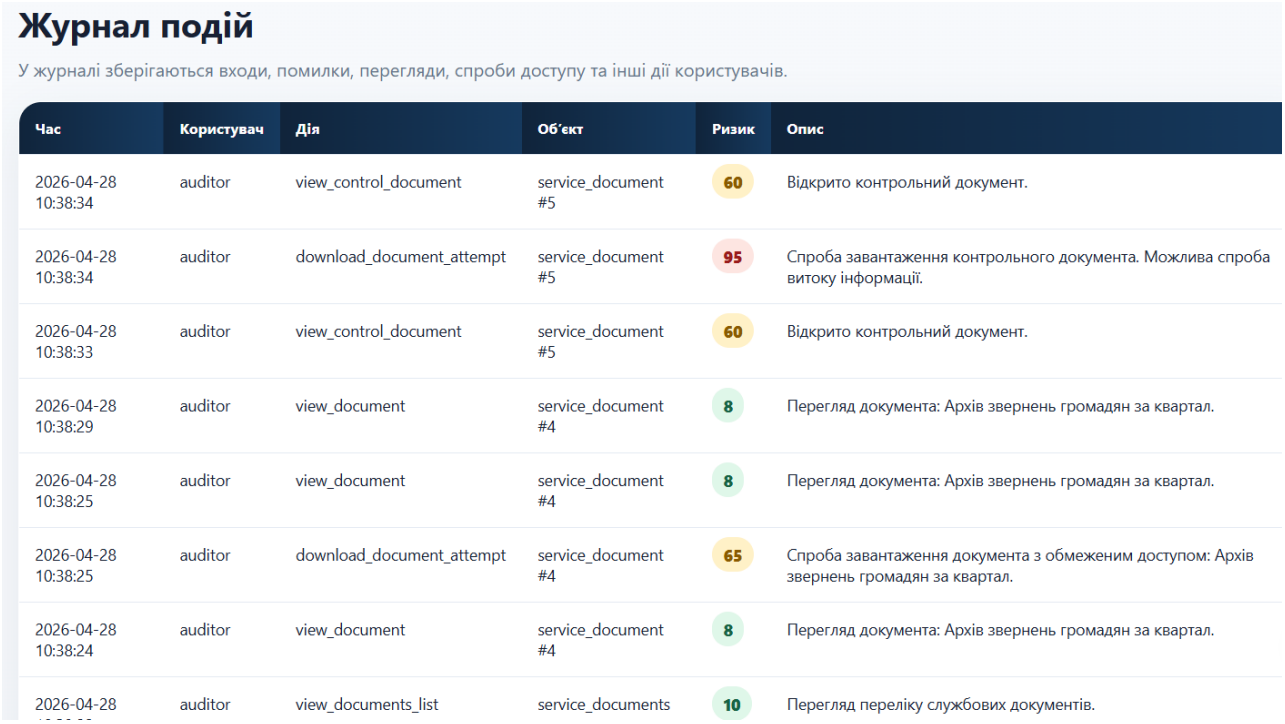
Таблиця 3.3 - Результати перевірки роботи системи

№	Сценарій перевірки	Опис дій користувача	Очікувана реакція системи	Фактична реакція
1	2	3	4	5
1	Нормальна робота	Перегляд окремих записів, стандартна активність	Відсутність обмежень	Обмеження відсутні
2	Невдалий вхід	Одноразове введення неправильного пароля	Фіксація події	Подія зафіксована
3	Повторні помилки входу	Кілька невдалих спроб підряд	Попередження або обмеження	Відображено попередження

Кінець таблиці 3.3

1	2	3	4	5
4	Часті запити	Велика кількість звернень за короткий час	Зниження рівня довіри	Рівень довіри знижено
5	Нетиповий час роботи	Виконання дій поза робочим графіком	Фіксація як підозрілої активності	Подія зафіксована

Як видно з таблиці, фактична реакція системи відповідає очікуваній. Це свідчить про те, що механізми захисту реалізовані коректно і забезпечують необхідний рівень контролю. Важливо, що система реагує не лише на критичні дії, а й на їх сукупність, що дозволяє виявляти складніші сценарії поведінки. Журнал подій, який використовується для аналізу, наведено на рисунку 3.4.



Журнал подій

У журналі зберігаються входи, помилки, перегляди, спроби доступу та інші дії користувачів.

Час	Користувач	Дія	Об'єкт	Ризик	Опис
2026-04-28 10:38:34	auditor	view_control_document	service_document #5	60	Відкрито контрольний документ.
2026-04-28 10:38:34	auditor	download_document_attempt	service_document #5	95	Спроба завантаження контрольного документа. Можлива спроба витоку інформації.
2026-04-28 10:38:33	auditor	view_control_document	service_document #5	60	Відкрито контрольний документ.
2026-04-28 10:38:29	auditor	view_document	service_document #4	8	Перегляд документа: Архів звернень громадян за квартал.
2026-04-28 10:38:25	auditor	view_document	service_document #4	8	Перегляд документа: Архів звернень громадян за квартал.
2026-04-28 10:38:25	auditor	download_document_attempt	service_document #4	65	Спроба завантаження документа з обмеженим доступом: Архів звернень громадян за квартал.
2026-04-28 10:38:24	auditor	view_document	service_document #4	8	Перегляд документа: Архів звернень громадян за квартал.
2026-04-28	auditor	view_documents_list	service_documents	10	Перегляд переліку службових документів.

Рисунок 3.4 - Журнал аудиту дій користувачів

На основі зафіксованих подій система формує загальну картину поведінки користувача. При цьому враховується не лише факт виконання операції, а й її частота, послідовність та контекст. Наприклад, окремий перегляд запису не

викликає реакції, однак серія таких дій за короткий проміжок часу вже розглядається як потенційний ризик. Це дозволяє виявляти неочевидні загрози, які складно зафіксувати при аналізі окремих подій.

Додатково перевірялась робота механізму оцінки рівня довіри. У процесі виконання дій, які відрізняються від звичайної поведінки, значення цього показника поступово зменшується. Це дозволяє відстежувати зміну поведінки у динаміці та враховувати накопичувальний ефект дій. При цьому кожна дія користувача розглядається у контексті загальної активності в системі, а не як ізольований випадок. Система аналізує частоту виконання операцій та їх характер з урахуванням рівня ризику. Також враховується повторюваність однакових або схожих дій у межах короткого проміжку часу. У разі виявлення нетипової активності рівень довіри зменшується пропорційно її потенційній небезпечності. Такий підхід дозволяє уникати різких змін показника через одиничні помилки користувача. Водночас накопичення кількох підозрілих дій формує більш суттєве зниження рівня довіри. Це забезпечує більш точну оцінку поведінки користувача в реальних умовах експлуатації системи. Динаміка зміни рівня довіри наведена у таблиці 3.4.

Таблиця 3.4 - Динаміка зміни рівня довіри користувача

Крок	Дія користувача	Характер дії	Значення рівня довіри (TS)	Зміна
1	Вхід у систему	Нормальна дія	100	-
2	Невдалий вхід	Поодинока помилка	95	−5
3	Повторна помилка	Повторювана помилка	90	−5
4	Часті запити	Інтенсивна активність	75	−15
5	Масовий перегляд даних	Потенційно ризикова дія	50	−25

Отримані значення показують, що рівень довіри знижується поступово. Це

дозволяє системі не реагувати різко на окремі дії, а приймати рішення на основі загальної поведінки користувача. Такий підхід є більш точним, оскільки дозволяє уникнути ситуацій, коли користувач блокується без достатніх підстав. Важливо, що система не реагує на окрему дію, а аналізує їх сукупність. Це дозволяє виявляти ситуації, які не є очевидними порушеннями, але можуть свідчити про підготовку до витоку інформації. Наприклад, поступове копіювання даних або робота у нетиповий час не виглядають як загроза окремо, але у поєднанні формують ризик.

При досягненні певного рівня ризику система починає реагувати. Спочатку це може бути попередження або фіксація події, що дозволяє відслідковувати ситуацію. Якщо ж відхилення продовжуються, застосовуються обмеження. У випадку критичного значення доступ користувача блокується, що дозволяє запобігти можливим наслідкам. Відображення таких ситуацій у системі наведено на рисунку 3.5.

