Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему
Комплексна система захисту розподіленого програмного середовища передачі даних

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека

КРМКБ.180131.22.01.19 ПЗ

Виконав: студент 2 курсу, група КБм-22-1 _____ Нагребецький О.В.
Підпис

Керівник к.т.н, доцент _____ Тітова В. Ю.
Підпис

Нормоконтролер старший викладач _____ Мостовий С.В.
Підпис

До захисту допускаю:
Зав. кафедри кібербезпеки, к.т.н., доц _____ Кльоц Ю.П.
Підпис

14 грудня 2023 р.

Хмельницький, 2023

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Кібербезпеки

Освітній рівень Магістр

Галузь знань 12 Інформаційних технологій

Спеціальність 125 Кібербезпека

Освітня програма Кібербезпека

ЗАТВЕРДЖУЮ

Зав. кафедри Ю.П. Кльоц

"30" 08 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Нагребецькому Олексію Валентиновичу
Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Комплексна система захисту розподіленого програмного середовища

Керівник роботи Тітова Віра Юріївна
Прізвище, ім'я, по батькові, науковий ступінь, вчене звання
кандидат технічних наук, доцент
Затверджена наказом № 30 ректора університету, додаток №25 від 15.08.2023

2. Строк подання студентом проєкту (роботи) на кафедру

3. Вихідні дані до проекту (роботи) створення повноцінного захищеного комплексу збереження інформації з розподілом навантажень та попереджень інформаційних загроз

4 Зміст пояснювальної записки (перелік питань, які потрібно розробити) Вступ. Дослідження предметної області у сфері побудови комплексного захищеного середовища. Передпроектне планування та облаштування складових проекту. Синтез захищеного комплексу обробки даних. Комплексна система захисту інформації та її тестування. Висновки

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

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали і посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Мостовий С.В., ст. викладач		

7. Дата видачі завдання «01» вересня 2023р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Вибір напряму дослідження і узгодження тематики КРМ з керівником	01.09.2023	
2	Ознайомлення з предметною областю; формулювання мети і задач дослідження; визначення об'єкта і предмета дослідження	04.09.2023	
3	Робота над розділом 1 – аналіз відомих моделей, методів за темою; постановка задачі	18.09.2023	
4	Робота над розділом 2 – розробка моделей і методів для вирішення поставленої задачі	02.10.2023	
5	Робота над розділом 3 – розробка алгоритмів і технологій, їх аналіз	16.10.2023	
6	Робота над розділом 4 – апробація запропонованих рішень	06.11.2023	
7	Робота над науковою публікацією	10.11.2023	
8	Узгодження отриманих результатів, оформлення пояснювальної записки згідно вимог	15.11.2023	
9	Попередній захист роботи	17.11.2023	
10	Захист роботи на засіданні ЕК	06.12.2023	

Студент _____ О.В Нагребецький
Підпис Ініціали, прізвище

Керівник проекту (роботи) _____ В.Ю. Тітова
Підпис Ініціали, прізвище

АНОТАЦІЯ

Тема кваліфікаційної роботи: Комплексна система захисту розподіленого програмного середовища передачі даних

Автор роботи: Нагребецький Олексій Валентинович

Керівник роботи: к.т.н., доц. Тітова Віра Юріївна

Загальний обсяг роботи: 70 сторінок, 33 рисунки, 2 таблиці, 35 додатків, 58 посилань.

Ключові слова: інформаційне середовище, комплекс, платформа.

Метою роботи є повноцінний робочий комплекс для збереження та передачі інформації у системі з імітацією роботи реальних користувачів.
Даний проект реалізує автоматизовану систему обміну інформації у мережі інтернет, за допомогою розподілених технологій між локальним та хмарним середовищем AMAZON та інкапсулює у собі відповідні хмарні можливості системи. Дана реалізація включає у собі сучасні практичні реальні можливості доступних технологій захисту та передачі інформації. Використання технологій PHP Laravel, можливостей зручних типів UI (Tailwind) та систем сповіщень на базі соціальних мереж. На базі цих засобів був розроблений проект, котрий включає у собі багатосторонні продумані рішення максимально наближені до реальних проектів.

Реалізований програмний продукт надає користувацький інтерфейс, повноцінне віддалене робоче середовище, функціонал обміну даних, робота за базою та оновленнями на сервері.

Аналіз можливих загроз, проведення тестів та використання методів шифрування. Використані технології у продукті - Postman API, PHP Artisan, Sanctum, Telegram, Laravel Fortify, Amazon S3 technologies (SQS, RDC, EC2), Tailwind, програми реалізації захисту програмних середовищ, методи абстрактної організації формувань.

13.12.2023 р.

ANNOTATION

Theme of qualification work: Integrated protection system for distributed software datá transmission environment

Author of the work: Nahrebetskyi Oleksii Valentynovych

Mentor: Ph.D. Titova Vira Yuriivna

Total volume of work: 70 pages, 33 figures, 2 tables, 35 appendices, 58 links.

Keywords: information environment, complex, platform.

The purpose of the work is a full-fledged working complex for storing and transmitting information in the system with simulation of real users.

This project implements an automated system for exchanging information on the Internet, using distributed technologies between the local and cloud environments of AMAZON and encapsulates the corresponding cloud capabilities of the system. This implementation includes modern, practical, real-world capabilities of available technologies for protecting and transmitting information. The use of PHP Laravel technologies, the capabilities of user-friendly UI types (Tailwind) and notification systems based on social networks. On the basis of these tools, a project was developed that includes multifaceted, well-thought-out solutions as close as possible to real projects.

The implemented software product provides a user interface, a full-fledged remote working environment, data exchange functionality, work with the database and updates on the server.

It analyzes possible threats, conducts tests, and uses encryption methods. The technologies used in the product include Postman API, PHP Artisan, Sanctum, Telegram, Laravel Fortify, Amazon S3 technologies (SQS, RDC, EC2), Tailwind, software environment protection programs, methods of abstract organization of formations.

13.12.2023

ЗМІСТ

ВСТУП

Напрямок дослідження, що реалізується в рамках поточної кваліфікаційної роботи: комплексна система захисту розподіленого програмного середовища передачі даних.

Актуальність дослідження. Дослідження існуючих рішень підходу комплексних систем захисту інформації та створення власних механізмів інформаційної безпеки.

По-перше, в епоху безпрецедентного технологічного прогресу повсюдне поширення взаємопов'язаних цифрових систем докорінно змінило наш спосіб життя і роботи. Однак ця цифрова трансформація також піддала нас новим і витонченим загрозам, які ставлять під загрозу цілісність, конфіденційність і доступність чутливої інформації.

По-друге, оскільки наша залежність від взаємопов'язаних мереж поглиблюється, потреба в надійних заходах кібербезпеки стає все більш першочерговою.

Закладення основ, еволюція ландшафту кібербезпеки, від перших днів існування ізольованих систем до нинішньої ери хмарних обчислень та Інтернету речей (IoT) цифрова екосистема стала свідком зміни парадигми, що вимагає відповідної еволюції підходів до кібербезпеки. Завжди, визначні інциденти в галузі кібербезпеки підкреслюються зростаючою складністю кіберзагроз, починаючи від атак на об'єкти критичної інфраструктури, з метою вимагання викупу і закінчуючи сучасними постійними загрозами (APT), націленими на урядові та корпоративні структури. Заглиблюючись в історичний контекст, основою для більш глибокого вивчення викликів, з якими стикаються сучасні фахівці з кібербезпеки є непроглядною пітьмою майбутнього. [1]

Спираючись на фундамент, закладений у зачатках розвитку кібербезпеки, наче розділі, у ньому проводиться всебічний аналіз поточного ландшафту загроз. Фахівці намагаються визначати і класифікувати різні типи кіберзагроз, починаючи від шкідливих програм і фішингових атак і закінчуючи

вразливостями нульового дня та інсайдерськими загрозами. Аналізування також розглядає мотиви кібератак, чи то фінансова вигода, шпигунство, активізм або просто підрив діяльності. Розуміння мотивів і методів, які використовують суб'єкти загроз, має на меті надати цілісне уявлення про багатогранні виклики, з якими стикаються фахівці з кібербезпеки. Крім того, в ньому підкреслюється глобальний характер кіберзагроз і необхідність спільного міжнародного підходу до кібербезпеки. [2]

Розглянуті існуючі рамки та стандарти кібербезпеки, якими організації керуються при створенні надійних механізмів захисту. У ньому критично оцінюються такі рамки, як NIST Cybersecurity Framework, ISO/IEC 27001 та CIS Controls, досліджуються їхні сильні сторони, обмеження та застосовність у різних контекстах. Також, розглядається роль дотримання нормативних вимог у формуванні практик кібербезпеки, підкреслюється необхідність для організацій узгоджувати свої підходи до безпеки з галузевими та регіональними нормативними актами. Вивчаючи ці рамки, дана робота прагне надати дорожню карту для організацій, які прагнуть посилити свою кібербезпеку та привести її у відповідність до найкращих галузевих практик. [3]

Отже, дослідження та впровадження нових методів, що використовуються у комплексних системах захисту розподілених програмних середовищ, стає важливою стратегією для забезпечення інформаційної безпеки у сучасному інформаційному середовищі. Правильний підхід до систем подібного класу, дозволить підвищити рівень захисту конфіденційної інформації, відповідати регуляторним вимогам і зменшити зовнішні та внутрішні загрози.

Мета дослідження - дослідити та проаналізувати сучасні стратегії кібербезпеки перед обличчям новітніх загроз, прагнучи забезпечити комплексне розуміння ландшафту, що розвивається, та запропонувати практичне рішення що буде ефективним механізмом захисту та користування.

Завдання дослідження - є створення власної системи розподіленого комплексу захисту та обробки інформації, система з віддаленим доступом,

механізмами валідації, обробки даних та користувацьким інтерфейсом відповідно до результатів досліджень.

Предмет дослідження – використання захищених аутсорс утиліт для захищеного комплексу обробки інформації, пошук методу альтернативи підтвердження дії реального користувача.

Об'єкт дослідження – забезпечення контролю доступу та інформаційної безпеки комплексної системи захисту розподіленого програмного середовища передачі даних.

Методи дослідження. Для дослідження комплексу розподіленого середовища обміну інформації, для забезпечення інформаційної безпеки використовуються наступні методи:

1. Аналіз інформаційних джерел: проводиться огляд наукових публікацій, статей, робіт, реальних існуючих відкритих програмних рішень, що стосуються систем комплексу розподіленого середовища обробки інформації. Цей аналіз допомагає отримати загальний огляд видів систем, а також нормативних вимог і стандартів до їх функціонування практично використаних існуючих рішень.

2. Аналіз стандартів і нормативних вимог: Проводиться детальний аналіз стандартів і вимог, які стосуються систем комплексних систем захисту розподіленого програмного середовища передачі даних. Це допомагає встановити вимоги до безпеки та функціонування таких систем.

3. Аналіз концепції інформаційної безпеки: Проводиться аналіз загальної концепції інформаційної безпеки комплексних систем захисту розподіленого програмного середовища передачі даних, включаючи методи захисту даних і виявлення загроз.

Наукова новизна дослідження покращення методу захисту даних шляхом розгалуження міжсерверного зв'язку у поєднанні з методом "Action Key", котра з комбінацією інших встановлених аплікацій дозволяє гарантувати, що певна дія була виконана саме користувачем.

Практична цінність отриманих результатів надати повноцінний комплекс, практично-зрозумілий посібник для фахівців з кібербезпеки, організацій і дослідників. Аналізуючи історичні тенденції, розглядаючи сучасний можливості та шляхи загроз, оцінюючи існуючі структури та досліджуючи потенціал нових технологій, це дослідження має на меті забезпечити зацікавлених осіб знаннями та інструментами, необхідними для навігації в складному ландшафті кібербезпеки в цифрову епоху.

Використані методи являють собою есенцію отриманих теоретичних знань та практичного досвіду під час роботи над подібними проектами та системами подібного класу. Усі рішення є максимально наближеними до реального комерційного проекту. Продукують повноцінний комплекс обробки та збереження інформації з аутсорс утилітами, серверною частиною, сформованою аплікацією середовища, системою повідомлень-сповіщень та зручним користувацьким інтерфейсом.

До переліку публікацій за темою кваліфікаційної роботи відноситься стаття подана у журнал Вісник ХНУ. Стаття за темою кваліфікаційної роботи наведена у Додатку А.

1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ У СФЕРІ ПОБУДОВИ КОМПЛЕКСНОГО ЗАХИЩЕНОГО СЕРЕДОВИЩА

1.1 Інформаційні ресурси, як об’єкт захисту інформації

В області досліджуваного спектру захисту інформації першочерговими є розуміння захисту та його функціональних складових. Систематизація об’єктів та суб’єктів складових взаємодій інформаційного поля. Під час дослідження даної тематики були виконані аналітичні висновки, а також аналіз комплексних систем захищених середовиз, як об’єкту захисту інформаціної складової.