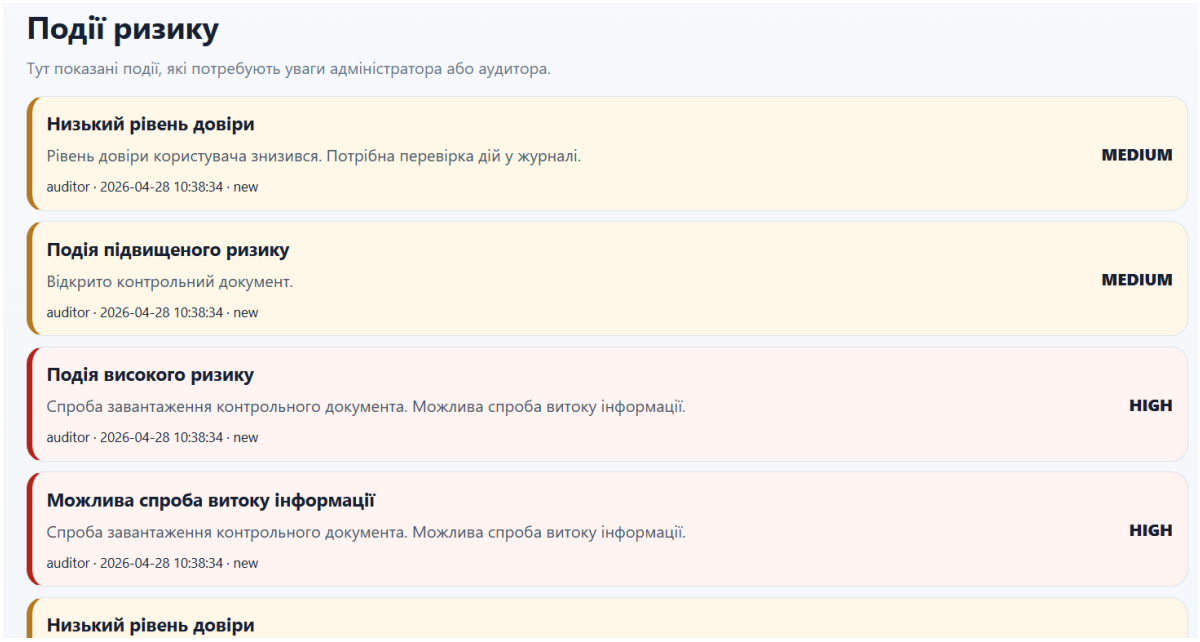


Рисунок 3.5 - Виявлення підозрілої активності

Проведена перевірка показує, що система здатна виявляти не лише явні порушення, а й приховані загрози, які формуються поступово. Це є суттєвою перевагою у порівнянні зі стандартними підходами, де реакція відбувається лише після фіксації конкретного інциденту. Додатково оцінювалась ефективність

реагування системи. У випадках, коли ризик досягає критичного рівня, блокування відбувається автоматично і не потребує втручання адміністратора. Це дозволяє зменшити навантаження на персонал і забезпечити швидку реакцію на загрози.

Для узагальнення результатів перевірки використовується співвідношення між кількістю виявлених ризикових ситуацій та загальною кількістю змодельованих сценаріїв. Це дозволяє оцінити ефективність системи не на рівні окремих прикладів, а у загальному вигляді. Чим більше ризикових ситуацій система здатна виявити, тим вищим є рівень її ефективності. Отримані результати дозволяють не лише оцінити загальну ефективність системи, а й визначити, як вона реагує на різні типи дій користувача. Це дає можливість порівняти поведінку системи у стандартних та ризикових сценаріях, а також виділити ті ситуації, які мають найбільший вплив на рівень безпеки. Такий підхід дозволяє більш обґрунтовано оцінити якість реалізованих механізмів захисту. Зведені результати оцінки наведено у таблиці 3.5.

Таблиця 3.5 - Оцінка ефективності виявлення загроз

Тип дії	Кількість випадків	Виявлено системою	Частка виявлення	Рівень ефективності
Невдалі входи	10	10	1.00	Високий
Часті запити	8	7	0.88	Високий
Масовий перегляд даних	5	5	1.00	Високий
Спроби отримання даних	4	4	1.00	Високий

Отримані результати свідчать про те, що система здатна ефективно виявляти ризикові ситуації. Особливо важливо, що вона реагує не лише на явні порушення, а й на зміну поведінки користувача. В результаті проведеної оцінки можна зробити висновок, що запропоновані заходи забезпечують своєчасне виявлення відхилень у

поведінці користувачів. Система не лише фіксує дії, а й дозволяє визначити їх потенційну небезпечність ще до моменту виникнення явного порушення. Реалізований підхід підтверджує свою ефективність у умовах, де основні загрози пов’язані не із зовнішніми атаками, а з діями авторизованих користувачів. Це дозволяє підвищити рівень захисту інформаційних ресурсів без ускладнення роботи системи.

3.4 Аналіз відповідності вимогам інформаційної безпеки

Аналіз відповідності системи вимогам інформаційної безпеки виконується з урахуванням основних принципів захисту інформації, а саме забезпечення конфіденційності, цілісності та доступності даних. Оцінка проводиться не лише на рівні окремих механізмів, а з точки зору їх комплексної роботи у процесі експлуатації системи. Такий підхід дозволяє визначити, наскільки реалізовані рішення відповідають як теоретичним вимогам, так і практичним умовам використання. Забезпечення конфіденційності інформації реалізується через обмеження доступу до даних та захист облікових записів користувачів. Паролі не зберігаються у відкритому вигляді, що виключає можливість їх використання у разі несанкціонованого доступу до бази даних. Доступ до інформації надається відповідно до ролі користувача, що дозволяє обмежити коло осіб, які можуть працювати з конкретними даними. Такий підхід відповідає принципу мінімально необхідних привілеїв, який є базовим для побудови систем захисту інформації.

Система враховує не лише факт доступу, а й характер використання інформації. Обмеження на виконання окремих дій, таких як масовий перегляд або спроби отримання даних у вигляді файлів, дозволяють знизити ризик витоку інформації. Це особливо важливо у випадках, коли загроза походить від авторизованих користувачів, які мають доступ до системи. Цілісність інформації забезпечується за рахунок фіксації всіх дій у журналі подій. Кожна зміна супроводжується записом, що містить інформацію про користувача, час та характер

виконаної операції. Це дозволяє відстежити всі зміни та визначити їх причину. Важливо, що журнал подій не передбачає редагування, що забезпечує достовірність інформації та унеможливлює приховування небажаних дій. Додатково реалізовано механізми, що дозволяють відновлювати дані у випадку помилок або збоїв. Це зменшує ризик втрати інформації та дозволяє підтримувати її цілісність навіть у нестандартних ситуаціях.

Доступність інформації забезпечується за рахунок стабільної роботи системи та можливості контролю навантаження. У нормальному режимі система не створює обмежень для користувачів, що дозволяє виконувати необхідні операції без затримок. У випадку підвищеної активності або підозрілої поведінки система може обмежувати окремі дії, що дозволяє уникнути перевантаження та зберегти працездатність. Оцінка відповідності системи нормативним вимогам виконується з урахуванням чинного законодавства України у сфері захисту інформації та персональних даних, а також загальноприйнятих принципів управління інформаційною безпекою. Реалізовані механізми відповідають вимогам щодо захисту персональних даних, контролю доступу та аудиту дій користувачів. Розробка повністю вписується в офіційні рамки і не порушує жодних державних правил. Вона залишається зручною для щоденної роботи працівників центру. Для узагальнення результатів аналізу відповідності використовується таблиця 3.6.

Таблиця 3.6 - Аналіз відповідності системи вимогам інформаційної безпеки

Вимога	Опис вимоги	Реалізація в системі	Результат	Коментар
1	2	3	4	5
Конфіденційність	Захист персональних даних від несанкціонованого доступу	Хешування паролів, розмежування доступу	Забезпечено	Дані не зберігаються у відкритому вигляді
Цілісність	Запобігання несанкціонованій зміні даних	Журнал подій, фіксація всіх змін	Забезпечено	Можливе відстеження змін

Кінець таблиці 3.6

1	2	3	4	5
Доступність	Забезпечення безперервної роботи системи	Контроль навантаження, обмеження дій	Забезпечено	Система працює стабільно
Контроль доступу	Обмеження прав користувачів	Рольова модель, перевірка кожної дії	Забезпечено	Доступ відповідає ролі
Аудит	Фіксація дій користувачів	Журнал подій без редагування	Забезпечено	Забезпечується достовірність
Виявлення загроз	Виявлення підозрілої активності	Аналіз поведінки, trust score	Забезпечено	Виявляються приховані ризики
Реагування	Реакція на порушення	Попередження, обмеження, блокування	Забезпечено	Реакція поступова
Захист від підбору	Запобігання brute-force атакам	Обмеження спроб входу	Забезпечено	Знижує ризик підбору
Контроль поведінки	Аналіз дій користувача	Оцінка частоти та типу дій	Забезпечено	Враховується контекст

Як видно з таблиці 3.6, система забезпечує відповідність не лише базовим вимогам інформаційної безпеки, а й додатковим аспектам, пов’язаним з контролем поведінки користувачів. Це означає, що захист реалізується не лише на рівні доступу до інформації, а й на рівні процесу її використання. Особливістю розробленого підходу є поєднання класичних механізмів захисту, таких як розмежування доступу та аудит, із аналізом дій користувача у динаміці. У традиційних системах контроль, як правило, обмежується перевіркою прав доступу під час входу. У даному випадку контроль продовжується протягом усього часу роботи, що дозволяє виявляти відхилення навіть у межах дозволених операцій.

Важливо, що система не створює жорстких обмежень без необхідності. Реакція залежить від характеру поведінки користувача і змінюється поступово. Це дозволяє уникнути ситуацій, коли користувач блокується через випадкову помилку. При досягненні критичного значення рівня довіри система автоматично застосовує блокування доступу користувача (рисунок 3.6).

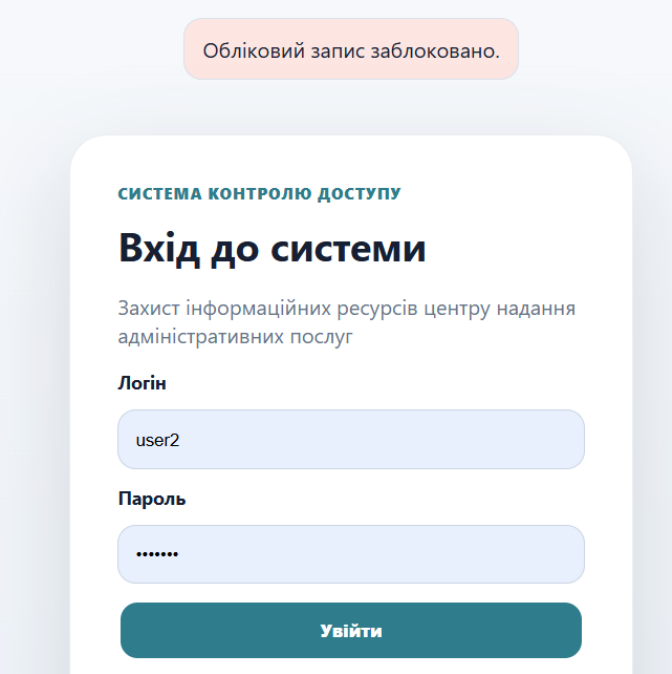
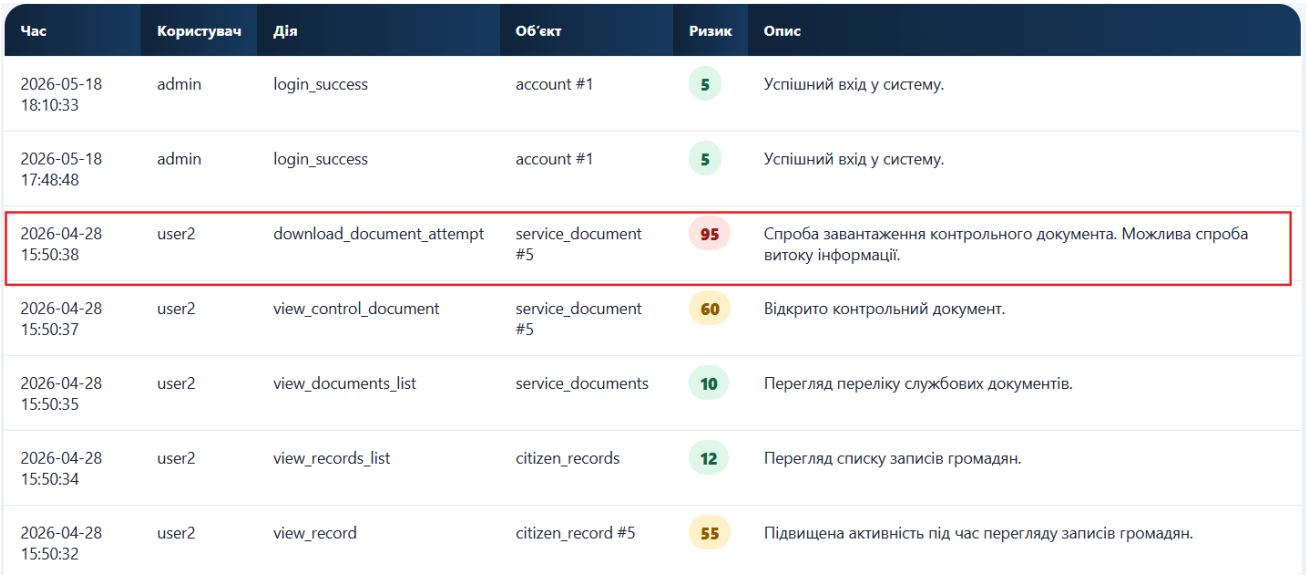


Рисунок 3.6 - Блокування користувача

Такий підхід є більш гнучким і краще відповідає реальним умовам експлуатації. Реалізовані механізми дозволяють виявляти загрози, які не мають явного характеру. Наприклад, поступове копіювання даних або виконання дій у нетиповий час не виглядають як порушення окремо, але у сукупності можуть свідчити про ризик. У традиційних системах такі ситуації часто залишаються непоміченими, тоді як у даному випадку вони враховуються при оцінці рівня довіри. Можна зробити висновок, що система відповідає не лише формальним вимогам інформаційної безпеки, а й враховує особливості реальної роботи користувачів. Це підвищує її ефективність у середовищах, де основні загрози пов’язані не із зовнішніми атаками, а з діями авторизованих осіб.

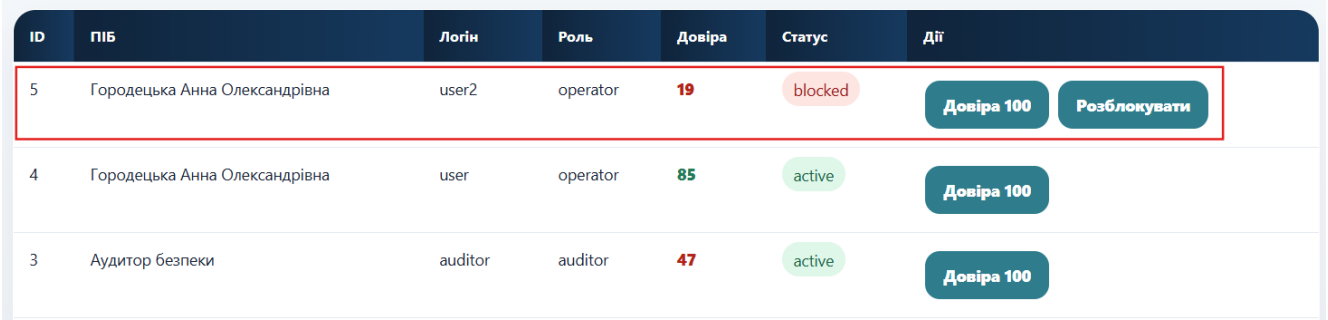
Усі дії, що призводять до блокування, автоматично записуються до журналу аудиту. У записі зберігається інформація про користувача, тип події, рівень ризику та причина спрацювання механізму безпеки. Завдяки цьому забезпечується повна фіксація критичних подій і можливість подальшого аналізу ситуації (рисунок 3.8).



Час	Користувач	Дія	Об'єкт	Ризик	Опис
2026-05-18 18:10:33	admin	login_success	account #1	5	Успішний вхід у систему.
2026-05-18 17:48:48	admin	login_success	account #1	5	Успішний вхід у систему.
2026-04-28 15:50:38	user2	download_document_attempt	service_document #5	95	Спроба завантаження контрольного документа. Можлива спроба витоку інформації.
2026-04-28 15:50:37	user2	view_control_document	service_document #5	60	Відкрито контрольний документ.
2026-04-28 15:50:35	user2	view_documents_list	service_documents	10	Перегляд переліку службових документів.
2026-04-28 15:50:34	user2	view_records_list	citizen_records	12	Перегляд списку записів громадян.
2026-04-28 15:50:32	user2	view_record	citizen_record #5	55	Підвищена активність під час перегляду записів громадян.

Рисунок 3.8 - Запис у журналі аудиту про блокування

Система змінює статус користувача в базі даних, що фактично забороняє подальшу авторизацію без участі адміністратора. Навіть при повторній спробі входу виконується перевірка актуального стану облікового запису, що не дозволяє обійти встановлені обмеження (рисунок 3.9).