Інформаційні ресурси є необхідністю, несуть у собі нематеріальну цінність активів організацій, підприємців чи установ. Їхній захист необхідний, для забезпечення конфіденційності, цілісності та доступності до приватних даних створюються величезні, складнопобудовані комплекси обчислень. Інформаційні ресурси охоплюють широкий спектр цифрових та фізичних активів, включаючи, але не обмежуючись ними:

Дані та їх збереження в базах даних:

Життєвою складовою будь-якого організму є операційний відділ. База даних являє собою мозок системи, вона зберігає, аналізує, повертає, обчислює, тощо. Ця складна система є основою збереження інформації та маніпулювання нею. Сучасні система збереження інформації вкладають у собі виключно хмарні (аутсорс) технології, тобто більшість баз даних і великих комплексів обрахувань даних, у масштабних компаніях таких як Google чи Adidas, Youtube, вони виключно користуються послугами інших великих організацій, що надають послуги з обчислювальних машин. Ця ніша розрослась настільки сильно, що за кожну дію несе відповідальність окрема організація зі своїми фахівцями. IT послуги компаній з надання серверної частини - є найбільшою. [4]

Сервери (пристрої збереження та обрахування) - це спеціалізована фізично-функціональна апаратура з обмеженим користувацьких інтерфейсом, що являє собою серце комплексної системи збереження інформації. Основною її

роллю є обрахування даних користувачів, підключених до їх сервісу. По факту, це звичайна металева коробка, яка напхана величезною кількістю потужних процесорів та пам’яті зі ще більшою охолоджувальною системою. Кількість можливих обрахувань у подібних комплексах може досягати трильйонів у секунду, що надає змогу одночасній роботі з більш ніж 100 тисячами користувачів по всьому світу. Світовими лідерами з надання серверів в оренду (хостингу) є Amazom, GoDaddy, HostGator, тощо. Ці системи являють собою величезні кластери розмірами з ангари, до яких звертаються тисячі користувачів у світі. Подібні системи потребують постійного нагляду, ретельних перевірок та окремим відбірок з політики безпеки. Сучасна тенденція конкурентоспроможного ринку змушує подібні організації вкладатися не лише у потужності цих систем, а й у їх захист. На допомогу таким компаніям приходять організації з кіберзахисту. Йде розподіл доступу у ланках співробітників, надання механізмів збереження, поширення копіювання інформації та передбачення її помірного користування. [5]

Мережі та комунікаційні системи:

Будь-який користувач не зможе передавати інформацію іншому користувачеві без фізичного контакту з ним. Мережеві комунікації являють собою судини та нерви комплексних систем передачі інформації. Канали зв'язку у таких системах слугують входами та виходами обчислювальної техніки для обміну даними між ними. Подібні системи надаються у користування операторами. Неважливо, яким саме методом буде передана інформація, оптоволокном, радіо хвилями чи коливаннями, тощо. Усі мережі потребують захисту, оскільки являють найбільш незахищену матеріальну складову захищених систем. Головною причиною є широке розгалуження, що не дає змоги цілодобово захищати такі системи від небажаних вторгнень. Окрім людського фактору, найбільш впливовими є природні чинники. Люта зима, сильна гроза, різкі перепади температури, тощо, усі ці аномалії слугують загрозою безперебійної передачі інформації у системі. Одночас, людський фактор несе у собі загрозу фізичного пошкодження, несанкціонованому доступу до джерел

інформації, перехопленню чи підробці даних під час передачі.. Захист цих систем має важливе значення для запобігання несанкціонованому доступу, перехопленню чи підробці даних під час передачі. Подібні системи, через свою розгалуженість потребують найбільшої кількості персоналу. З пошуку нових методів передачі даних, їх експлуатації та догляду, а для їх захисту взагалі створюються окремі підрозділі у кожному раніше перерахованому етапі. [6-7]

Програмне забезпечення:

Інформаційні ресурси завжди працюють за певними алгоритмами, ці алгоритми, можна назвати необхідними ресурсами для функціонування організму в цілому. Такі системи є на кожному етапі передачі даних між системами на апаратному чи програмному рівнях. Програми створюють логіку передачі, зберігання, метаморфози та аналізу даних. Усі потужності “заліза” являють собою виключно шматок металу без подібних людських налаштувань. Оскільки, ці системи є виключно людським ресурсом, тобто втіленням інтелектуальної праці спеціалістів, вона є найбільш вразливою для людських вторгнень. Такі системи несуть загрозу для стабільної роботи системи через людський фактор. Невірність побудови обрахувань чи алгоритму шифрувань, невірне практичне користування апаратних потужностей, непередбачені реальні кейси поломок, тощо. Подібні помилки несуть проблеми не лише у передачі інформації, а й її збереженні. [8]

Одним з таких прикладів, можна привести поломку GeForce RTX 4090, котра викликала масовий збій в обрахуванні необхідної потужності BAT у відеокарті, через яку вона буквально згорала. Оскільки, цю помилку не знайшли одразу при тестах, її вихід на ринок змусив втратити компанію чималих коштів. У свою ж чергу, це викликало проблему - чому апаратна система не отримала механізмів блокування подібного? [9]

Загалом, програмне забезпечення можна розподілити таким чином:

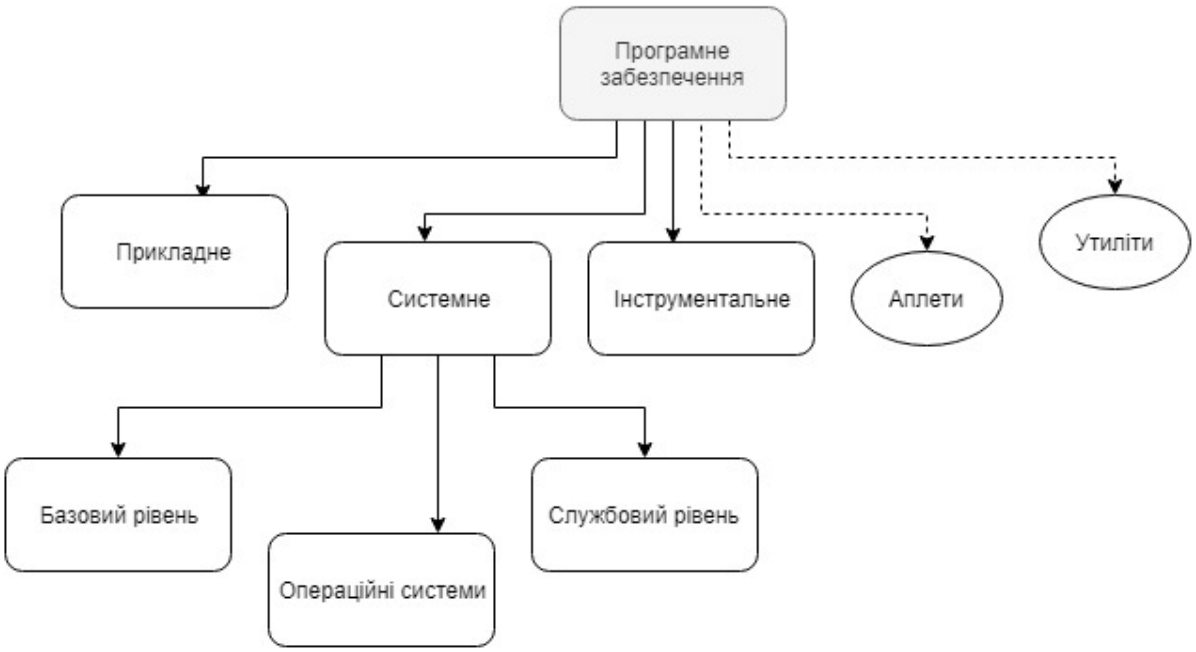


Рисунок 1.1 - Розподіл програмного забезпечення за класами

У даній роботі найбільша увага буде приділена утилітам, прикладному та службовому рівням.

Документація або політика безпеки:

Система, що побудована виключно для пояснення користування системою загалом, її обслуговуванню та реакціям на загрози. Передбачають у собі й захист самої документації, оскільки, більшість таких матеріалів несуть у собі конфіденційну інформацію, пояснення методів роботи у системі, загальне уявлення та опису, як система буде реагувати на проблеми в системі та пояснення самих проблем у системі. Створення посібників, документацій чи інших письмових матеріалів, котрі спрямовані для полегшення роботі людини зі системою є складним та багатоетапним процесов. [10]

Зазвичай, такі системи не є обов’язковими, вони з’являються за потребою, під час росту регулятора (самого об’єкту захисту), вони включають у собі усім відомі правила, які є регламентованими порадами від місцевих (державних) чи приватних організацій, діяльність яких спрямована на роботоспроможність та захист даних систем.

Основною проблемою є “Фактор неефективності” таких комплексних рішень. ПІБ не приносить користі, якщо ігнорується, виключним фактором є сама її наявність у організації, як формальний аргумент перевірки чи спору. Таку збірку можна розділити за 2 критеріями - формулювання та практично-необхідне застосування. [11]

Формулювання ПІБ полягає у собі розуміння користувача (читача) як безпосереднього об’єкту результатів обробки системи. Досить складно знайти консенсус у цій справі, оскільки люди, які пишуть подібні матеріали значно краще володіють теоретичною складовою специфіки системи, а перехідний користувач може й уявлення не мати про приховані системні операції, що проходять повз його зору, у фоновому режимі. Правильна структура та чітке пояснення у системі “це працює так” і “у випадку цього - зроби ось це” дозволяє звичайному користувачу системи оперативно реагувати на проблеми, без втручань висококваліфікованого персоналу. Окрім цього, правильний опис здатен закрити прогалини у захисті системи, оскільки будь-яка система має свої системні необхідності у наданні доступів людиною до обробки чи збереженню власної конфіденційної інформації.

Практично ефективні - це широке пояснення самої роботоздатності системи. Не завжди методи, що описані у документації можуть призвести до бажаного результату. Цією причиною є постійна модернізація системи, її видозмінення, у таких випадках важко передбачати істотні зміни, можливі помилки та їх рішення. Як би гарно не була сформульована відповідь для запиту користувача, якщо воно не приносить потрібного результату ні на жоден реальний запит під час експлуатації, така документація не варта своїх матеріальних вкладень та не несе жодної доцільної цінності. [12]

У ході аналізу даного підпунтку, варто зазначити, що подібна комплексна система передачі інформації повинна містити у собі чітко розподілені етапи побудови системно-серверної частини та документацію. Оскільки, рішення є виключно абстрактним, не варто витрачати багато ресурсів для створення повноцінної ПІБ, на її заміну можна обійтись невеликими записками для

пояснення загальних правил користування. Головної ж уваги потребує реалізація методів чи використання вже існуючих для захисту системи та збереженню її роботоздатності при різних факторах взаємодії зі зовнішніми чинниками.

1.2 Інформаційні загрози та методи протидії ним

З огляду на те, що перелік інформаційних загроз не може бути вичерпним, варто розглянути методів розподілу, заснованих на подібних критеріях. Таким чином, більшість загроз можна розділити за виміром інформаційної безпеки, на який спрямована загроза:

Загрози конфіденційності (несанкціонований доступ до інформації). Загроза порушення конфіденційності означає, що інформація стає відомою особам, які не мають права доступу до неї. Це відбувається, коли здійснюється доступ до інформації з обмеженим доступом, що зберігається в комп'ютерних системах або передається з однієї системи в іншу. Термін "компрометація" використовується по відношенню до загрози порушення конфіденційності. Такі загрози можуть виникати внаслідок "людського фактору" (наприклад, ненавмисної передачі привілеїв іншому користувачеві), збоїв у програмному або апаратному забезпеченні. Інформація з обмеженим доступом включає державну таємницю (комерційну таємницю, персональні дані), професійну таємницю: лікарську, адвокатську, банківську, нотаріальну, страхову, таємницю слідства та судочинства; листування, телефонні розмови, поштові, телеграфні та інші повідомлення (конфіденційні); відомості про сутність винаходів, корисних моделей і промислових зразків до їх офіційної публікації (ноу-хау)). Загрози цілісності (незаконна модифікація даних). [13]

Загрози цілісності - це загрози чи вразливості, пов'язані з можливістю надійності та узгодженої передачі даних, їх зміни та точність результуючих частин, що зберігається в інформаційних системах. Даний сегмент являє собою розподіл на випадкові зміни користувачів, шкідливе програмне забезпечення та взлом через злоумисників, збої у роботі жорстких накопичувачів даних, помилки

запрограмованих пристроїв, природні катаклізми, перехоплення типу “людина посередині”, підсуховуванню, недостатній валідації вхідних даних, застарілості систем чи відсутності регулярного резервного копіювання. [14]

Загрози доступності відповідають за ризики чи вразливості, котрі порушують або ж погіршують доступність і функціональність інформаційних систем. Доступність систем даних має головне значення для безперервної роботи комплексу обробки та збереження інформації. Основними загрозами для доступності до даних можна вважати атаки на відмову в обслуговуванні (DDos), перевантажений трафік на обробці запитів, порушення роботи внутрішніх чи зовнішніх сервісів, збої в роботі серверів чи програмного забезпечення, втрати підключення до живлення, помилки конфігурацій, а також природні катаклізми. Взагалі, основними проблемами є технічно-фізичні чинники, пов’язані з перезавантаженням системи або ж її фізичними пошкодженнями. [15]

Беручи до уваги подібну кількість факторів несправності, варто зазначити системні операнди, котрі будуть з ними боротись. У сучасному світі диференціювання поставлених задач дозволяє прискорити роботу та полегшити процеси розробки.

За походженням джерела загроз слід поділити на внутрішні та зовнішні. До внутрішніх варто віднести усі джерела загроз внутрішнього походження, тобто проблеми які з них виникають є результатом взаємодії елементів комплексу системи. До зовнішніх ж варто вносити усі чинники, котрі впливають на систему не знаходячись у системі.

Подолання інформаційних загроз є більш складним та розгалуженим етапом під час створення захищених комплексів захисту інформації. Подібною причиною є постійне реагування та конкуренція організацій, роботою яких є передбачити та звести до мінімуму будь-які матеріальні збитки причиною яких є ці загрози. [16]

Одним з початкових етапі є СУБІ - система управління безпекою інформації. Під час її створення визначається мета захисту, причини її необхідності та кількість інформації яка буде охоплена системою. Розглядаються

законно-регулятивні норми, що стосуються захисту та обробки інформації. Проводиться аналіз загроз і побудова сценаріїв реагувань.

Рисунок 1.2 – Схема класифікацій загроз на інформаційних ресурсах

Створюється політика, у якій зазначається основні принципи безпеки, рівні конфіденційності, цілісності та доступності до інформаційного джерела системи.

Політика ролей та відповідальності з приводу безпеки інформації спеціалістів установи, її співробітників та аутсорс персоналу. Проводиться навчання з питань безпеки та уніфікація основних методів протидії загрозам.

Важливим питанням також являється практична ефективність та зручність у користуванні системою. Політика збереження електронних носіїв та паперових документів з важливою інформацією. Типи довірених технологій. [17]

Будь-яка захищена система має істотну проблему - це втрачання своїх інтелектуальних потужностей з часом. Для передбачення подібних проблем вводиться дві попереджуючі системні методи - регулярна оцінка ризиків та забезпечення безпеки та конфіденційності робочого функціоналу системи.

Проведення оцінок ризиків надає можливості передбачати потенційні загрози та знайти методи протистояння проти них, такі рішення потребують постійних вкладень у людські та матеріальні ресурси, проте вони окуповються при першій ж загрозі для системи.

Окрім цього, оцінка ризиків дозволяє скорегувати сили організації на необхідні рішення. Прикладом такого, може бути постійне надання системних прогалин найвищих пріоритетів, подібна методика перекриває невдалості системи найшвидше, що дозволяє підвищити ефективність роботи й безпеку самої системи.

Не менш важливою складовою будь-якого комплексу є система повідомлень про виявлення загроз, а також повідомлень про помилки, що вже відбулись у системі. Якщо перехопити помилку нескладно, бо вже існують механічні методи обробки, то передбачення загроз і повідомлення про них є більш тяжкою складовою. Для подібного використовують треновані аналітичні штучні інтелекти, або ж відділи аналізу. У таких відділах працюють спеціалісти, котрі мають спеціалізацію пов’язану з постійним моніторингом даних. [18]

Висновком даного пункту є поглиблена оцінка ризиків комплексу системи, створення рівнів доступу та методів реагування, спілкування з користувачами програми та безпосередня взаємодія систем між собою. Необхідно створити відповідні моделі загроз, оцінки ризиків, рівні доступу, розподілити інформацію на сегменти доступності, встановити функціонал контролю та моніторингу у системі.

1.3 Інформаційна безпека комерційних гігантів

У даному пункті я хочу провести дослідження по справжньому потужних організацій пов’язаних зі збереженням та передачею даних користувачів. Даний аналіз допоможе зрозуміти на що варто звернути увагу та які ресурси вже є поширеними та використовуються у системах подібного класу.