ID	ПІБ	Логін	Роль	Довіра	Статус	Дії
5	Городецька Анна Олександрівна	user2	operator	19	blocked	Довіра 100 Розблокувати
4	Городецька Анна Олександрівна	user	operator	85	active	Довіра 100
3	Аудитор безпеки	auditor	auditor	47	active	Довіра 100

Рисунок 3.9 - Зміна статусу користувача в базі даних

Було також перевірено реакцію системи на повторні спроби доступу після блокування. Усі такі запити відхиляються автоматично, а відповідні події фіксуються в журналі. Це підтверджує коректну роботу механізму блокування та його стійкість до повторних спроб входу з боку користувача (рисунок 3.10).

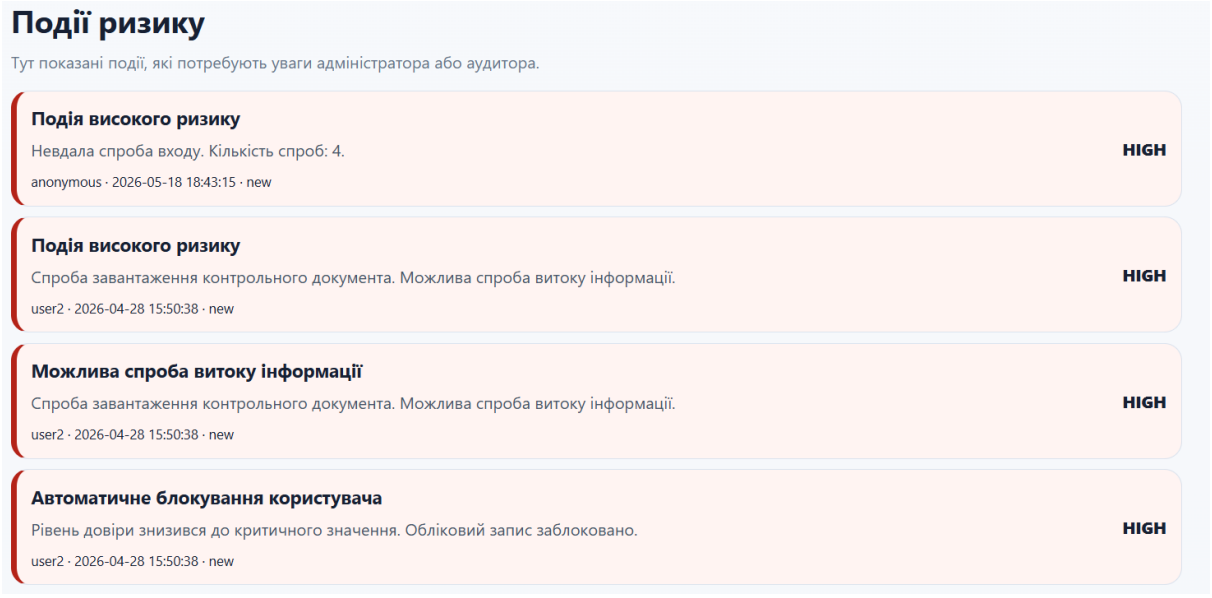


Рисунок 3.10 - Спроба повторного входу після блокування

Після фіксації спроби повторного входу заблокованого користувача система завершує обробку запиту та не виконує подальших операцій з обліковим записом. Усі такі звернення розглядаються як неуспішні та автоматично додаються до журналу подій для подальшого аналізу. Це дозволяє контролювати не лише факт блокування, а й поведінку користувача після його застосування. Таким чином забезпечується повний цикл фіксації подій - від нормальної роботи до моменту обмеження доступу та подальших спроб його обходу. Реалізований механізм підтверджує, що система коректно підтримує стан заблокованого користувача та запобігає будь-яким діям без втручання адміністратора.

У процесі дослідження роботи інформаційної системи було проаналізовано різні варіанти взаємодії користувачів із програмним середовищем. Особлива увага приділялася не лише стандартним сценаріям роботи, але й ситуаціям, які можуть створювати підвищений ризик для безпеки даних або стабільності функціонування системи. На практиці навіть незначні відхилення у поведінці користувача можуть свідчити про помилки використання, спроби несанкціонованого доступу або автоматизовану активність. Саме тому важливим елементом реалізованої системи є постійний контроль дій користувачів, аналіз їхньої активності та фіксація подій у журналі. У сучасних інформаційних системах недостатньо обмежуватися лише

процедурою авторизації. Після успішного входу система повинна продовжувати аналіз дій користувача протягом усього часу роботи із програмою. Такий підхід дозволяє своєчасно виявляти нетипову поведінку, оцінювати рівень ризику та реагувати на потенційно небезпечні дії ще до моменту виникнення інциденту безпеки. Наприклад, велика кількість однотипних операцій, часті спроби доступу до закритих функцій або масове копіювання інформації можуть свідчити про спробу отримання або виведення даних поза межами дозволених повноважень.

Важливим у системі є формування журналу подій. У ньому зберігаються записи про дії користувачів, помилки входу, зміни даних, спроби доступу до окремих функцій та інші події, які мають значення для подальшого аналізу. Наявність журналу дозволяє відстежити послідовність виконаних дій, встановити причини виникнення певних ситуацій та забезпечити контроль за роботою користувачів у системі. Крім цього, журналювання відіграє важливу роль під час аналізу інцидентів безпеки, оскільки дає можливість швидко визначити джерело проблеми та оцінити масштаби потенційного впливу. Приділено оцінюванню рівня ризику дій користувача. Реалізований механізм не розглядає всі дії однаково, а враховує їхній характер, частоту виконання та можливий вплив на роботу системи. Наприклад, разова помилка введення пароля не становить суттєвої загрози, тоді як багаторазові спроби входу або масове видалення інформації можуть свідчити про ризикову активність. У таких випадках система поступово посилює контроль, знижує рівень довіри до користувача та може застосовувати додаткові обмеження доступу.

Для більшого представлення результатів аналізу доцільно узагальнити основні сценарії взаємодії користувача із системою, можливі ризики та відповідні реакції механізмів безпеки. У таблиці нижче наведено приклади типових ситуацій, які можуть виникати під час роботи інформаційної системи, а також описано реакцію системи та її вплив на загальний рівень захисту інформації. Такий підхід дозволяє більш наочно оцінити ефективність реалізованих механізмів контролю, аудиту та реагування на потенційно небезпечні дії користувачів.

Таблиця 3.7 - Аналіз роботи системи у різних режимах роботи

№	Ситуація / сценарій	Характер дій користувача	Рівень ризику	Реакція системи	Результат роботи системи	Вплив на безпеку
1	2	3	4	5	6	7
1	Стандартна авторизація	Коректний вхід у систему	Низький	Надання доступу	Користувач успішно працює	Ризик відсутній
2	Разова помилка входу	Одноразове неправильне введення пароля	Низький	Фіксація події	Подія записується у журнал	Контроль активності
3	Повторні помилки входу	Кілька помилок поспіль	Середній	Попередження користувача	Обмеження подальших спроб	Зниження ризику brute-force
4	Часті запити до системи	Велика кількість звернень за короткий час	Середній	Зниження Trust Score	Контроль інтенсивності	Виявлення нетипової активності
5	Масовий перегляд даних	Перегляд значної кількості записів	Високий	Фіксація підозрілої активності	Дії аналізуються системою	Запобігання витоку
6	Робота у нетиповий час	Активність поза робочим графіком	Середній	Додатковий моніторинг	Подія вважається підозрілою	Підвищення контролю
7	Повторення однотипних дій	Багаторазове виконання однакових операцій	Середній	Аналіз поведінки	Формування оцінки ризику	Виявлення автоматизованих дій
8	Спроба доступу до закритих функцій	Використання недоступних можливостей	Високий	Блокування операції	Доступ заборонено	Захист привілейованих функцій
9	Інтенсивне копіювання інформації	Поступове копіювання даних	Високий	Зменшення рівня довіри	Активується контроль	Зниження ризику витоку
10	Нетипова зміна записів	Часте редагування інформації	Середній	Запис у журнал подій	Фіксація змін	Контроль цілісності
11	Масове видалення інформації	Видалення великої кількості записів	Критичний	Обмеження доступу	Припинення операцій	Запобігання втраті даних
12	Спроба зміни журналу подій	Намір приховати сліди активності	Критичний	Заборона редагування	Журнал залишається незмінним	Забезпечення достовірності