Першою прикладною системою я обрав Google, її величезна кількість користувачів та розгалужена багаторівнева система з безлічі сервісів створює справжнього мегалодона у світі інформаційної безпеки. Від зрозумілої політики безпеки, взаємодій з користувачами до багаторівневого захисту кожної складової їх процесу.

Почнемо з головного екрану та розгляду існуючих утиліт акаунту. Мінімалістичний стиль UI оманює око, наче каже що усе просто. Та за кожним вікном побудована величезна схема та проведена робота багатьох відділів, з метою надання найякіснішого функціоналу.

Я не планую розглядати кожне вікно, а виключно три, котрі найбільш важливі та необхідні для правильного розуміння функціоналу системи. А саме: Data & Privacy, Security та Security Recommendations (на рисунку 1.3).

Дана система зберігає не лише незмінну інформацію про користувача, а й постійно аналізує та відслідковує його дії у мережі. Подібна інформація дозволяє сервісу заробляти тим, що налаштовує рекламу та контент на відповідні смаки користувача. Персоналізація системи збереження та розповсюдження інформації є необхідною. Усі права реалізації такої інформації повинна проходити виключно, відповідно до його побажань.

Двофакторна автентифікація (2FA) додає додатковий рівень безпеки для вашого облікового запису Google. Окрім введення пароля під час входу, для цього зазвичай потрібно отримати код підтвердження на телефон. Google пропонує функцію перевірки паролів, яка може перевіряти надійність і безпеку ваших паролів. Він сповістить вас, якщо будь-який із ваших паролів було зламано через порушення даних.

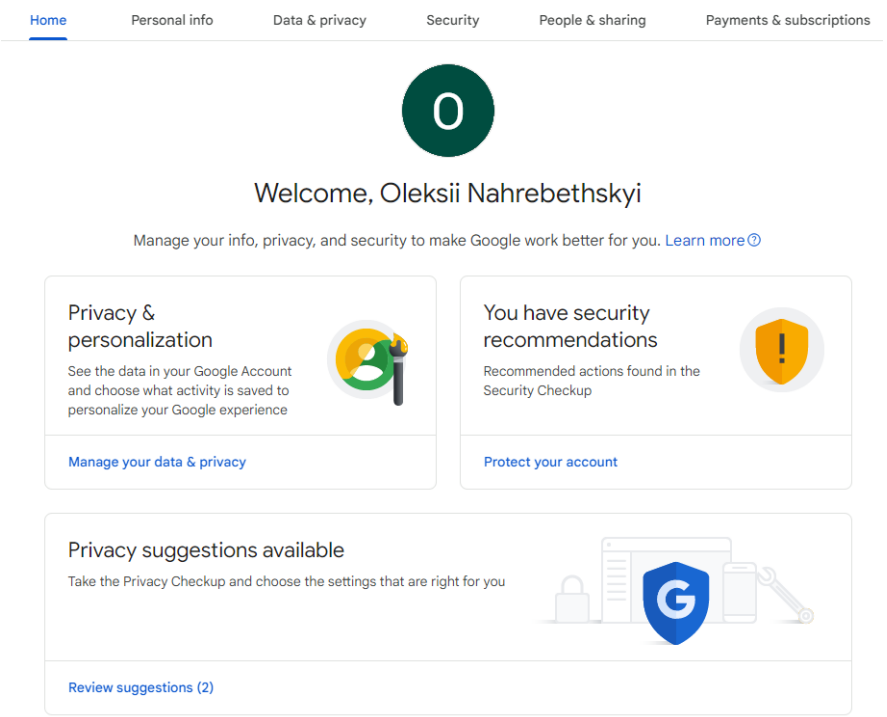


Рисунок 1.3 – Головне вікно профілю користувача у кабінеті Google

Одним з головних для користувача функціоналів є система відслідковування його діяльності. Дана система дозволяє самому користувачеві знайти аномалії діяльності свого акаунту та сліди несанкціонованого доступу. Подібний моніторинг дозволяє передбачити діяльність зловмисника, або ж побачити що відбулося з акаунтом. Дану систему варто вважати звичайним логуванням. На кожну дію користувача відбувається запис відповідного екшену.

Переглянемо налаштування безпеки та його функціонал. Google пропонує різноманітні параметри безпеки та конфіденційності, які користувачі можуть налаштувати у своїх профілях для додаткового захисту своєї інформації. Ці налаштування охоплюють такі аспекти, як доступ до облікового запису, обмін даними та можливості відновлення.

Налаштування параметрів відновлення облікового запису, наприклад адресу електронної пошти та номер телефону для відновлення. Відновлення доступу до акаунту у випадках втрати доступу до нього з різних причин, допомагає користувачам обходитись без допомоги спеціаліста.

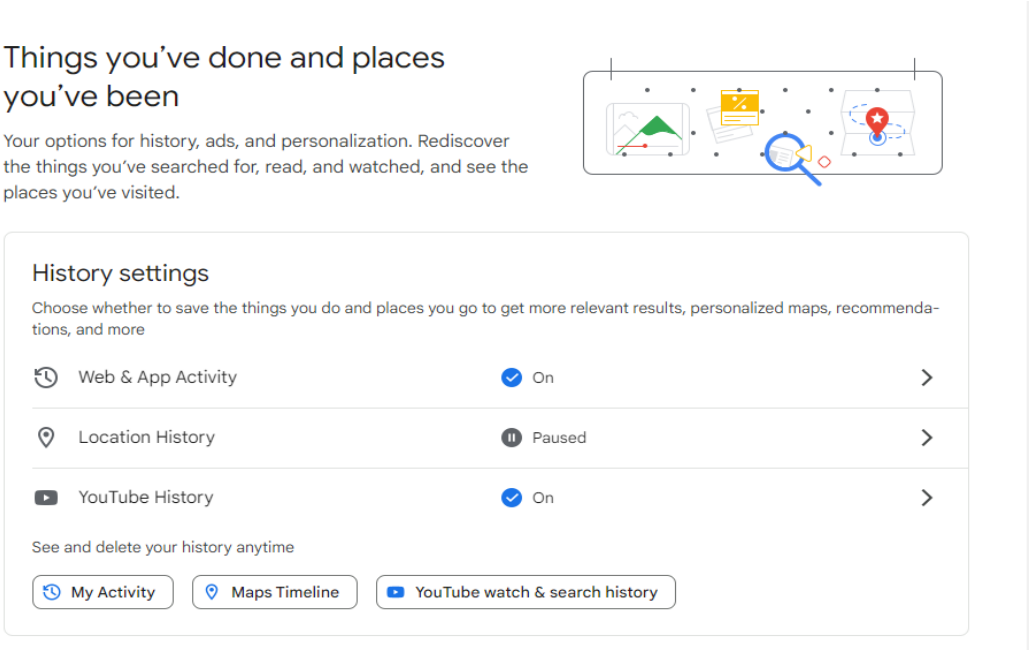


Рисунок 1.4 – Налаштування відслідковування у кабінеті Google

Інспектор конфіденційності (Google Activity Monitoring) Google дозволяє переглядати та змінювати параметри конфіденційності, пов'язані з обміном даними, персоналізацією реклами та тим, хто може бачити інформацію клієнта сервісу. Налаштування дій, які зберігатимуться у сервісі облікового запису Google. Це стосується активності в Інтернеті та додатків, історії місцезнаходжень, історії YouTube тощо. Є можливість вимкнути або ввімкнути певні елементи управління відповідно до уподобань користувача.

Дозволи та перевірка безпеки від Google допомагає ознайомитись з функціоналом захищеності системи. Існує системи дозволів на сторонні програми та веб-сайти, які використовують обліковий запис для реєстрації у цих сервісах. Відстежується місцезнаходження пристроїв на котрих використовується обліковий запис. А також, транзакції через сервіс та де вони відбулись.

Окрім, усього цього існує функціонал прав на дані, котрі зберігаються на сервері, у користувачів є можливості поділитись доступом у різних варіаціях, пряме керування ними і виключний доступ лише авторизованим користувачам. [19]

Ознайомившись функціонал більш пов’язаний з користувачами, звернемось до функціоналу системи безпеки налаштувань провайдеру, серверу, баз даних та усе що пов’язане з залізом. Досліджуваним гігантом я обрав “Amazon”. Його широкий спектр налаштувань дозволяє закрити потреби найбільш широкої аудиторії реальних спеціалістів сфери кібербезпеки. Кількість їх утиліт та проміжних сервісів настільки велика, що не дозволяє осилити кожну з них окремо одній людині..

Одним з перших пунктів, що варто зрозуміти це виділити окремі сервіси які будуть необхідні проекту. Провівши невелике дослідження я обрав такий список (зображено на таблиці 1.1).

У цьому списку є сервіси, без яких не може бути побудоване жодне з середовищ з базою даних. З матеріальної точки зору вони не такі дорогі, а сервіси Amazon, здатні вимикати їх при необхідності, заради економії, там постійно йде моніторинг трафіку, що дозволить зекономити. Окрім цього, існує додатковий сервіс, котрий впроваджує рекомендації для підтримки такого продукту, він постійно звіряє найнижчі ціни ринку та дозволяє швидко переносити сервер на інший, з метою економії.

Усі сторонні сервіси, захищені гарантією компанії Amazon, відповідають стандартам захисту систем подібного класу та регулярно перевіряються ревізорами та аудиторами.

Варто зазначити, що справжні потужні проекти, котрі використовують функціонал Amazon, мають значно більший список використаних утиліт. Це пов’язано з кількістю одночасних користувачів, безпекою, відказостійкістю системи, регіональними пропозиціями комерційно-потенціального ринку та банальною зручністю обслуговування комплексу обробки інформації. [20]

Висновком досліджуваних систем є методологія побудови зручного у користуванні сервісу для користувачів та персоналу обслуговування. Використання сучасних методів заготовлених технологій захисту та відповідності політикам безпеки типу GDPR та ISO. Створення функціоналу керування облікового запису та організації.

Таблиця 1.1 – Сервісне дослідження обраних підсистем функціоналу Amazon

Назва сервісу	Його опис	Складність реалізації
Launch Wizard	Сервіс, який допомагає вам налаштовувати, конфігурувати та розгортати корпоративні робочі навантаження відповідно до передових практик.	Низька, має зручний функціонал, котрий піднімає захищений сервіс у пару кліків. Має зручну документацію і є одним із базових технологій для проектів
IAM (Identity and Access Management)	Система керування доступів до сервісних налаштувань, операцій над ролями, політиками та ідентифікацією провайдерів	Низька, базовий функціонал керування користувачами та їх доступами до серверної частини проекту
RDS (Relational Database Service)	Сервіс реляційних баз даних, їх підняття, операції та скаляції хмарних середовищ зберігання даних	Середня, утримує у собі величезний функціонал для керування інформацією у системі.
EC2 Elastic Compute Cloud	Сервіс для безпечного та динамічного формування обчислювальних потужностей у системі	Середня, грає роль серверну частину, зв’язок між системою та користувачем. Без цього сервісу обійтись буде неможливо
SQS (Simple Queue Service)	Сервіс хмарного динамічного формування черги для обрахування внутрішніх івентів системи поза серверної частини	Складна, грає роль оптимізації та асинхронного обрахування у системі. Дозволяє розділяти комплекс на декілька робочих підланок

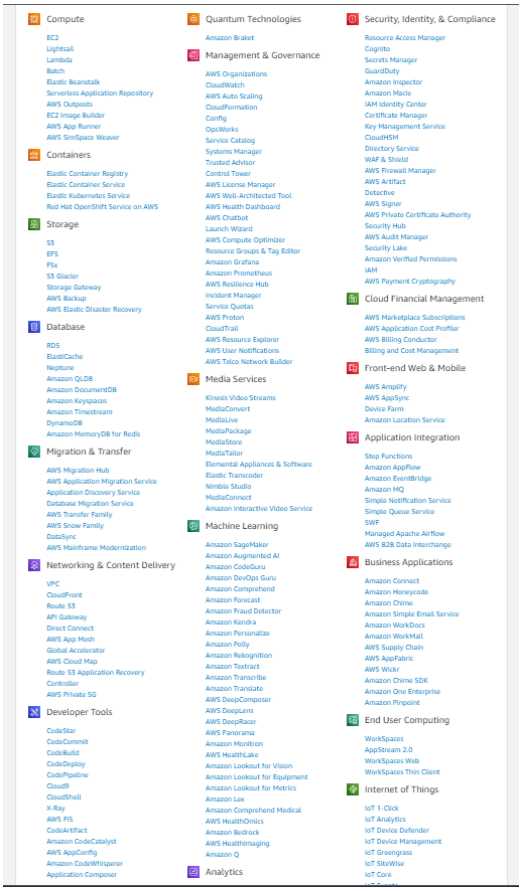


Рисунок. 1.5 – Список утиліт сервісу Amazon

1.4 Програмна складова захищеного середовища обробки інформації

У даному пункті я планую провести дослідження над технологіями програмування, що надають змогу використовувати вже існуючі методи для збору, захисту, збереження та керування даними.

Розглянемо гарний принцип у формі: «Чим менше користувач знає, як це працює, тим краще для нього.» Велика кількість систем використовує підхід AJAX для цієї мети. Ajax — це окрема мова, створена для реалізації інтерфейсів користувача та веб-додатків, де веб-сторінка не перезавантажується, а надсилає запит на сервер у фоновому режимі та отримує звідти необхідні дані для створення сторінки для користувача. AJAX є невід'ємною частиною концепції DHTML.

AJAX став темою для розмов у лютому 2005 року після публікації статті Джессі Джеймса Гаррета «Новий підхід до веб-додатків». AJAX не є окремою технологією. Ця ідея дозволяє реалізувати нові способи взаємодії серверних систем. Головною особливістю є асинхронне звернення до “беку” системи та обрахування результатів запитів без втрати керування користувачем контролю над веб-додатков. Усі завантаження відбуваються паралельно (при можливості), щоб надати юзеру зручне користування функціоналом програми.

Головним надбанням є об’єктивізація методів взаємодії UI та FrontBack складових методами XMLHttpRequest, AJAX jQuery, визначення функціонування хендлерів та відправки даних.

Об’єкт XMLHttpRequest є ядром AJAX. Він надає методи та властивості для створення HTTP-запитів до сервера та асинхронної обробки відповіді сервера. Основні кроки включають створення екземпляра об’єкта, визначення деталей запиту та визначення способу обробки відповіді. [21]

```
var xhttp = new XMLHttpRequest();
xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
        // Handle the response
        console.log(this.responseText);
    }
};
xhttp.open("GET", "example.com/data", true);
xhttp.send();
```

Рисунок. 1.6 – Приклад побудови запиту

Розробники можуть визначити функції для обробки різних етапів запиту AJAX за допомогою події onreadystatechange (для XMLHttpRequest) або за допомогою Promise і .then() (для вибірки). Це дозволяє виконувати операції, коли надсилається запит, коли сервер відповідає та коли відповідь обробляється. На запити AJAX поширюється Політика однакового походження, яка з міркувань

безпеки обмежує запити одним доменом. Запити з різних джерел можна обробляти за допомогою таких методів, як перехресне використання ресурсів (CORS) або JSONP (JSON із заповненням).