Кінець таблиці 3.7

1	2	3	4	5	6	7
13	Повторні підозрілі дії	Накопичення ризикової активності	Високий	Подальше зниження Trust Score	Посилення контролю	Аналіз поведінки у динаміці
14	Досягнення критичного рівня ризику	Сукупність небезпечних дій	Критичний	Автоматичне блокування	Обліковий запис деактивовано	Запобігання інциденту
15	Повторна спроба входу після блокування	Спроба обійти обмеження	Критичний	Відхилення авторизації	Доступ неможливий	Підтримка стану блокування
16	Робота у нормальному режимі після перевірки	Відсутність ризикових ознак	Низький	Відсутність втручання	Система працює стабільно	Комфортна робота користувача
17	Аналіз сукупності дій	Оцінка контексту поведінки	Середній	Формування загального ризику	Визначення рівня довіри	Виявлення прихованих загроз
18	Виявлення аномальної активності	Нетипова послідовність операцій	Високий	Формування сповіщення	Інформування адміністратора	Своєчасне реагування
19	Контроль навантаження	Підвищена активність користувачів	Середній	Обмеження інтенсивності	Стабілізація роботи системи	Підтримка доступності
20	Завершення ризикової сесії	Критична зміна поведінки	Критичний	Примусове завершення роботи	Припинення доступу	Мінімізація наслідків інциденту

Аналіз наведених сценаріїв дозволяє зробити висновок, що система забезпечує комплексний контроль дій користувачів та здатна своєчасно реагувати на потенційно небезпечну активність. При цьому механізми захисту працюють не ізольовано, а взаємодіють між собою, формуючи єдину систему моніторингу та контролю. Журналювання подій, оцінювання рівня ризику, контроль доступу та аналіз поведінки користувача дозволяють виявляти підозрілі дії ще на ранніх етапах їхнього виникнення. Результати показують, що навіть за наявності стандартних механізмів автентифікації необхідно реалізовувати додаткові засоби контролю активності користувачів. Це пов’язано з тим, що загрози інформаційній безпеці можуть виникати не лише через зовнішні атаки, але й унаслідок внутрішніх

помилок, необережних дій або навмисних спроб порушення правил доступу. Саме тому система повинна аналізувати не окремі події, а повну картину поведінки користувача протягом усього сеансу роботи.

Використання механізму оцінювання довіри дозволяє реалізувати більш гнучкий підхід до захисту інформації. Замість негайного блокування при будь-якому відхиленні система поступово підвищує рівень контролю залежно від накопичення ризикових ознак. Це дозволяє уникнути зайвих обмежень для користувачів, які працюють у штатному режимі, та одночасно забезпечує своєчасне реагування у випадках підозрілої активності. Такий підхід позитивно впливає як на рівень безпеки, так і на зручність використання системи. Реалізований підхід забезпечує підтримку цілісності та доступності інформації. Контроль змін у даних, обмеження критичних операцій та автоматичне блокування небезпечних дій дозволяють зменшити ризик втрати або пошкодження інформації. У випадку виникнення аномальної активності система не лише фіксує подію, а й виконує дії, спрямовані на мінімізацію можливих наслідків. Це особливо важливо для інформаційних систем, у яких обробляються персональні дані або службова інформація.

Результати проведеного аналізу показують ефективність реалізованих механізмів контролю доступу та моніторингу активності користувачів. Система забезпечує постійне спостереження за діями користувачів, своєчасно виявляє ризикові ситуації та підтримує стабільний рівень захисту інформаційних ресурсів. Узагальнення типових сценаріїв роботи дозволяє краще оцінити особливості функціонування системи та підтверджує доцільність використання поведінкового аналізу як додаткового механізму забезпечення інформаційної безпеки.

3.5 Висновок до розділу 3

У третьому розділі було розглянуто практичну реалізацію системи захисту інформаційних ресурсів та проведено оцінку її ефективності. На відміну від

попередніх розділів, де увага зосереджувалась на аналізі та проєктуванні, у даному розділі основний акцент зроблено на перевірці того, як система працює в реальних умовах. У процесі реалізації було впроваджено підсистеми контролю доступу, криптографічного захисту облікових даних та аудиту подій. Контроль доступу організовано на основі ролей користувачів із урахуванням їхніх функцій. Додатково використано механізм оцінки поведінки, який дозволяє змінювати рівень довіри залежно від дій користувача. Це забезпечує більш гнучкий підхід до безпеки у порівнянні зі статичними обмеженнями.

Проведена оцінка показала, що система коректно фіксує дії користувачів, визначає ризикові ситуації та реагує на них без необхідності ручного втручання. Журнал подій забезпечує повну історію операцій, а механізм сповіщень дозволяє своєчасно виявляти потенційні загрози. У разі досягнення критичного рівня довіри система автоматично обмежує доступ, що зменшує ймовірність подальших порушень. Аналіз відповідності підтвердив, що реалізовані рішення враховують основні вимоги інформаційної безпеки, зокрема щодо забезпечення конфіденційності, цілісності та доступності даних. При цьому система не ускладнює роботу користувачів і може бути використана у звичайних умовах функціонування ЦНАП. Результати розділу свідчать про те, що запропонована система захисту є працездатною, логічно узгодженою та придатною до практичного застосування. Реалізований підхід дозволяє не лише обмежувати доступ до інформації, а й контролювати поведінку користувачів, що є важливим у сучасних умовах обробки персональних даних.

ВИСНОВКИ

У кваліфікаційній роботі розглянуто питання забезпечення захисту інформаційних ресурсів Центру надання адміністративних послуг. У процесі виконання роботи було проаналізовано особливості функціонування інформаційної системи ЦНАП, зокрема структуру інформаційних ресурсів, типи оброблюваних персональних даних, а також організацію доступу користувачів до інформації. Проведений аналіз показав, що основні ризики пов’язані не лише з технічними недоліками, а й з особливостями щоденної роботи персоналу. Значна кількість операцій виконується вручну, що підвищує ймовірність помилок і створює умови для несанкціонованого доступу до даних. Окрему небезпеку становить можливість непомітного копіювання інформації у процесі звичайної роботи.

Було досліджено нормативно-правову базу у сфері захисту інформації та персональних даних. Встановлено, що функціонування інформаційних систем державних установ жорстко регламентується, однак практична реалізація вимог потребує додаткових технічних і організаційних рішень. Це підтверджує необхідність створення систем, які не лише формально відповідають вимогам, а й реально забезпечують контроль за обробкою інформації. На основі проведеного аналізу було визначено основні загрози та вразливості інформаційних ресурсів ЦНАП, а також сформовано вимоги до системи захисту. У подальшому розроблено модель загроз і порушника, що дозволило деталізувати можливі сценарії порушення безпеки.

У роботі запропоновано архітектуру системи захисту, яка поєднує класичні підходи до контролю доступу з механізмами аналізу поведінки користувачів. Особливу увагу приділено можливості автоматичного виявлення підозрілої активності без необхідності постійного контролю з боку адміністратора. У процесі реалізації було створено програмний прототип системи, який включає підсистеми автентифікації, контролю доступу, криптографічного захисту облікових даних та аудиту подій. Додатково реалізовано механізм оцінки рівня довіри користувача, який змінюється залежно від його дій. Такий підхід дозволяє враховувати не лише

права доступу, а й реальну поведінку користувача.

Проведена оцінка показала, що система коректно фіксує дії користувачів, визначає ризикові ситуації та реагує на них у автоматичному режимі. У разі досягнення критичного рівня довіри система обмежує доступ, що дозволяє запобігти подальшим порушенням. Це особливо важливо в умовах роботи ЦНАП, де значна частина ризиків пов’язана з людським фактором. Аналіз відповідності вимогам інформаційної безпеки підтвердив, що запропоновані рішення забезпечують конфіденційність, цілісність та доступність інформації. Реалізована система не ускладнює роботу користувачів і може бути використана у практичній діяльності.

У цілому поставлена мета роботи досягнута. Запропонована система захисту дозволяє підвищити рівень безпеки інформаційних ресурсів ЦНАП та забезпечити контроль за діями користувачів у процесі обробки персональних даних. Отримані результати можуть бути використані як основа для подальшого розвитку систем захисту в установах, що працюють з великим обсягом інформації.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Про адміністративні послуги : Закон України від 06.09.2012 № 5203-VI. URL: https://zakon.rada.gov.ua/laws/show/5203-17 (дата звернення: 26.02.2026).

2. Про інформацію : Закон України від 02.10.1992 № 2657-XII. URL: https://zakon.rada.gov.ua/laws/show/2657-12 (дата звернення: 26.02.2026).

3. Деякі питання електронної взаємодії електронних інформаційних ресурсів : Постанова Кабінету Міністрів України від 08.09.2016 № 606. URL: https://zakon.rada.gov.ua/laws/show/606-2016-п (дата звернення: 26.02.2026).

4. Про захист інформації в інформаційно-комунікаційних системах : Закон України від 05.07.1994 № 80/94-ВР. URL: https://zakon.rada.gov.ua/laws/show/80/94-вр (дата звернення: 15.03.2026).

5. Про захист персональних даних : Закон України від 01.06.2010 № 2297-VI. URL: https://zakon.rada.gov.ua/laws/show/2297-17 (дата звернення: 15.03.2026).

6. НД ТЗІ 2.5-004-99. Критерії оцінювання захищеності інформації в комп'ютерних системах від несанкціонованого доступу. Київ : ДСТСЗІ СБУ, 1999. 46 с.

7. НД ТЗІ 2.5-005-99. Класифікація автоматизованих систем і перелік обов’язкових завдань із захисту інформації від несанкціонованого доступу. Київ : ДСТСЗІ СБУ, 1999. 15 с.

8. Про доступ до публічної інформації : Закон України від 13.01.2011 № 2939-VI. URL: https://zakon.rada.gov.ua/laws/show/2939-17 (дата звернення: 22.03.2026).

9. Про електронну ідентифікацію та електронні довірчі послуги : Закон України від 05.10.2017 № 2155-VIII. URL: https://zakon.rada.gov.ua/laws/show/2155-19 (дата звернення: 22.03.2026).

10. ДСТУ ISO/IEC 27001:2023. Інформаційні технології. Методи захисту. Системи управління інформаційною безпекою. Вимоги. URL: https://zakon.isu.net.ua/sites/default/files/normdocs/dstu_iso_iec_27001_2023.pdf (дата звернення: 22.03.2026).

11. Концепція застосування блокчейн-технологій для підвищення

захищеності персональних даних платформи «дія». Кібербезпека: освіта, наука, техніка. 2024. URL: https://csecurity.kubg.edu.ua/index.php/journal/article/view/681/564 (дата звернення: 22.03.2026).

12. Про основні засади забезпечення кібербезпеки України : Закон України від 05.10.2017 № 2163-VIII. URL: https://zakon.rada.gov.ua/laws/show/2163-19 (дата звернення: 22.03.2026).

13. NIST SP 800-207. Zero Trust Architecture. Gaithersburg : NIST, 2020. URL: https://csrc.nist.gov/pubs/sp/800/207/final (дата звернення: 15.04.2026).

14. Загальний регламент про захист даних (GDPR) : Регламент ЄС № 2016/679 від 27.04.2016. URL: https://zakon.rada.gov.ua/laws/show/984_011 (дата звернення: 15.04.2026).

15. Інститут інформації, безпеки і права НАПрН України. Науковий висновок 2024/5. URL: https://ippi.org.ua/sites/default/files/2024-5.pdf (дата звернення: 15.04.2026).

16. ДСТУ ISO/IEC 27002:2023. Інформаційні технології. Методи захисту. Звід правил для заходів інформаційної безпеки. Київ : ДП «УкрНДНЦ», 2023. 162 с.

17. НД ТЗІ 1.1-003-99. Термінологія в галузі технічного захисту інформації. Основні поняття. Київ : ДСТСЗІ СБУ, 1999. 24 с.

18. Про хмарні послуги : Закон України від 17.02.2022 № 2075-IX. URL: https://zakon.rada.gov.ua/laws/show/2075-20 (дата звернення: 15.04.2026).

19. Updated Digital Identity Guidelines. NIST Insights. 2026. URL: https://www.nist.gov/blogs/cybersecurity-insights/lets-get-digital-updated-digital-identity-guidelines-are-here (дата звернення: 22.04.2026).

20. Загальні вимоги до кіберзахисту об'єктів критичної інфраструктури : Постанова КМУ від 19.06.2019 № 518. URL: https://zakon.rada.gov.ua/laws/show/518-2019-п (дата звернення: 22.04.2026).

21. Методика виявлення аномалій взаємодії користувачів з інформаційними ресурсами організації. Захист інформації. 2024. URL:

https://journals.dut.edu.ua/index.php/dataprotect/article/view/2827/2728 (дата звернення: 22.04.2026).

22. NIST SP 800-122. Guide to Protecting the Confidentiality of Personally Identifiable Information (PII). Gaithersburg : NIST, 2010. URL: https://csrc.nist.gov/publications/detail/sp/800-122/final (дата звернення: 22.04.2026).

23. Про затвердження Порядку ведення Реєстру об'єктів критичної інформаційної інфраструктури : Постанова КМУ від 09.10.2020 № 943. URL: https://zakon.rada.gov.ua/laws/show/943-2020-п (дата звернення: 22.04.2026).

24. Про електронні документи та електронний документообіг : Закон України від 22.05.2003 № 851-IV. URL: https://zakon.rada.gov.ua/laws/show/851-15 (дата звернення: 22.04.2026).

25. Про електронні комунікації : Закон України від 16.12.2020 № 1089-IX. URL: https://zakon.rada.gov.ua/laws/show/1089-20 (дата звернення: 22.04.2026).

26. Про Національну програму інформатизації : Закон України від 04.02.1998 № 74/98-ВР. URL: https://zakon.rada.gov.ua/laws/show/74/98-вр (дата звернення: 22.04.2026).

27. Довідник ЦНАП м. Києва. Київ : КМДА, 2025. URL: https://kyivcnap.gov.ua/Content/dovidnyk.pdf (дата звернення: 22.04.2026).

28. Рекомендації CERT-UA щодо безпеки IT-інфраструктури. URL: https://cert.gov.ua/recommendation/16904 (дата звернення: 22.04.2026).

29. NIST SP 800-53 Rev. 5. Security and Privacy Controls for Information Systems and Organizations. URL: https://csrc.nist.gov/pubs/sp/800/53/r5/upd1/final (дата звернення: 22.04.2026).

30. NIST Lightweight Cryptography Standard for Small Devices. 2025. URL: https://www.nist.gov/news-events/news/2025/08/nist-finalizes-lightweight-cryptography-standard-protect-small-devices (дата звернення: 22.04.2026).

31. Про інформаційно-комунікаційну систему «Вулик» : Постанова КМУ від 2024 р. № 1441. URL: https://zakon.rada.gov.ua/laws/show/1441-2024-п (дата звернення: 22.04.2026).

32. НД ТЗІ 1.1-002-99. Загальні положення щодо захисту інформації в

комп'ютерних системах від несанкціонованого доступу. Київ : ДСТСЗІ СБУ, 1999. 12 с.

33. Про затвердження Мінімальних вимог до захисту інформаційних систем : Постанова КМУ від 29.03.2006 № 373. URL: https://zakon.rada.gov.ua/laws/show/373-2006-п (дата звернення: 25.04.2026).

34. Захист інформації в комп'ютерних система : навч. посібник / КПІ ім. Ігоря Сікорського. 2025. URL: https://scs.kpi.ua/storage/2025/09/zi_in_cs.pdf (дата звернення: 25.04.2026).

35. НД ТЗІ 3.7-003-2005. Побудова комплексних систем захисту інформації на об'єктах інформаційної діяльності. Київ : ДССЗЗІ, 2005. 28 с.

36. НД ТЗІ 3.7-001-99. Методичні вказівки з розроблення технічного завдання на створення КСЗІ. Київ : ДСТСЗІ СБУ, 1999. 22 с.

37. НД ТЗІ 1.4-001-2000. Типове положення про службу захисту інформації в автоматизованій системі. Київ : ДСТСЗІ СБУ, 2000. 18 с.

38. НД ТЗІ 3.1-001-07. Створення комплексних систем захисту інформації. Порядок розроблення та впровадження заходів із захисту інформації. Київ : ДССЗЗІ, 2007. 18 с.

39. НД ТЗІ 2.5-008-02. Вимоги із захисту конфіденційної інформації в автоматизованих системах класу «1». Київ : ДСТСЗІ СБУ, 2002. 22 с.

40. НД ТЗІ 3.7-004-99. Побудова комплексних систем захисту інформації. Загальні положення. Київ : ДСТСЗІ СБУ, 1999. 20 с.

41. CIS Controls Version 8. CIS. URL: https://www.cisecurity.org/controls/v8 (date of access: 28.04.2026).

42. ENISA Threat Landscape 2025 | ENISA. Home | ENISA. URL: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2025 (date of access: 28.04.2026).

43. NIST Computer Security Resource Center | CSRC. URL: https://csrc.nist.rip/external/nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.08042021-draft.pdf (дата звернення: 28.04.2026).

44. ДСТУ ISO/IEC 27005:2023 Інформаційна безпека, кібербезпека та

захист конфіденційності. Настанова керування ризиками інформаційної безпеки (ISO/IEC 27005:2022, IDT). БУДСТАНДАРТ Online - нормативні документи будівельної галузі України. URL: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=104400 (дата звернення: 29.04.2026).

45. DSpace :: ELAKPI :: Репозитарій КПІ ім. Ігоря Сікорського. URL: https://ela.kpi.ua/server/api/core/bitstreams/0dbba7ac-40e4-4a83-80b7-f729109cfe9a/content (дата звернення: 29.04.2026).