Багато сучасних JavaScript-фреймворків, таких як Angular, ReactVue.js, спрощують процес асинхронного виконання запитів і управління станом, вбудовуючи в своє ядро функціональність AJAX. AJAX - це потужна технологія для створення динамічних та адаптивних веб-додатків. Однак, впроваджуючи функціональність AJAX, розробники повинні пам'ятати про міркування безпеки, такі як запобігання міжсайтовому скриптингу (XSS) та міжсайтовому підробленню запитів (CSRF). [22]

Тепер розглянемо системну ієрархію, побудову проекту та методологію загального підходу до структурної ідеології системи. Маючи певний досвід у програмуванні, я розумію важливість швидкого реагування на конфлікти, оптимізацію та помилки у системі. Ці питання вирішуються вірною структурою коду, що дозволяє працювати над проектом іншим спеціалістам у майбутньому.

Виключенням є використання аутсорс бібліотек для використання функціоналу інших програм. До прикладу, зв’язок проекту та систем хмарного середовища збереження та обробки даних, пеймент систем, бібліотек створення UI інтерфейсу, тощо.

Звернемося до сучасних технологій побудови користувацького інтерфейсу, таких як Bootstrap, Tailwind, тощо. Я обрав Tailwind через його зручність, можливості та досвід його використання. Дана технологія являє собою передумовлений компілятор, він дозволяє автоматизовано створювати стабільний програмований інтерфейс для різних девайсів користувачів. [23]

Для реалізації технологій проекту мій погляд впав на технології PHP, версії 8.1 (останньої найстабільнішої версії). Він постійно оновлюється має у собі технології ланцюгових викликів, концепції Фібр (асинхронного підходу до програмування виконання коду в потоці управління об’єктом, без блокування попереднього), автоматичної перевірки об’єктної складової інформації, виклику функцій та загальні покращення та оптимізацію у обрахунку системи.

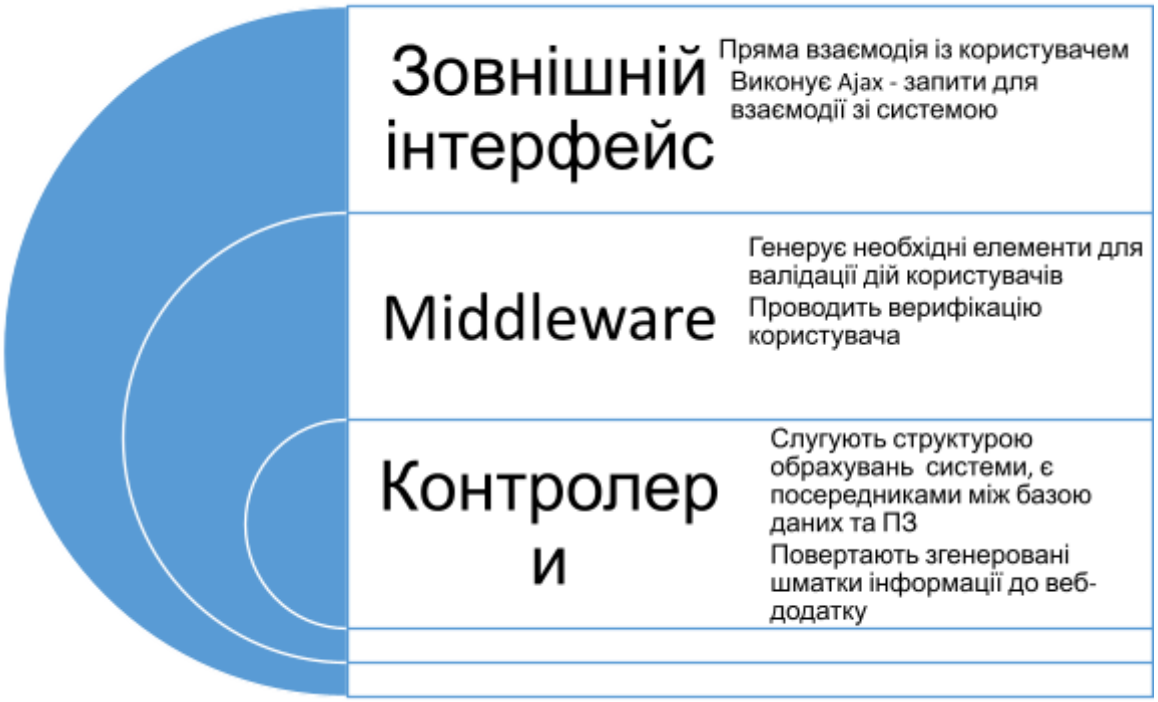


Рисунок. 1.7 – Схематичне зображення структури обробки інформації

PHP (рекурсивна абревіатура від PHP: Hypertext Preprocessor - препроцесор гіпертексту) - це широко використовувана універсальна скриптова мова уніфіційоного призначення з відкритим вихідним кодом, яка особливо добре підходить для веб-розробки і може бути вбудована в HTML. PHP можна використовувати на всіх основних операційних системах, включаючи Linux, багато варіантів Unix (включаючи HP-UX, Solaris і OpenBSD), Microsoft Windows, macOS, RISC OS і, можливо, інші. PHP також має підтримку більшості сучасних веб-серверів. Сюди входять Apache, IIS і багато інших. А також будь-який веб-сервер, який може використовувати двійкові файли PHP FastCGI, такі як lighttpd і nginx. PHP працює або як модуль, або як CGI-процесор.

Однією з найсильніших і найважливіших особливостей PHP є підтримка широкого спектру баз даних. Написати веб-сторінку з підтримкою баз даних неймовірно просто, використовуючи одне з розширень для конкретної бази даних (наприклад, для mysql), або використовуючи рівень абстракції, такий як PDO, або підключаючись до будь-якої бази даних, що підтримує стандарт Open Database

Connection, за допомогою розширення ODBC. Інші бази даних можуть використовувати cURL або сокети, наприклад, CouchDB. [24] У проекті будуть використані технології Composer, Laravel та Artisan, для реалізації методів захисту, обробки баз даних та шифрування генерованих ключів.

Composer - це менеджер пакунків прикладного рівня, призначеного для мови програмування PHP, надає стандартизований формат керування залежними частинами програмного забезпечення та необхідними бібліотеками. Його розробили Ніл Адерман та Джорді Богджано, які наразі підтримують проект. Розробка розпочалася у квітні 2011 року, а перший реліз відбувся 1 березня 2012 року. Запуск відбувається з командного рядка і завантажує залежності програми (бібліотек, тощо). Здатний встановлювати пакети плагінів та інших даних. Часто реалізується автоматизоване завантаження класів бібліотек для полегшення використання коду інших розробників. Використовується, як невід'ємна частина PHP-проектів відкритого вихідного коду. [25]

Laravel та Artisan - безкоштовні фреймворки вихідного коду, для керування апаратного забезпечення системи обробки даних та надають можливості створювати та викликати консольні команди, контролери, міграції та репозиторії. Вони значно полегшують та прискорюють розробку та тестування додатків. Всі бібліотеки безкоштовні та мають MVC подібну модульну структуру, з власними менеджерами залежностей та доступу до реляційних баз даних. [26]

1.5 Постановка задачі

Дослідивши методи сучасних комплексних систем захисту розподіленого програмного середовища передачі я отримав загальне уявлення використаних технологій їх створення та покращень. Самі ж етапи слід розподілити на чіткі структуровані частини, які допоможуть окреслити етапи розробки. Саме виконання поставленої задачі включатиме у собі велику кількість рішень декількох спеціалістів та результатів досліджень.

Я виділив такі етапи, на які слід звернути увагу під час виконання поставленої задачі:

- забезпечення безпечних каналів передачі даних для захисту від несанкціонованого доступу є основним завданням.
- оскільки програмні додатки та сховища даних охоплюють різні платформи, забезпечення інтероперабельності та сумісності з одночасним дотриманням стандартів безпеки стає складним завданням. Створення безперебійної та безпечної системи зв'язку між різними системами має вирішальне значення.
- гарантія цілісності та автентичності даних, що передаються, має важливе значення для запобігання фальсифікації та несанкціонованим змінам під час процесу передачі. Впровадження механізмів перевірки цілісності даних під час передачі є значним викликом.
- ландшафт загроз постійно розвивається, регулярно з'являються нові та витончені кіберзагрози. Інтегрована система захисту повинна бути адаптивною і проактивною, здатною виявляти і пом'якшувати загрози, що розвиваються, для забезпечення постійної безпеки.
- дотримання норм і стандартів захисту даних є постійним викликом, особливо в розподілених середовищах, де дані можуть перетинати міжнародні кордони. Дотримання нормативних вимог і забезпечення конфіденційності даних є критично важливим аспектом системи захисту.
- система повинна бути масштабованою, щоб відповідати зростаючим обсягам передачі даних в розподілених середовищах, зберігаючи при цьому оптимальну продуктивність. Баланс між заходами безпеки та необхідністю ефективної передачі даних є значним технічним викликом.
- перевірка ідентичності користувачів і забезпечення контролю доступу в розподіленому середовищі може бути складним завданням. Створення надійних механізмів автентифікації користувачів і детального контролю доступу є важливими компонентами інтегрованої системи захисту.

Велику кількість захисних механізмів слід реалізувати вже готовими та перевіреними методами чи бібліотеками, котрі включатимуть у себе такі можливості. Подібний підхід надасть гарантії надійності методів та значно пришвидшить розробку програмного забезпечення. Окрім цього, воно надасть постійні оновлення алгоритмів безпеки та оптимізації середовища, без постійного її моніторингу та внесення змін людиною, для її підтримки.

Висновок, потрібно використовувати сучасні технології у розробці комплексного програмованого середовища з останніми пакетами оновлень для гарантування безпеки додатку, його зручності та най оптимізованого функціоналу.

2 ПЕРЕДПРОЕКТНЕ ПЛАНУВАННЯ ТА ОБЛАШТОВУВАННЯ СКЛАДОВИХ ПРОЕКТУ

2.1 Створення організаційних залежностей системи

У даному розділі необхідно розробити структуроване планування розробки та функціоналу у системі. Розглянути залежності керування обліковим записом та залежними даними користувачів.

Необхідним є розгляд рівня конфіденційності передачі інформації. Моє планування передбачає розподіл інформації за грифами доступу на рівні. У кожного користувача будуть свої можливості. Якщо у власника організації буде максимальний доступ, то у звичайного незареєстрованого гостя він буде мінімальним.

Для цього побудуємо таблицю розподілу доступності інформації, що має у собі певний рівень конфіденційності. Розподіл зроблено за класифікацією - К (критична), В (вільного доступу), Т (таємна), ЦТ (цілком таємна). Кожен тип інформації, при її пошкодженні чи втраті буде нести абстраговані матеріальні втрати для компанії,

Користування такою інформацією матиме дозволи (Permissions) для кожного з користувачів, їх керування доступністю інформації, передачі доступності та типів інформації.

Відповідно до грифу доступності буде розподіл по типу акаунтів користувача. Власник - матиме повний доступ до інформації, включаючи Критичну та Цілком Таємну. Адміністратори - окремі користувачі акаунтів, які матимуть подібні права до власника. Менеджери - мають прямі права на користування та редагування критичного рівня. Для інших буде виключно доступна виключно відкрита інформація та функціонал платформи.

Сама ж система буде розпланована у майбутньому на підрівні, кожен рівень матиме свої особливості в обслуговуванні та імплементованого функціоналу.

Впровадження системи ролей дозволить планувати розробку поетапно та надасть вибірковий захищений функціонал по потребі користувачів.

Задля безпеки та розуміння системи буде створена довідникова база з коротким описом роботи та можливостей

Таблиця 2.1 – Побудова рівня конфіденційності

№ п\п	Інформація	Гриф доступу
1	2	3
1	Ідентифікатори та паролі системного адміністратора та інших осіб, що мають доступ до управління	К
2	Ідентифікатори та паролі користувачів робочих станцій	Т
3	Клієнтська база та їх конфіденційна приватна інформація	К
4	Відомості стосовно діяльності клієнтів	К
5	Відомості про постачальників	К
6	Відомості про організацію та технічні засоби реалізації основних задач компанії	К
7	Методи шифрування персональних даних, кешування одиниці збереження інформації, що є у користуванні внутрішніми сервісами, обробкою даних, методи зв’язку між серверами	ЦТ
8	Інформація, що стосується особи працівника	К
9	Відомості про зміст бази даних захисту (облікові записи, паролі користувачів зі всіма атрибутами захисту)	Т
10	Технічні заходи щодо захисту конфіденційної інформації	ЦТ
11	Способи утворення сторонніх запитів, що регламентовані системою	ЦТ

Кінець таблиці 2.1

1	2	3
12	Токени для інформаційного обміну між системою та ботом Телеграм	ЦТ
13	Методи опрацювання токенів всередині системи, що були створені сторонніми засобами, такими як Sanctum, Fortify	Т
14	Кількість та методи функціонування запитів до контролерів, їх необхідні змінні	ЦТ
15	Доповідні записки, довідки, інформаційні листи, методичні рекомендації з питань збереження конфіденційної інформації	К
16	Облікова картка користувача	К
17	Адреси користувачів	К
18	Ознайомча інформація про співробітників та компанію	В

Доступ цілком таємної інформації буде виключним для персоналу, котрі працюють з системою - це девелопери системи. Власнику не є необхідним розуміти повний функціонал системи, для нього достатньо розуміти її функціонал.

2.2 Початкові налаштування програмованого середовища

У даному розділі будуть проведені передпроектні налаштування для роботи з програмованим середовищем PHPStorm. Будуть завантажені необхідні

бібліотеки та пакети для роботи з різними існуючими утилітами подібних проектів.

Першим етапом буде створення пустого Laravel проекту. Я буду використовувати OpenServer для локальної копії проекту, для цього запустимо певні команди у папці доменів.

Open Server Panel - безкоштовне програмне середовище, розроблене спеціально для потреб веб-розробників. Інструмент необхідний для створення, налагодження та тестування власних сайтів на локальному комп'ютері.

Використовуючи цю програму можна зробити аналог Linux серверів під Windows і запускати сайти, написані, наприклад, на PHP. Компоненти, представлені в цій платформі, допомагають у роботі з графікою, редагуванні коду, тексту, архівуванні.

Особливості Open Server Panel - великий набір інструментів для створення сайтів, наявність сервера FTP FileZilla, можливість створення домену шляхом створення звичайної папки, наявність вбудованого графічного і текстового редакторів, інтегровані HTTP, MySQL, PHP модулі, вбудований менеджер закачувань можливість одночасної роботи з Denwer і Xampp наявність підтримки доменів, захист сервера від зовнішнього доступу за допомогою логіна і пароля, поширюється на безкоштовній основі, зручний користувацький інтерфейс, мінімальні системні вимоги. [27]

Після цього приступимо до конфігурації Composer у пустому проекті Laravel. Для цього потрібно встановити необхідну програму з офіційного сайту. Створюємо відповідну конфігурацію та запускаємо команду composer update. Після установки, ми отримаємо головне вікно проекту Laravel з базовим функціоналом проекту та початковими міграціями системи.

Даний проект матиме початкові конфігурації, моделі, блейди та інше (приклад конфігурації на рисунку 2.1). Вони будуть модифікуватись по ходу роботи над проектом, відповідно до необхідності.

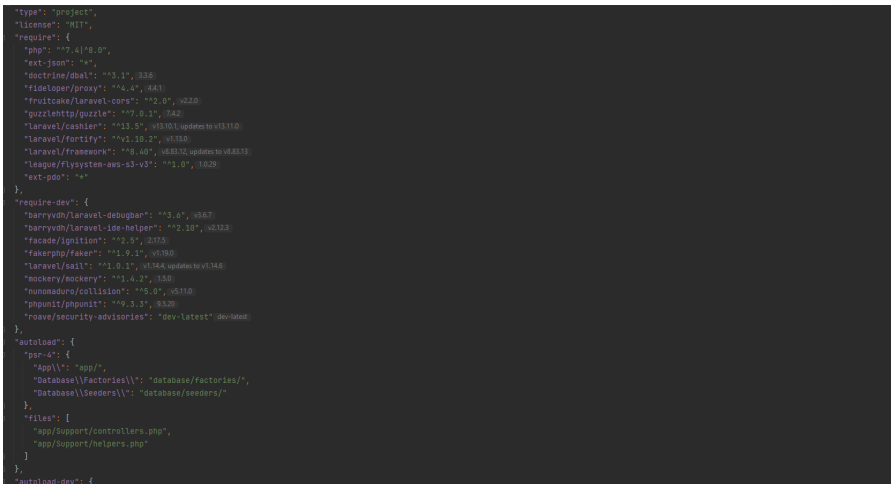


Рисунок 2.1 – Зовнішній вигляд конфігурації composer.json

2.3 Планування інтегрованих систем та попереднє тестування

У даному розділі я планую розробити методи захисту системи та підключення аутсорс утиліт, провести створення серверу та зробити відповідні налаштування утиліт.

Розпочнемо з Amazon, як було оговорено у 1 розділі, його функціонал надзвичайно широкий. Я працюватиму з 3 основними з них - EC2, SQS та RDS. Я планую розділи їх на окремі частини, для створення відказостійкого зв’язку та за малі кошти отримати досить великі потужності. Зрозуміло, що даний функціонал не безкоштовний, проте мінімальний пакет дозволяє отримати високі потужності.

Розпочнемо зі створення та запуску сервера. Для цього скористаємося EC2 механізмами. Я обрав програмоване середовище на базі Ubuntu версії 22.04 LTS, налаштував t2.micro систему та під’єднав EBS (Elastic Block Store) систему на 30GB, завдяки безкоштовному пробному періоду.

Створюємо RSA ключ згідно регіону сервера, така система дозволить уникнути небажаних сторонніх підключень до серверу.

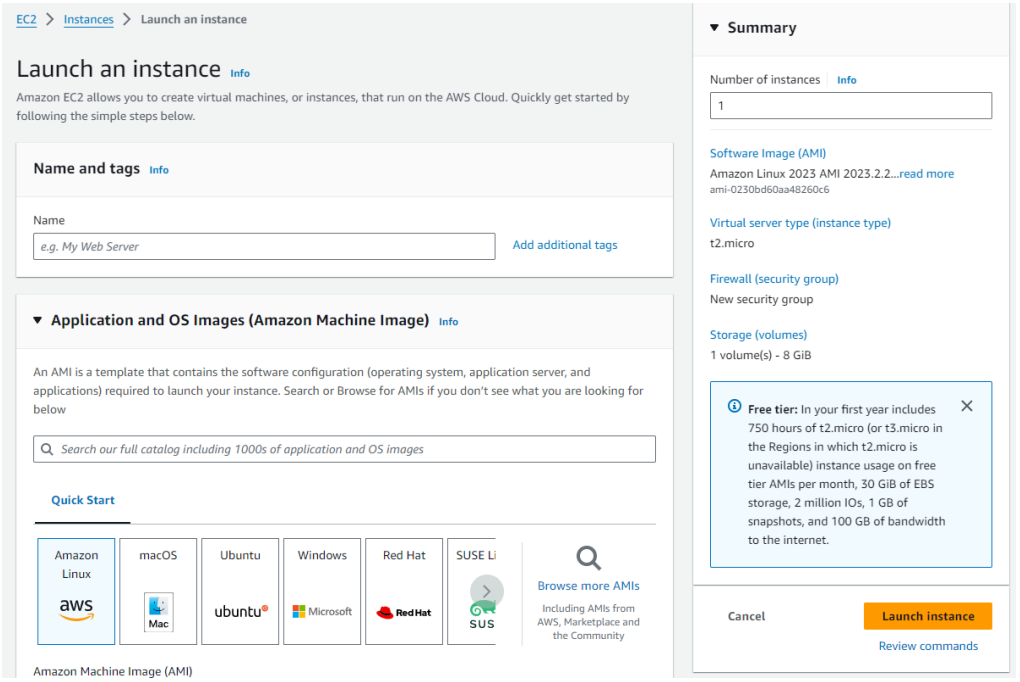


Рисунок 2.2 – Зовнішній вигляд конфігурації EC2

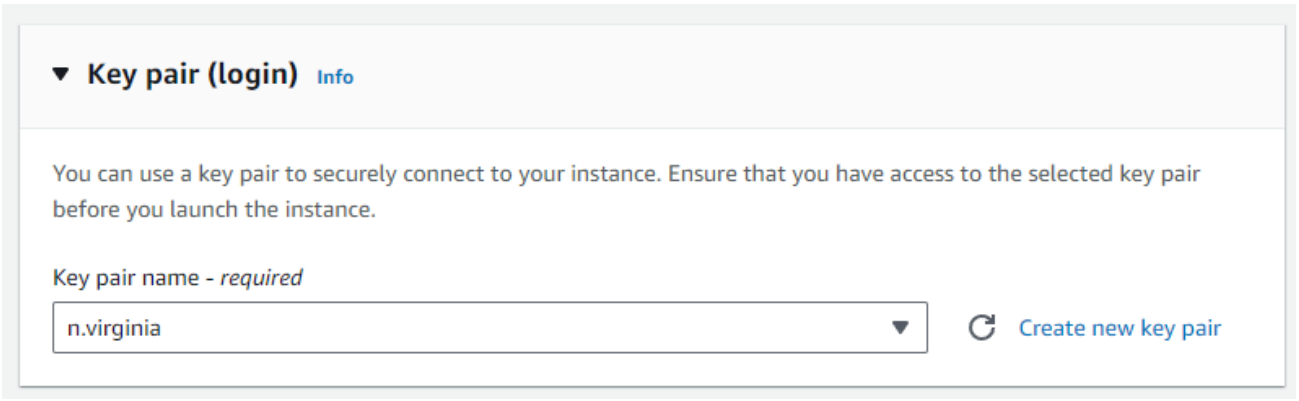


Рисунок 2.3 – Створення криптографічного ключа
для закритого зв’язку

Робимо наступні налаштування пов'язані з політикою безпеки групи користувачів серверу, задаємо мінімальні розміри сховища та операційних потужностей. Оскільки розробка планується на веб додаток відкритого доступу, я надаю дозвіл для кожного підключення до сервера, однак буду блокувати небажані підключення додаючи їх до чорного листа.

Рисунок 2.4 – Успішне підняття серверу

Наступним етапом я планую створити свою систему підтвердження дії користувача. Хоча, я й планую використовувати функціонал CSP (Content - Security - Policy) та його підсилення не буде зайвим, тим паче, воно не буде забирати на себе великої кількості навантажень.

Даний метод називається “Action Token”. Механізм буде слугувати базовим одноразовим валідатором дії користувача у системі.

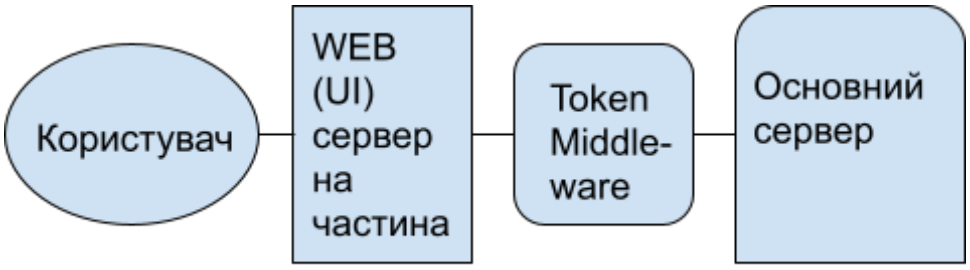


Рисунок 2.5 – Схематичне зображення Action Token

Основа даного методу полягає у генерації тимчасового коду на кожну дію користувача та видалення його при закритті вікна дії, при помилці дії чи поверненні успішного результату. Така можливість надасть гарантувати, що кнопку було натиснуто саме нашим, живим користувачем у обраному ним вікні. Та не дозволить зловмисникам виконувати дії від його імені, у разі несанкціонованого доступу до платформи. Генерація ключа буде відбуватись на основі алгоритму хешу, унікальних даних користувача та долі часу, у яку відбувався запит.

Займемось розподілом функцій даного комплексу. Розділимо їх на 3 етапи:

1) Багаторівнева робота програмованого комплексу шляхом підключення декількох рівнів утиліт на базі Amazon

2) API функціонал з доступом до програмного середовища без необхідності у користуванні WEB частиною

3) Система сповіщень, отримання повідомлень, збереження повідомлень аутсорс сервісом.

Основною ідеєю розподілу серверних складових є користування EC2 та SQS налаштувань, що захистять сервіс від DDoS атак, його буде легше “захищати” через єдиний щільний зв’язок доступу, а також відказостійкість. Я повторю дії, для підняття іншого сервера, який відповідатиме виключно за систему обробки Ajax запитів, інший сервер матиме прямий доступ до бази даних, оброблятиме запити та відправляють їх на фронт. Система SQS буде крайнім випадком, нажаль вона недешева. І не хотілось би витрачати таку кількість грошей без конкретної необхідності. Для цього створимо окреме налаштування у файлі конфігурацій Laravel.

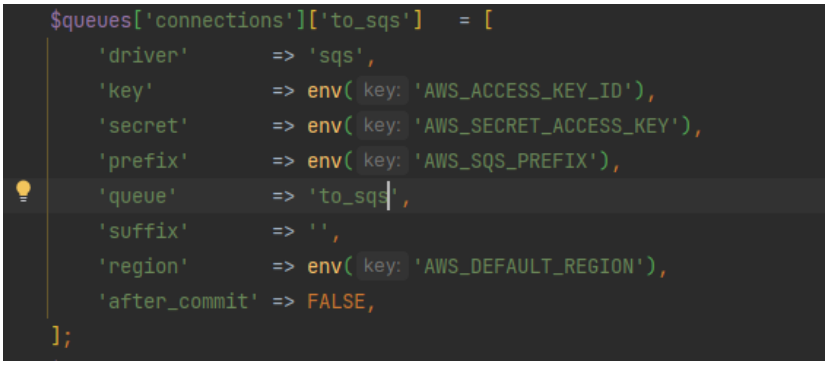


Рисунок 2.6 – Вигляд конфігурації направлення трафіку на SQS

Такі налаштування дозволять відслідковувати перевантаження системи та перенаправляти івенти до SQS черги, що дозволить розвантажити сервер на досить тривалий проміжок часу.

Взагалі, Amazon Simple Storage Service (Amazon S3) — це широко використовувана служба зберігання об’єктів, яку надає Amazon Web Services (AWS).

Він забезпечує масштабоване, довговічне та безпечне сховище для різних типів даних, таких як файли, зображення, відео та резервні копії. Amazon S3 розроблено для високої надійності та доступності, що робить його популярним вибором для компаній і розробників для зберігання та доступу до даних через Інтернет.

Основними функціями Amazon S3 виділяють - масштабованість, він може масштабуватися для розміщення практично необмежених обсягів даних. Він призначений для роботи з великими обсягами даних і підходить як для невеликих програм, так і для рішень корпоративного рівня.

Довговічність і надійність: S3 зберігає дані з резервуванням між об’єктами та пристроями в регіонах AWS. Досягає високої міцності та надійності.

Ця послуга розроблена, щоб гарантувати термін служби 99,999999999% (11 дев'яток) об'єктів протягом року.

Безпека: Amazon S3 забезпечує багаторівневий захист даних у стані спокою. Механізми контролю доступу, такі як списки контролю доступу (ACL) і політики сегментів, дозволяють користувачам контролювати, хто може отримати доступ до їхніх даних.

Крім того, використовувати AWS Identity and Access Management (IAM) досить зручно для керування доступом користувачів до ресурсів S3.

Керування життєвим циклом даних S3 дозволяє користувачам визначати політики життєвого циклу для автоматичного переміщення об’єктів між класами зберігання та видалення об’єктів, які більше не потрібні. Це допомагає оптимізувати витрати, переміщуючи дані, до яких рідко звертаються, у менш дорогі класи зберігання.

Керування версіями Amazon S3 дозволяє користувачам зберігати кілька версій об’єкта в одному сегменті. Це корисно для відстеження змін об’єкта з часом і для відновлення після випадкового видалення або перезапису.

Data Transfer Acceleration: S3 Transfer Acceleration використовує глобальну мережу доставки вмісту Amazon CloudFront (CDN) для прискорення завантаження та завантаження об’єктів. Це особливо корисно для програм, які мають користувачів по всьому світу.

Завантаження блоками: є здатність використовувати багатокомпонентне завантаження, щоб завантажувати великі об’єкти блоками, щоб покращити продуктивність і стійкість, якщо під час завантаження виникають проблеми з мережею.

Окрім цього, Amazon S3 підтримує сповіщення про події, які дозволяють користувачам налаштовувати тригери для певних подій (таких як створення або видалення об’єкта). Ці події можна використовувати для автоматизації робочих процесів та інтеграції з іншими службами AWS.

Amazon S3 широко використовується для різноманітних цілей, зокрема для резервного копіювання даних, розповсюдження вмісту, розміщення статичного веб-сайту та внутрішньої пам’яті для програм і служб, що працюють на AWS.