46. Zero-trust based dynamic access control for cloud computing - Cybersecurity. SpringerLink. URL: https://link.springer.com/article/10.1186/s42400-024-00320-x (date of access: 29.04.2026).

ДОДАТОК А

КРБКБ. 220236.22.02.24 Е8

Модель загроз

Контроль доступу Trust Score

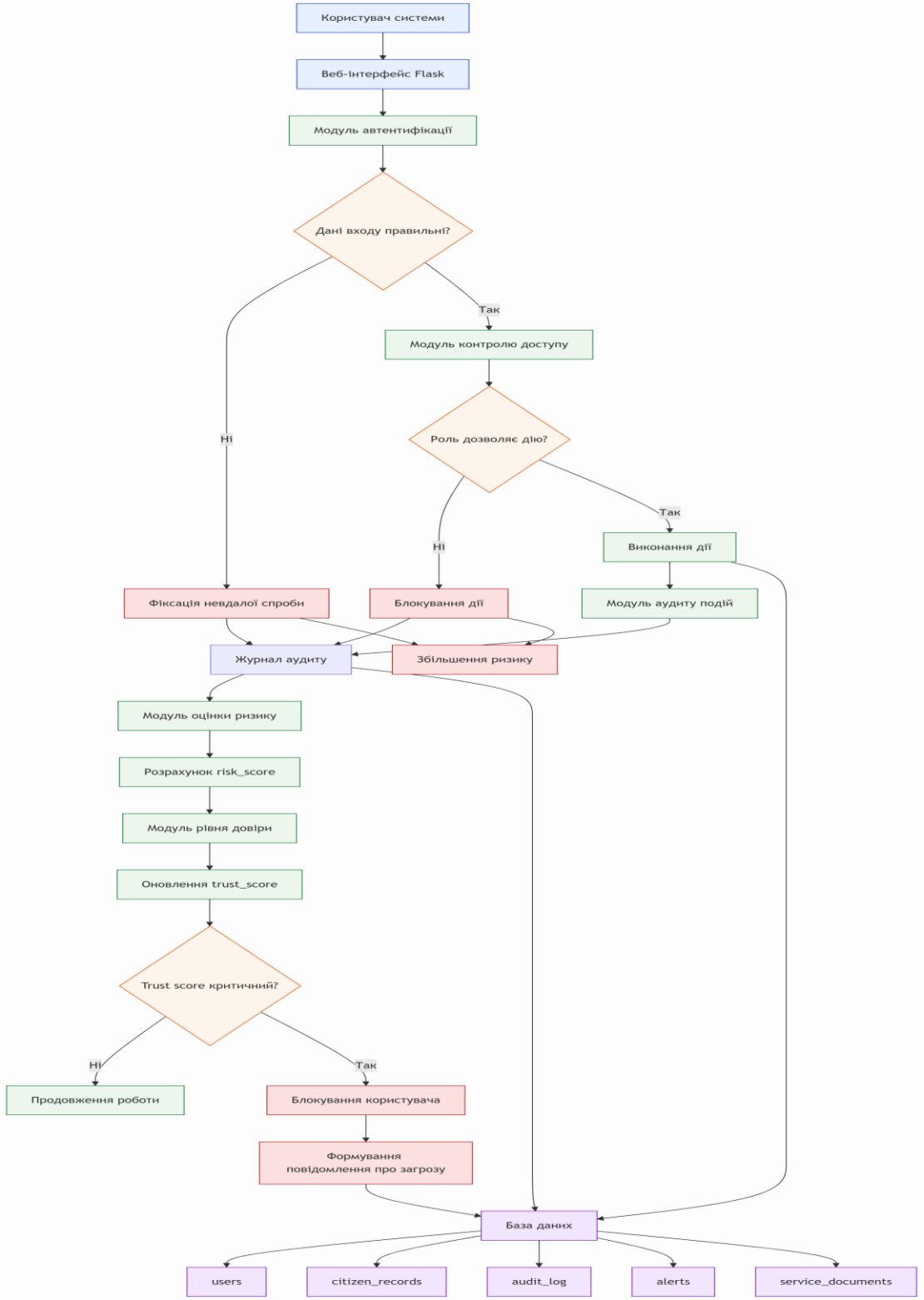
Алгоритм роботи системи

Структурна схема інформаційної системи ЦНАП

					КРБКБ. 220236.22.02.24 Е8		
Зм.	Арк.	№ докум.	Підпис	Дата	Система захисту інформаційних ресурсів центру надання адміністративних послуг. Пояснювальна записка	Літера	Маса	Масштаб
Розроб.		Городецька А.О.				Н		
Перевір.		Петяк Н. С.						
Т. контр.						Аркуш	Аркушів 1	
Н. контр.		Петяк Н. С.				ХНУ, КБ-22-2		
Затв.		Кльод Ю.П.						

ДОДАТОК Б

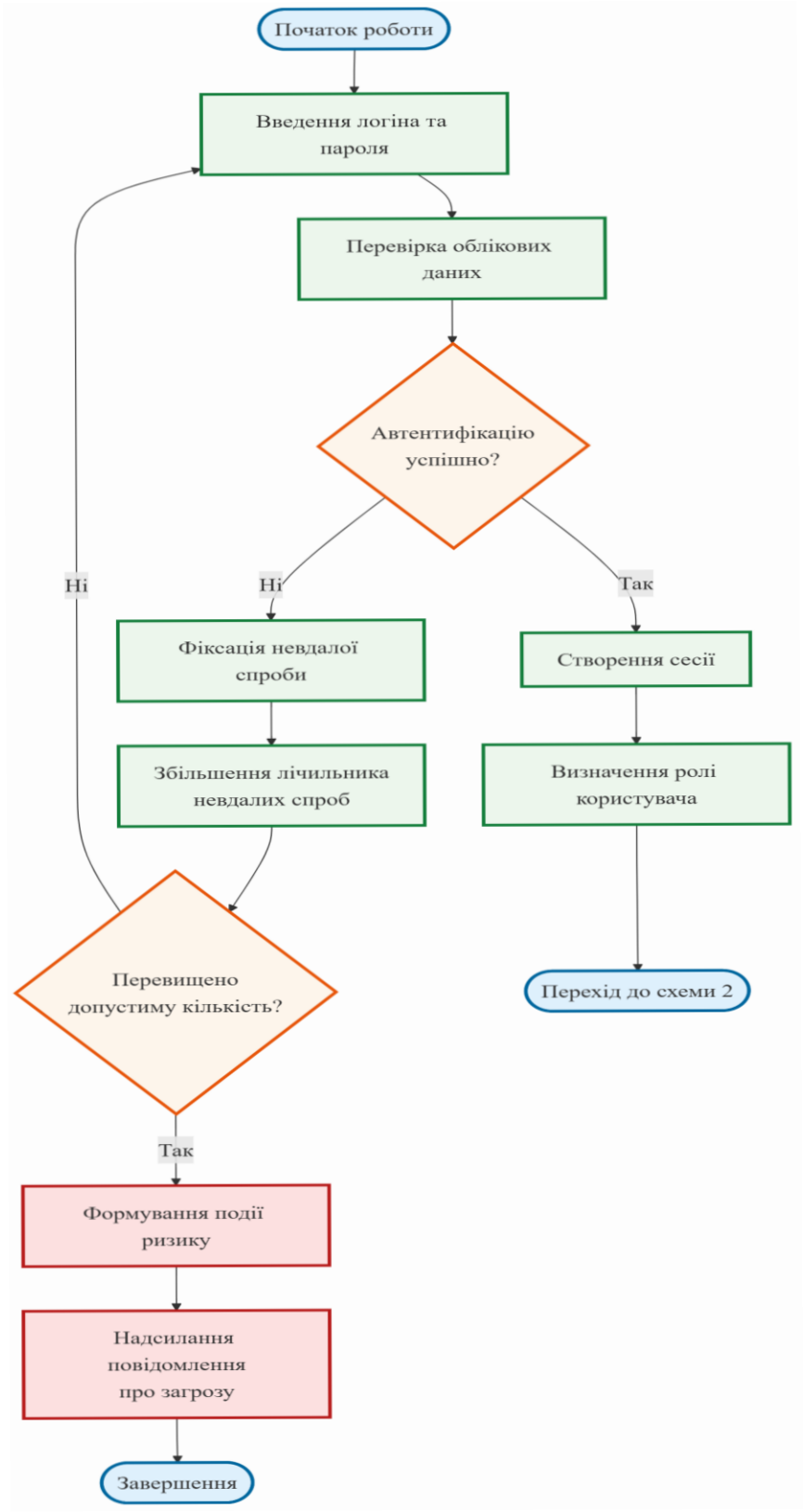
КРКБ.220236.22.02.24 Е8



Зм.	Арк.	№ докум.	Підпис	Дата	КРКБ.220236.22.02.24 Е8		
Розроб.		Гордацька А. О.			Система захисту інформаційних ресурсів центру надання адміністративних послуг. Пояснювальна записка	Літера	Маса	Масштаб
Перевір.		Пятяк Н. С.				Н		
Т. контр.						Аркуш	Аркушів 1	
Н. контр.		Пятяк Н. С.				ХНУ, КБ-22-2		
Затв.		Ключ Ю. П.						

ДОДАТОК В

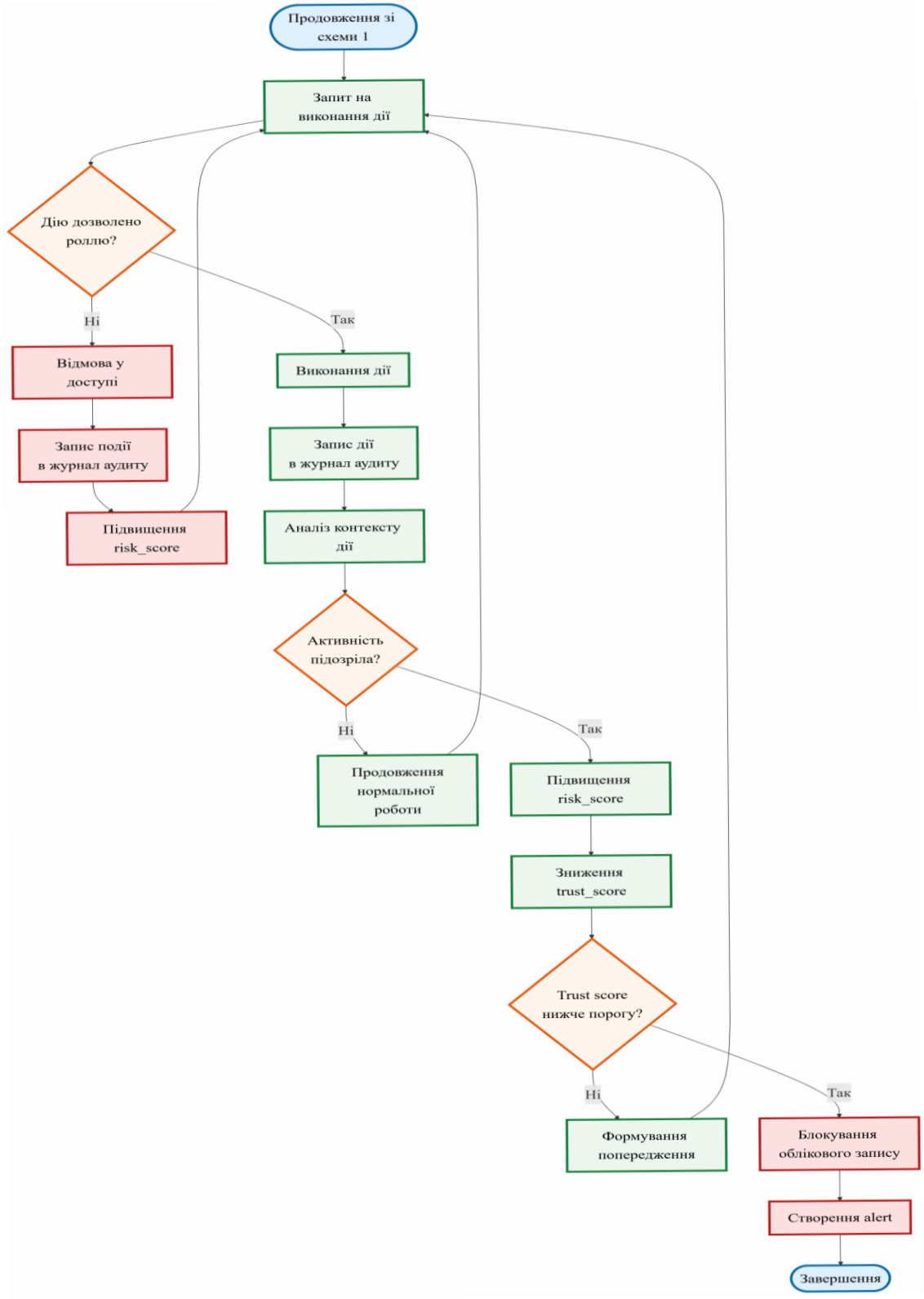
КРКБ.22 02 36.22 .02 .24 Е8



					КРКБ.220236.22.02.24 Е8		
Зм.	Арк.	№ докум.	Підпис	Дата	Система захисту інформаційних ресурсів центрів надання адміністративних послуг Пояснювальна записка	Літера	Маса	Масштаб
Розроб.		Гордацька А. О.				Н		
Перевір.		Пятяк Н. С.						
Т. контр.						Аркуш	Аркушів 1	
Н. контр.		Пятяк Н. С.				ХНУ, КБ-22-2		
Затв.		Клюцю Ю. П.						

ДОДАТОК Г

КРКБ.22 02 36.22.02.24 Е8



					КРКБ.220236.22.02.24 Е8			
Зм.	Арк.	№ докум.	Підпис	Дата	Система захисту інформаційних ресурсів центру надання адміністративних послуг. Пояснювальна записка	Літера	Маса	Масштаб
Розроб.		Гощацька А. О.				Н		
Перевір.		Пеник Н. С.						
Т. контр.						Аркуш	Аркушів 1	
Н. контр.		Пеник Н. С.				ХНУ, КБ-22-2		
Затв.		Кльоц Ю. П.						

ДОДАТОК Д

```
def current_user():
    if "user_id" not in session:
        return None

    conn = get_db()
    user = conn.execute(
        "SELECT * FROM users WHERE id = ?",
        (session["user_id"],)
    ).fetchone()

    conn.close()
    return user


def change_trust(user_id, delta):
    if not user_id:
        return None

    conn = get_db()

    user = conn.execute(
        "SELECT trust_score FROM users WHERE id = ?",
        (user_id,)
    ).fetchone()

    score = None

    if user:
        score = max(
            0,
            min(100, user["trust_score"] + delta)
        )

        conn.execute(
            "UPDATE users SET trust_score = ? WHERE id = ?",
            (score, user_id)
        )

        conn.commit()

    conn.close()
    return score


def check_trust_limits(user_id, username, score):
    if score is None:
        return

    if score <= 25:

        conn = get_db()

        conn.execute(
            "UPDATE users SET status = 'blocked' WHERE id = ?",
            (user_id,)
        )
```

```
            conn.commit()
            conn.close()

            add_alert(
                user_id,
                username,
                "high",
                "Автоматичне блокування користувача",
                "Рівень довіри знизився "
                "до критичного значення."
            )

        elif score <= 50:

            add_alert(
                user_id,
                username,
                "medium",
                "Низький рівень довіри",
                "Потрібна перевірка "
                "дій користувача."
            )


    def log_event(
            action,
            object_type=None,
            object_id=None,
            risk_score=0,
            description=""
    ):

        user_id = session.get("user_id")

        username = session.get(
            "username",
            "anonymous"
        )

        conn = get_db()

        conn.execute("""
        INSERT INTO audit_log (
            user_id,
            username,
            action,
            object_type,
            object_id,
            risk_score,
            description,
            created_at
        )
        VALUES (?, ?, ?, ?, ?, ?, ?, ?)
        """, (
            user_id,
            username,
            action,
            object_type,
```

```
            object_id,
            risk_score,
            description,
            now()
        ))

        conn.commit()
        conn.close()

        if risk_score >= 75:

            add_alert(
                user_id,
                username,
                "high",
                "Подія високого ризику",
                description
            )

        elif risk_score >= 50:

            add_alert(
                user_id,
                username,
                "medium",
                "Подія підвищеного ризику",
                description
            )


    def roles_required(*roles):

        def decorator(func):

            @wraps(func)
            def wrapper(*args, **kwargs):

                user = current_user()

                if not user or user["role"] not in roles:

                    score = change_trust(
                        session.get("user_id"),
                        -10
                    )

                    check_trust_limits(
                        session.get("user_id"),
                        session.get("username"),
                        score
                    )

                    log_event(
                        "access_denied",
                        "page",
                        None,
                        80,
                        "Спроба доступу "
                        "без відповідних прав."
```

```
            )

            flash(
                "Доступ заборонено.",
                "danger"
            )

            return redirect(
                url_for("dashboard")
            )

        return func(*args, **kwargs)

    return wrapper

return decorato
```