Розрахункова модель ціноутворення забезпечує простий і доступний спосіб зберігання та доступу до ваших даних. [28]

Фрагмент коду перевірки перевантаження сервісу та перенаправлення черги на SQS

```
try {
                dispatch(
                        new Event(
                                $event->getId(),
                                $event->getToken(),
                                $event->GetMessage(),
                                $event->getDate()
                        )
                )->onConnection('jobs_' . sprintf('%02d',
systemController()->isOverloaded() ? '__jobs' : 'to_sqs'));
        } catch (Throwable $e) {
```

```
        echo $e->getMessage() . PHP_EOL;
    }
```

Тепер повернемось до питанні API. Основна ідея полягає в тому, щоб дозволити користувачам використовувати базу даних поза програмним середовищем. То з цією метою система запитів API була реалізована Laravel Sanctum service. І у майбутньому він буде перевірений стороннім програмним середовищем POSTMan.

PostMan - це платформа для створення та використання API. PostMan надзвичайно зручний та дозволяє тестувати систему максимально до реально циклу. API оптимізує співпрацю користувача та платформи, щоб клієнти моги модифікувати функціонал під власні потреби. Платформа PostMan включає повний набір інструментів, які допоможуть прискорити розробку API. [28]

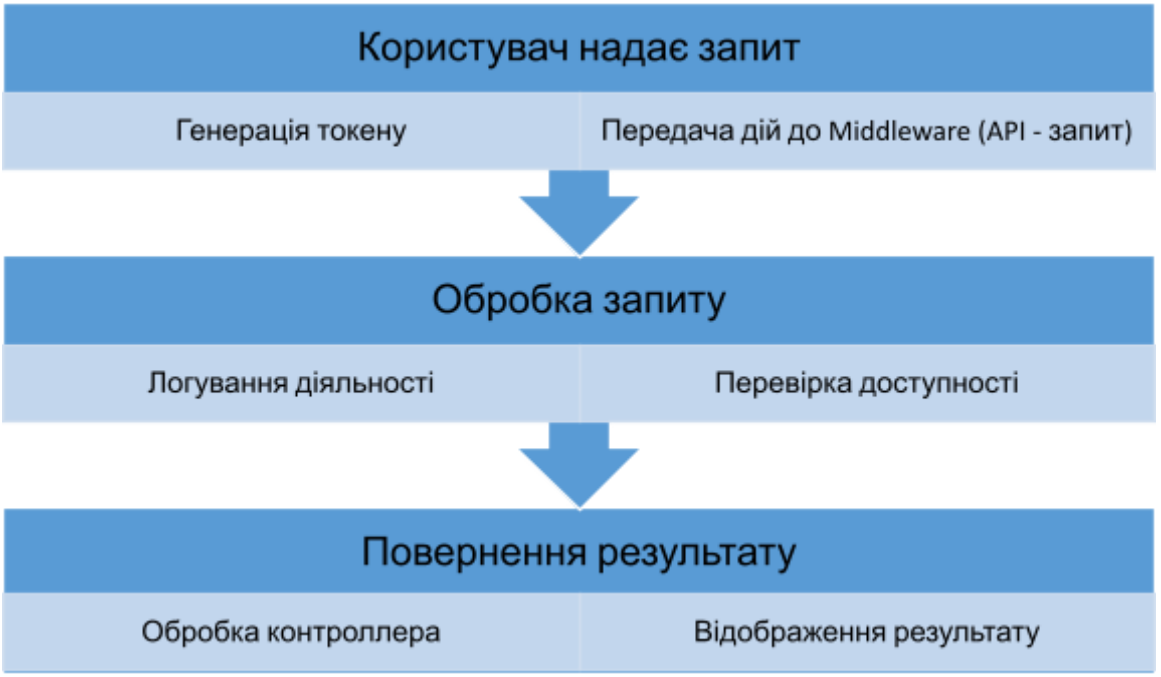


Рисунок 2.7 – Схема почергової обробки API запиту

PostMan - це платформа для створення та використання API. PostMan надзвичайно зручний та дозволяє тестувати систему максимально до реально циклу. API оптимізує співпрацю користувача та платформи, щоб клієнти моги

модифікувати функціонал під власні потреби. Платформа PostMan включає повний набір інструментів, які допоможуть прискорити розробку API.

Для системи нотіфікацій створимо телеграм-бота. Особливістю Телеграму є зручне API та безкоштовне необмежене користування для надсилання та отримання повідомлень.

Система генерує хеш-коди, коли користувачі створюють нові облікові записи, змінюють паролі або запрошують нових користувачів. Найкращим рішенням для цього є надсилання повідомлень через різні захищені служби, які надають такі функції. Це зручно, безпечно та дозволяє надсилати повідомлення за лічені секунди (з хорошим підключенням до Інтернету). Уже встановлена бібліотека, інтегрує базовий функціонал у проекті та дозволяє досить просто користуватись сервісом сповіщень. [29]

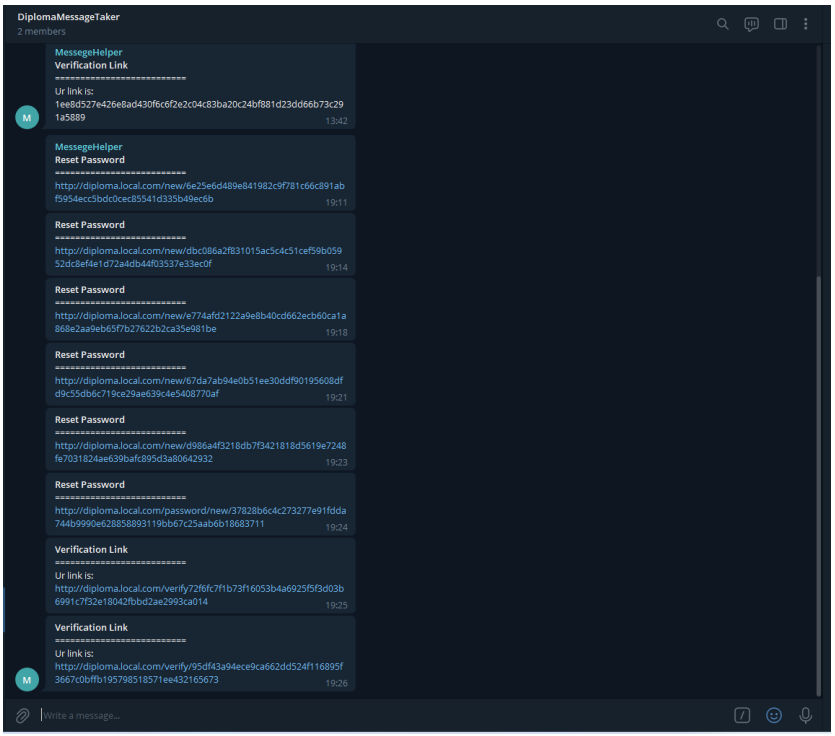


Рисунок 2.8 – Вигляд отриманих повідомлень у телеграмі після обробки подій системою

Дані сповіщення призначені не лише для клієнтів, це дозволяє програмістам швидко реагувати на неполадки у системі та приступити до їх вирішення. Даний бот матиме змогу відправляти повідомлення особисто клієнту.

Взагалі, телеграм має обмеження у обробці запитів, це пов’язане з відказостійкістю системи. Якщо ми побачимо помилку у каналі досить часто, це означатиме, що проблема є критичною і необхідна оперативне втручання до функціоналу системи. У майбутньому планується додати ще декілька телеграм ботів, для уніфікації спеціальних сповіщень до різних каналів та окремих налаштувань під кожного бота

Рисунок 2.9 – Схематичне зображення опрацювання подій у системі

У даній схемі (на рисунку 2.9) пояснено поетапне опрацювання системою сповіщень, вона є багаторівнева, оскільки сповіщення можуть користуватись не

тільки виключно Телеграмом. Після невеликих змін у коді та завантаженні бібліотек, з’явиться можливість обміну повідомленнями і з іншими месенджерами (Viber, WhatsApp, тощо).

Головною причиною обрання Телеграму стала економічна складова системи, інших захищених середовищ соціальних мереж з такими ж можливостями не було знайдено..

Повернемось до більш організаційного питання, а саме - впровадження політики безпеки у систему. Я не планую роботи і повноцінно, з розподілом повних посадових обов’язків, етапів реагувань на критичні сповіщення та розподілом дій між працівниками. Я виділю декілька основних кроків, які будуть реалізовані у даній системі.

Створення комплексної політики безпеки є важливим для захисту інформації та активів у вашій організації.

Політика безпеки — це набір інструкцій і правил, які визначають, як захищаються інформаційні системи та ресурси.

Ось загальний огляд створення політики безпеки:

1.Вступ, Мета: Чітко формулюємо мету політики безпеки. Сфера: визначимо сферу дії політики та ідентифікуємо систему, активи та посади, до яких вона буде застосована.

2. Окреслюємо основні цілі політики безпеки, зокрема: конфіденційність, цілісність і доступність інформації.

3. Визначимо обов'язки осіб або команд, залучених до підтримки та впровадження заходів безпеки. Визначимо ролі для співробітників, адміністраторів і персоналу служби безпеки.

4. Створимо детальні заходи контролю доступу, включаючи процедури автентифікації та авторизації користувачів. Опис керування обліковими записами користувачів, політику паролів і обмеження доступу.

5. Класифікуємо дані на основі чутливості та вирішуйте, як обробляти кожен рівень класифікації. Визначимо вимоги до шифрування конфіденційної інформації.

6. Безпека мережі та заходи, які необхідно вжити для захисту мережевої інфраструктури вашої організації. Включає конфігурацію брандмауера, системи виявлення/запобігання вторгненням і моніторинг мережі.

7. Endpoint Security. Детальний опис заходів безпеки для конкретного пристрою, включаючи антивірусне програмне забезпечення, виявлення кінцевої точки та рішення для реагування.

8. План реагування на інциденти, також е є необхідним, але цей пункт буде упущено з недостатньою кількістю реальних кейсів та абстрагованістю функціоналу проекту. Розробка план виявлення інцидентів безпеки, реагування та відновлення та процедури звітування про порушення безпеки.

9. Фізична безпека буде проігнорована, оскільки єдиним фізичним суб'єктом даної системи є сервери та програміт. Програміст захисту не потребує.

10. Інформація про безпеку та навчання підкреслює важливість поінформованості працівників про безпеку. Необхідно створити навчальну програму, щоб ознайомити своїх співробітників із найкращими методами безпеки.

11. Безпека постачальників і сторонніх постачальників, стандарти для оцінки та управління ризиками безпеки, пов'язаними зі сторонніми постачальниками.

Визначимо вимоги безпеки в контрактах і угодах.

12. Аудит безпеки та відповідність це рутинний аудит безпеки та процедури оцінки. Забезпечення дотримання галузевих кодексів і стандартів, що стосуються організації.

13. Документація та записи, визначимо вимоги до документації щодо процедур безпеки та інцидентів. Встановлення практики ведення записів щодо діяльності, пов'язаної з безпекою також необхідні.

14. Процес перегляду та оновлення - це процес для регулярного перегляду та оновлення політик безпеки для адаптації до мінливих загроз і технологій.

15. Забезпечення відповідності та наслідки порушення політики безпеки. Окреслимо механізми забезпечення відповідності та покарання за невиконання.

16. Додамо будь-які додаткові документи чи посилання, пов’язані з вашою політикою безпеки.

Наша політика безпека повинна бути адаптована до конкретних потреб і характеристик організації, подібної до реальних. Регулярний перегляда та оновлення будуть за необхідності політики, щоб переконатися, що вони ефективні для вирішення нових проблем безпеки.

Також важливо залучати ключових зацікавлених сторін до розробки та реалізації політики, щоб гарантувати, що політика широко розуміється та дотримується.

В результаті даного розділу були проведені підготовчі планування та налаштування для майбутньої системи обробки інформації.

Були чітко сформульовані покладені задачі для адміністративної роботи та впроваджені основні цілі у розробці документації, включена мета побудови політики безпеки та ознайомлення персоналу з функціоналом, задля збереження цілісності та конфіденційності збереження даних у системі. В цілому, даний розділ можна вважати завершеним.

3 СИНТЕЗ ЗАХИЩЕНОГО КОМПЛЕКСУ ОБРОБКИ ДАНИХ

3.1 Створення захищеної бази даних

Основна мета проектування бази даних — зменшити надлишковість збережених даних. Правильне планування бази даних забезпечує оптимальне використання операційного та дискового простору, надає доступ до зручних механізмів для модифікації даних і забезпечує високу цілісність бази даних. Якщо неправильно працювати з базою, можна отримати дублікати даних.

Система управління та моніторингу не повинна створювати проблем під час аналізу та її використання. Надмірне дублювання може призвести до подальших помилок у виведенні та організації даних. Всі копії даних необхідно контролювати. Якщо одна копія змінена, інші копії також повинні бути змінені.

Нормалізація — це ще один процес, який скорочує структури даних до стану, що забезпечує оптимальні умови для вибору, включаючи, модифікацію та видалення даних. [30]

Досягти цього можна шляхом поділу великих обсягів даних на менші об’єкти зберігання інформації. Кінцевою метою нормалізації є отримання розумного проекту інформаційного центру, який швидко й точно обробляє прямі запити та не створює проблем під час будівництва.

У теорії нормалізація відношень відноситься до формальних засобів для обмеження формування відношень (таблиць).

У даному випадку я буду використовувати MySQL, мені він добре знайомий та я маю певний досвід у його користуванні. Сервер MySQL має наступні функції:

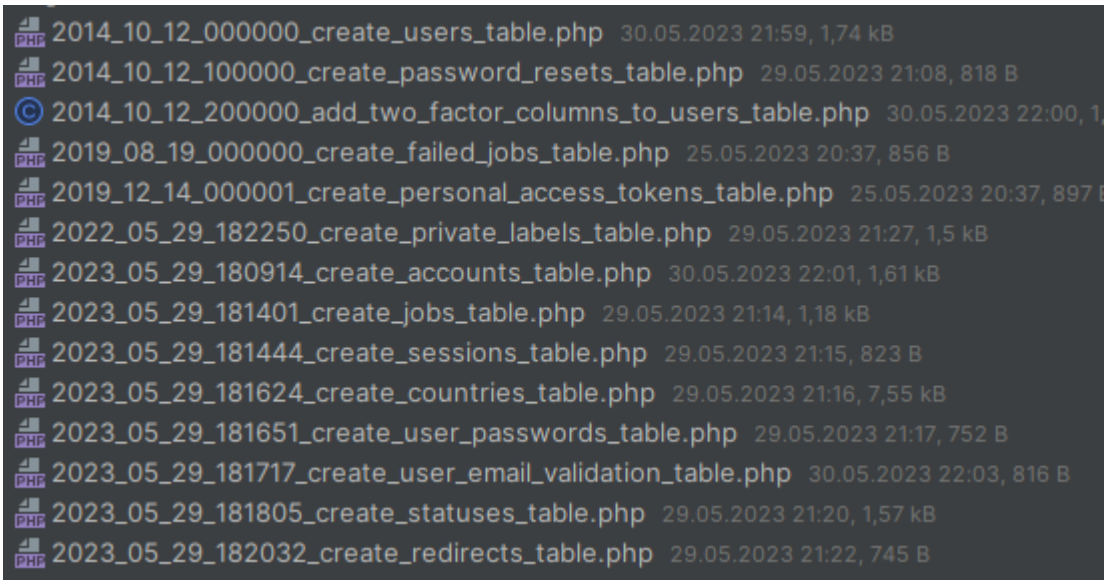


Рисунок 3.1 – Частина створених міграцій для обробки таблиць бази даних

Система керування реляційними базами даних «RDBMS». Це Структурно побудова даних функціонує не інакше, ніж модель реляційної бази даних, що дозволяє виконувати операції з даними відповідно до правил реляційної алгебри, вперше сформульованих Е. Ф. Коддом у 1970 році. На основі SQL кожна база даних має свої особливості при створенні запитів. Однак, як і більшість систем, MySQL має добре сформований формат запиту, який логічно побудований відповідно до «людської логіки» і не є складним. Адміністратори, користувачі та програмісти додатків використовують мову структурованих запитів (SQL).

Масштабованість, на додаток до попереднього пункту, є ще одним варіантом створення окремого унікального ідентифікатора, крім звичайного поля ID. Це робиться для підключення до інших баз даних або перенесення до існуючої бази даних. Також, це надає змогу зменшити виникнення колізій під час міграції даних.

Варто зауважити, що згенероване поле, як правило, буде 128-156-бітним хешем, оскільки ймовірність збігу згенерованого хешу має бути якомога меншою, щоб уникнути колізій.

Системи MySQL також мають розподілені ядра зберігання даних, які можна порівняти з віртуальними машинами. Крім того, він оснащений оптимізацією та підтримкою Laravel Builder, які дозволяють створювати користувацькі перевірені запити у вашому сховищі без надсилання запитів у форматі стрічки, що додатково зменшує пам’ять, витрачену на обробку даних. [31]

Використовується кілька механізмів збереження цілісності. Забезпечення декларативної посилальної цілісності (зв’язки таблиць) (DRI) дозволяє користувачам встановлювати обмеження даних і зв’язки між таблицями, щоб відповідати ключовим словам таблиці, у багато разів скорочуючи необхідні обчислення. Це також необхідно для узгодження правил зберігання або посилальної цілісності таблиці.

Для роботи з базою даних я використовую модельну архітектуру, контролери та репозиторії, подібна сегментація коду є досить зручною, для ефективної роботи.

Основні операції виконуються мовою Builder Eloquent. Laravel Query Builder надає зручний і простий для розуміння інтерфейс для створення та виконання запитів до бази даних. Це дозволяє вам взаємодіяти з базою даних без написання необроблених запитів SQL. Eloquent — це ORM, який зіставляє таблиці бази даних з об’єктами, тоді як Query Builder — це більш прямий спосіб взаємодії з базою даних за допомогою гнучкого синтаксису. Можна використовувати будь-який інструмент залежно від уподобань і потреб проекту.

3.2 Імплементація методів захисту для Laravel

У даному розділі я буду працювати над логікою, середовищем, переналаштуванням вже завантажених базових методів генераторів платформ та вноситиму свої покращення.

Оскільки функціонал програми не був зазначений, я вирішив його обмежити.. Це пов’язано з термінами, будь-яку програму можна покращувати

роками, у мене цього часу немає. Тому, я проводжу паралель, усі користувачі будуть мати можливість обміну повідомленнями, ми їх розподілимо на рівні, відповідно до грифу доступу кожного з користувачів. Laravel вже має імплементовані генератори моделей, котрі допомагають пришвидшити роботу, скориставшись ними ми отримали такий список моделей програми.

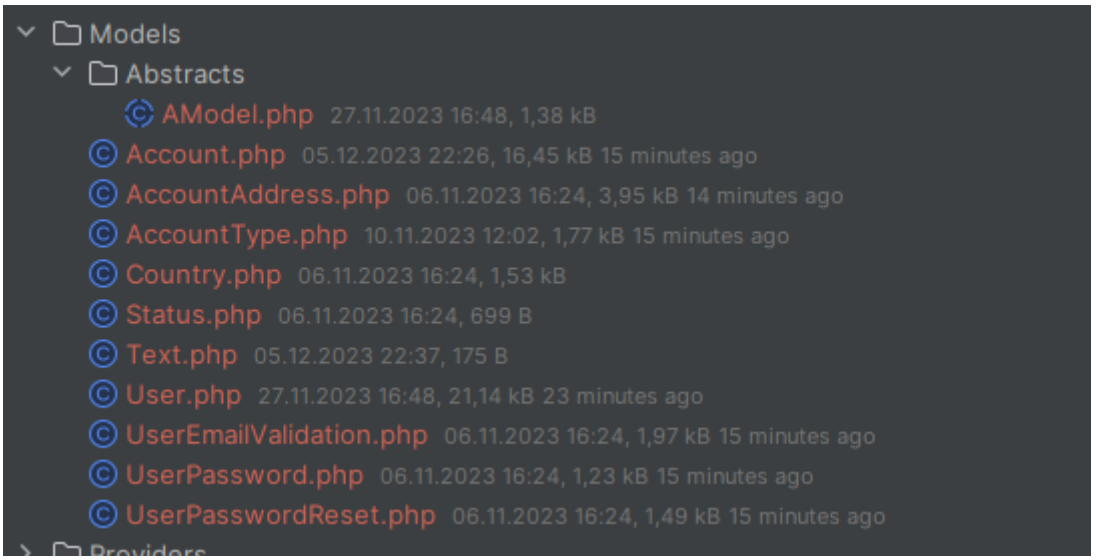


Рисунок 3.2 – Моделі програми

Кожен клас має власні поля, взаєпов’язаність з іншими та має окремі таблиці для збереження даних.

Після створення моделей, аналогічний способом створюємо контролер та репозиторій під кожний з цих класів.

Облаштовуючи налаштування безпеки я вирішив додати велику кількість функціоналу з обмеження дій користувачів.

Я виділив окремі пункти необхідні для цього:

1) Налаштування файлу .htaccess на базі NGIX конфігурацій. У даному випадку я планую працювати з масивом бази, що дозволяє безпосередньо блокувати користувачів системи у різних випадках.

2) Встановив rate_limiter. Це спеціальна функція на базі Laravel, надає змогу блокувати завелику кількість звернень за одиницю часу.

3) Налаштував менеджмент сесій системи.

4) Створив методи валідації вхідних та вихідних даних. Додав відповідні перехоплення системою у результаті помилок.

5) Створив систему логування у системі пов’язану з редагуванням даних користувачів, акаунту чи моделей.

6) Створив ліміти від атак методів brute force.

7) Ввів виключну можливість звернень за обраними рутами і ніяк інакше

8) Система має кастомізовану логіку звернень, відповідей і інших механізмів сервісу, що ускладнить будь-яку атаку на сервіс. [32]

Контроль доступу: Обмежено доступ до інформації на основі ролі та відповідальності, щоб запобігти несанкціонованому доступу.

```php
switch ($routeName) {
    case RouteName::ACTION_ADMIN_PERMISSIONS:
        return self::ADMIN_EDIT;
    case RouteName::ACTION_ACCOUNT_NEW_CUSTOMER:
        return self::CUSTOMER_ADD;
    case RouteName::PAGE_PARTNERS:
    case RouteName::CONTENT_CUSTOMERS:
    case RouteName::PAGE_ACCOUNT_USERS:
    case RouteName::PAGE_ACCOUNT_ACTION_LOG:
    case RouteName::PAGE_ACCOUNT_ENTITY_LOG:
    case RouteName::POPUP_ACCOUNT_PARTNER_DETAILS:
        return self::CUSTOMER_VIEW;
    case RouteName::POPUP_CUSTOMER_CREATE:
        return self::CUSTOMER_EDIT;
    case RouteName::CONTENT_USERS:
    case RouteName::PAGE_USERS:
    case RouteName::POPUP_USER_DETAILS:
        return $authAccount->account_type->getPrefix() . '_user_view';
    case RouteName::ACTION_USER_INVITE:
        return $authAccount->account_type->getPrefix() . '_user_add';
    case RouteName::ACTION_USER_PERMISSIONS:
```

Рисунок 3.3 – Фрагмент допусків по типу акаунту

Шифрування: Захист даних за допомогою шифрування робить їх нечитабельними навіть у разі отримання несанкціонованого доступу без відповідного ключа дешифрування.

Автентифікація та авторизація: перевірка особи користувача та надання відповідного рівня доступу залежно від його ролі.

Політика та процедури безпеки: встановлення та впровадження політики та процедур, що регулюють обробку, зберігання та передачу конфіденційної інформації.

Проінформованість про безпеку: створимо “культуру” безпеки, навчаючи співробітників найкращим практикам безпеки та потенційним загрозам.

Реагування на інциденти та планування аварійного відновлення: Створюємо механізми на реакції на інциденти безпеки та витоки даних, щоб мінімізувати шкоду та забезпечити швидке відновлення.

3.3 Створення користувацького інтерфейсу

Інтерфейс користувача одна з найважливіших частин для користувача, він повинен бути оптимізованим, нативно зрозумілим та зручним. Мій вибір впав на TailWind, це зручна скриптова бібліотека генератор, подібна до інших бібліотек такого класу. У ньому завантажена приємна палітра кольорів, маштабовані розміри і що найголовніше, є бібліотека з готовими фрагментами елементів, що пришвидшують роботу з побудовою інтерфейсу.

Програмний інтерфейс користувача є важливою частиною зв’язку між людиною та програмним середовищем. Необхідно розробити його таким чином, щоб він був зручним у користуванні, нативно зрозумілим, мав приємний зовнішній вигляд та, головне, оптимізованим.

Сучасні бібліотеки мають широкий функціонал елементів, допоміжних засобів, текстових полів та селекторів, з вже створеними унітарними стилями, що дозволяє сегментарно будувати загальний дизайн та полегшувати етапи розробки та їх покращення.

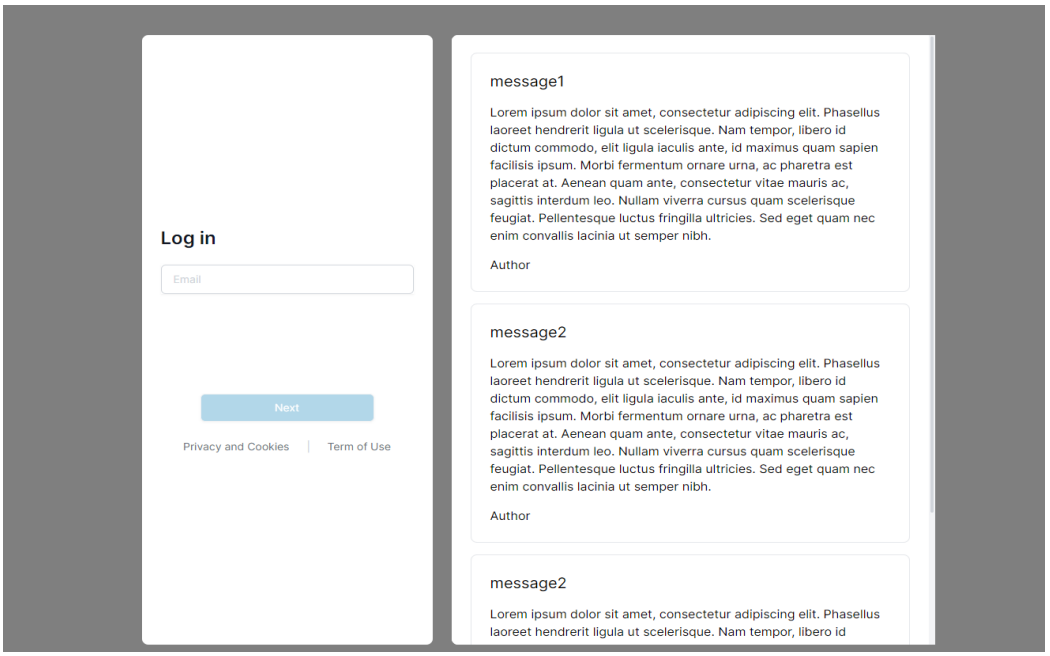


Рисунок 3.4 – Екземпляр інтерфейсу на сторінці логування

На даному рисунку (див. 3.3.1) - показано фрагмент сторінки логування з відповідними полями, а також вікном відкритої інформації з системи. У правому фрагменті власники акаунтів здатні залишати блоги, статі, новини, тощо. Це буде єдиним доступним функціоналом для незареєстрованих відвідувачів.

3.4 Реалізація базових методів безпеки користувача

Sanctum надає простий спосіб автентифікації односторінкових програм (SPA), які повинні взаємодіяти з API, запущеними на Laravel.

Ці SPA можуть знаходитися в тому самому сховищі, що й ваша програма Laravel, або вони можуть бути повністю окремими репозиторіями.

Наприклад, SPA, створений за допомогою програм Vue CLI або Next.js. Sanctum не використовує жетони для цієї функції. Натомість Sanctum використовує вбудовану службу автентифікації сеансу на основі файлів cookie Laravel. Для цього Sanctum зазвичай використовує захист веб-автентифікації Laravel.

Забезпечує захист від CSRF, автентифікації сесії, а також захищає від витоку облікових даних через XSS.

Sanctum намагається автентифікуватися за допомогою файлів cookie, лише якщо вхідний запит надходить із його власного інтерфейсу SPA. Коли Sanctum обробляє вхідний HTTP-запит, він спочатку перевіряє наявність файлу cookie автентифікації. Якщо його немає, Sanctum перевіряє заголовок авторизації на дійсний маркер API. [33]

Хешування 256 — це багатоетапний тип шифрування даних за методом Меркле-Демгарда. Оскільки, ми зберігаємо великі обсяги даних користувачів, ми повинні писати унікальний код для кожної дії та шифрувати конфіденційну інформацію (втручання в базу даних не дає зловмиснику жодної інформації; ніхто, крім самої програми, не розуміє інформації).[34]

Вцілому, API це незначимий функціонал, він необхідний у окремих випадках. Коли наші користувачі хочуть отримати функціонал платформи без візуальної складової, або коли ми втрачаємо доступ за стандартними методи входу, у нас з’являється альтернатива для дій. Якраз і другий пункт надає змогу розподілити функціонал проміжними методами та отримати доступ.

```php
<?php
Route::group(['middleware' => ['api_auth']], static function() {

});
Route::group(['middleware' => ['auth:sanctum']], function() {
    //Private Routes
    Route::get( uri: '/testRequest', action: 'Api\SanctumApiController@testRequest');
    Route::post( uri: '/logout', action: 'Api\SanctumApiController@logout');
});
//Public Routes
Route::post( uri: '/register', action: 'Api\SanctumApiController@register');
Route::post( uri: '/registerComplete', action: 'Api\SanctumApiController@registerComplete');
Route::post( uri: '/forgotPassword', action: 'Api\SanctumApiController@forgotPassword');
Route::post( uri: '/resetPassword', action: 'Api\SanctumApiController@resetPassword');
Route::post( uri: '/login', action: 'Api\SanctumApiController@login');
```

Рисунок 3.5 – Реалізовані API методи

Були реалізовані методи необхідні для дій користувача. У випадку несанкціонованого входу, до його облікового запису та втрати паролю, маючи

згенерований хеш-код, який відіграє роль унікального секретного ключа, він буде здатним відновити пароль, вийти з акаунту, тощо.

Займемось функціоналом 2MFA - двофакторна аутентифікація. Fortify — це система білінгу та аутентифікації користувачів, рекомендована самим сервісом Laravel. [35] Його відмінною рисою є наявність повноцінної системи аутентифікації, яка автоматично інтегрується в проект, а також шифрування та моніторинг входів у систему.

Оскільки сама система Artisan може генерувати (і редагувати) стандартні міграції, деякі поля для двофакторного входу до облікового запису додаються до основної моделі користувача.

Недолік цієї системи полягає в тому, що система користувача недостатньо підготовлена, оскільки немає підтвердження того, що користувач дійсно зберіг створені ним шифри.

Тож після внесення деяких змін до основних класів використання я отримав таке рішення: Ця система використовує хеш, подібний до системи SHA-1, і створює 160-бітний алгоритм генерації випадкових чисел.

Коли ви скануєте QR-код, ви отримуватимете 6-значне випадкове число кожні 30 секунд.

Якщо у користувачів виникають проблеми з зчитуванням QR-коду, для отримання генератора, вони можуть завантажити окремі плагіни чи аплікації з підтримкою такого функціоналу.

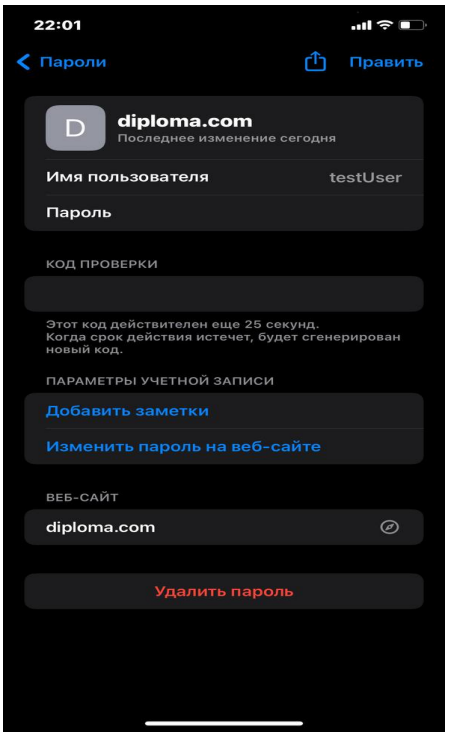


Рисунок 3.6 – Збережений 2ФА код у пристрої на базі IOC
(неможливо показати код на зображенні, через політику безпеки системи)

Окрім автоматично згенерованих кодів, алгоритм впровадить 10 невідновних кодів, які людина може використати для зміни налаштувань 2ФА. Я додам метод зкидання 2ФА виключно власникам акаунтів та адміністраторів, для зручності. Згідно мого досвіду, не всі люди зберігають коди відновлень..

3.5 Реалізація методу Action Token

Основна ідея даного методу полягає у генерації тимчасового хеш-коду на кожен запит на дію користувача. При відкритті модульного вікна, повертається масив даних, що динамічно формує вікно дії, називається він popup. Після валідації на фронті, перед перевіркою дії йде запит до основного сервера, він звіряє чи є хеш і чи він відповідає хешу збереженого у сесії. Хеш з фронту буде відправлятись виключно нашими скриптами, що не дозволить емулювати діяльність живого користувача.

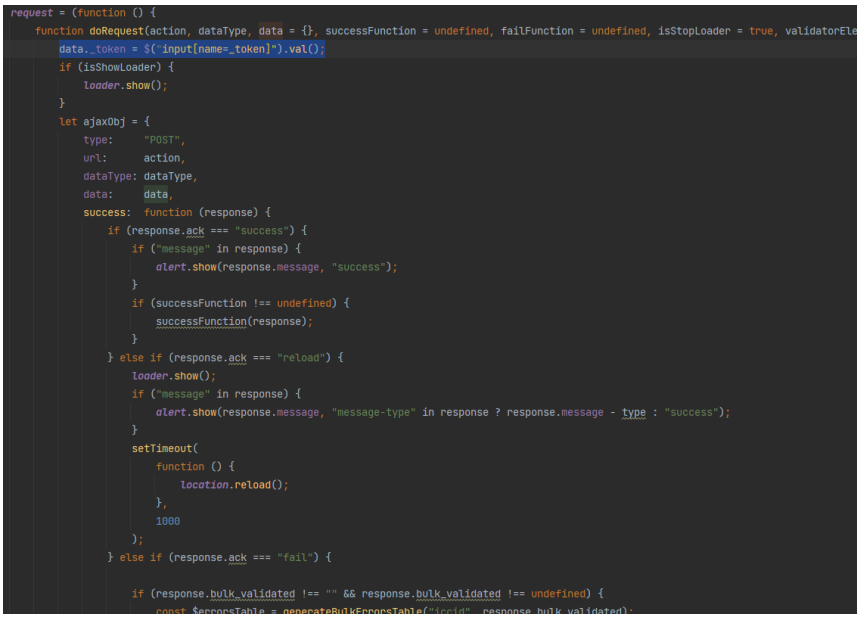
Єдиним недоліком такого методу, є метод атаки за допомогою технології фреймінгу (iFrame). Коли на підробленому доменному імені системи, створюється невелика плівка, котра відкриває у собі справжній сайт.

```php
/**
 * Handle an incoming request.
 * @param Request $request
 * @param Closure $next
 * @return mixed
 */
public function handle(Request $request, Closure $next)
{
    $response = $next($request);
    if ($response instanceof BinaryFileResponse || $response instanceof StreamedResponse) {
        return $response;
    }
    $response->header('Strict-Transport-Security', 'max-age=31536000');
    $response->header('X-Frame-Options', 'DENY');
    $response->header('X-Content-Type-Options', 'nosniff');
    $response->header('Referrer-Policy', 'same-origin');
    $response->header('Cache-Control', 'no-cache, no-store, must-revalidate');
    $response->header('Permissions-Policy', 'microphone=()');
    header_remove( name: "X-Powered-By");
    header_remove( name: "Server");
    $response->header('X-Powered-By', 'Agenses');
    return $response;
}
```

Рисунок 3.7 – Переписаний клас хедерів у побудові сайту для блокування шкідливого ПЗ

Користувач може не звернути увагу на підроблене ім’я домену, бо воно буде досить схожим на справжнє та виконуватиме дії над даними комплексу системи. Тому, перед реалізацією системи Action Token ми заблокуємо можливість відкривати нашу платформу через сторонні сервіси.

Будемо вважати, що проблему подолано, повернімось до нашого хешу. Оскільки, ми використовуємо методи Ajax, додамо обов’язкове поле, що буде використовуватись у системі. Для цього на стороні фронту створимо змінну, яка буде динамічно зберігати токен. А на етапі запиту ми включатимо значення, що зберігається на фронті на даний момент.



```
request = (function () {
    function doRequest(action, dataType, data = {}, successFunction = undefined, failFunction = undefined, isStopLoader = true, validatorEle
        data._token = $("input[name=_token]").val();
        if (isShowLoader) {
            loader.show();
        }
        let ajaxObj = {
            type:     "POST",
            url:      action,
            dataType: dataType,
            data:     data,
            success:  function (response) {
                if (response.ack === "success") {
                    if ("message" in response) {
                        alert.show(response.message, "success");
                    }
                    if (successFunction !== undefined) {
                        successFunction(response);
                    }
                } else if (response.ack === "reload") {
                    loader.show();
                    if ("message" in response) {
                        alert.show(response.message, "message-type" in response ? response.message - type : "success");
                    }
                    setTimeout(
                        function () {
                            location.reload();
                        },
                        1000
                    );
                } else if (response.ack === "fail") {

                    if (response.bulk_validated !== "" && response.bulk_validated !== undefined) {
                        const $errorsTable = generateBulkErrorsTable("#grid", response.bulk_validated);
```

Рисунок 3.8 – Додавання токену до Ajax метода звернень

3.6 Висновки про створене програмне забезпечення

Ця система є складною системою обробки інформації, яка взаємодіє з локальною базою даних. Передача різних типів інформації про обробку - це масив даних, що містить зашифровані токени користувача, який захищає від підробки, крадіжки особистих даних, небезпечних дій, спрямованих на пряме пошкодження, і внесення помилок у систему. У системі реалізовано наступні механізми:

- Повні облікові записи користувачів і автентифікація
- Хешування на льоту та створення унікальних токенів користувачів у середовищі
- Розподіл інформації про рівень доступу та шифрування конфіденційних даних
- Децентралізація доступу з локальна мережа через ARI
- Запити до програмного продукту за допомогою токенів користувача Відстеження користувачів

- Архітектура, побудована для обробки додатків, дозволяє отримати доступ до окремих розділів сторінки під час взаємодії користувача
- Використовує методи Ajax для скритних механізмів обробки даних та запитів непомітних для користувача. Він також обробляє запити «під капотом», тому жодна третя сторона не може уявити, що робить система.
- Створено метод перевірки за допомогою перевірки користувача та сповіщень платформи Telegram.
- Створено двоетапний метод аутентифікації за допомогою QR-коду
- 2 етапний файервол на усіх етапах зв’язку між WEB - server та основним сервером
- Використання сторонніх утиліть від Amazon та Телеграм
- Зручний користувацький інтерфейс
- Методологія запиту через Action Token

Система створена з гнучкими в оптимізації методами, чистим зрозумілим кодом, правильною структурою запитів, захищеним середовищем обробок інформації, сторонніми сервісами для підтримки функціоналу та рекомендаціями з безпеки для користувачів.

Якщо розбирати кожен компонент окремо, як складову комплексної системи захисту інформації, були пропрацьовані такі моменти:

Оцінка ризиків щоб виявити потенційні вразливості та загрози інформаційній безпеці. Це передбачає оцінку ймовірності та впливу різних ризиків, що допоможе визначити пріоритетність заходів безпеки.

Розробка політики інформаційної безпеки, яка окреслює підхід організації до захисту даних. Ці політики повинні охоплювати такі сфери, як класифікація даних, контроль доступу, шифрування, реагування на інциденти та обов'язки працівників.

Надійні засоби контролю доступу, щоб забезпечити доступ до конфіденційної інформації лише уповноваженим особам. Це включає автентифікацію користувачів, доступ на основі ролей та принцип найменших

привілеїв. У системі даний функціонал випливає з методів Auth, Permissions, Account Type.

Шифрування для захисту даних як під час передачі, так і в стані спокою. Це додає додатковий рівень безпеки, ускладнюючи доступ неавторизованих осіб до конфіденційної інформації або її розшифрування.

Брандмауери та системи виявлення/запобігання вторгнень для моніторингу та контролю мережевого трафіку. Ці інструменти допомагають виявляти підозрілі дії та реагувати на них, запобігаючи несанкціонованому доступу та потенційним кібератакам.

Тренінги з підвищення обізнаності про безпеку працівникам про найкращі практики інформаційної безпеки та потенційні ризики. Навчальні програми повинні охоплювати такі теми, як обізнаність про фішинг, безпечні паролі та важливість повідомлення про інциденти, пов'язані з безпекою.

Надійні заходи безпеки кінцевих точок для захисту окремих пристроїв (комп'ютерів, смартфонів тощо). Це може включати антивірусне програмне забезпечення, рішення для виявлення та реагування на інциденти на кінцевих точках (EDR) та регулярне виправлення програмного забезпечення.

План реагування на інциденти щоб забезпечити швидке та ефективне реагування на інциденти безпеки. Цей план повинен містити процедури виявлення, звітування та усунення порушень безпеки.

Надійну система резервного копіювання та відновлення даних, щоб запобігти втраті даних у разі інциденту безпеки або збою системи. API запити реалізують окремий функціонал допуску та управління системою. У випадку втрати свого паролю, користувачі маючи унікальний ключ (який не надається просто так) здатні виходити зі свого акаунту, блокувати його, змінювати паролі.

Заходи фізичної безпеки для захисту інформаційних активів, що зберігаються в приміщенні. Це може включати контроль доступу до центрів обробки даних, систем спостереження та захищених сховищ. Даний етап був проведений емуляцією роботи реальної комплексної системи. У випадку

серверних налаштувань, були введені правила доступу, груп та дозволів. Усі підключення проводяться з використанням RSA ключів.

Дотримання нормативних вимог щодо захисту даних та конфіденційності. Це включає розуміння та дотримання таких законів, як Загальний регламент про захист даних (GDPR) або інших галузевих вимог.

Дотримання правил та відповідних норм щодо захисту даних і конфіденційності. Це включає розуміння та дотримання таких законів, як Загальний регламент про захист даних (GDPR) або інших галузевих вимог. Безпека постачальників і третіх сторін:Оцінюйте та керуйте практиками безпеки сторонніх постачальників і партнерів. Переконайтеся, що вони відповідають стандартам безпеки вашої організації, щоб запобігти вразливостям через зовнішні з'єднання.

Постійне вдосконалення системи інформаційної безпеки, щоб адаптуватися до нових загроз і технологій. Проводьте періодичні оцінки, вивчайте інциденти, пов'язані з безпекою, і відповідно коригуйте заходи безпеки.

Створення комплексної системи інформаційної безпеки - це безперервний процес, який вимагає поєднання технологій, політики, освіти та пильного моніторингу. Регулярне оновлення та адаптація до нових загроз є важливими для підтримання ефективності заходів безпеки з плином часу.

4 КОМПЛЕКСНА СИСТЕМА ЗАХИСТУ ІНФОРМАЦІЇ ТА ЇЇ ТЕСТУВАННЯ

4.1 Опис основного функціоналу системи

У даному комплексі впроваджена велика кількість захисного функціоналу, впроваджена система ролей, організована псевдодокументація та політика використання. Створені рекомендації для користувачів системи задля захисту їх конфеденційної інформації.

Будь-який користувач з підключенням до інтернету буде здатним отримати інформацію з найнижчим рівнем критичності, подібно до стрічки новин. Сама ж структура виглядає, як платформа для блогерів, письменників чи читачів. Вони матимуть змогу публікувати свої нотатки, але виключно від імені акаунту власника (фірми) з якою вони співпрацюють. Якщо звичайний користувач хоче зареєструватись, він отримає доступ до даних, виключно, основного акаунту системи.

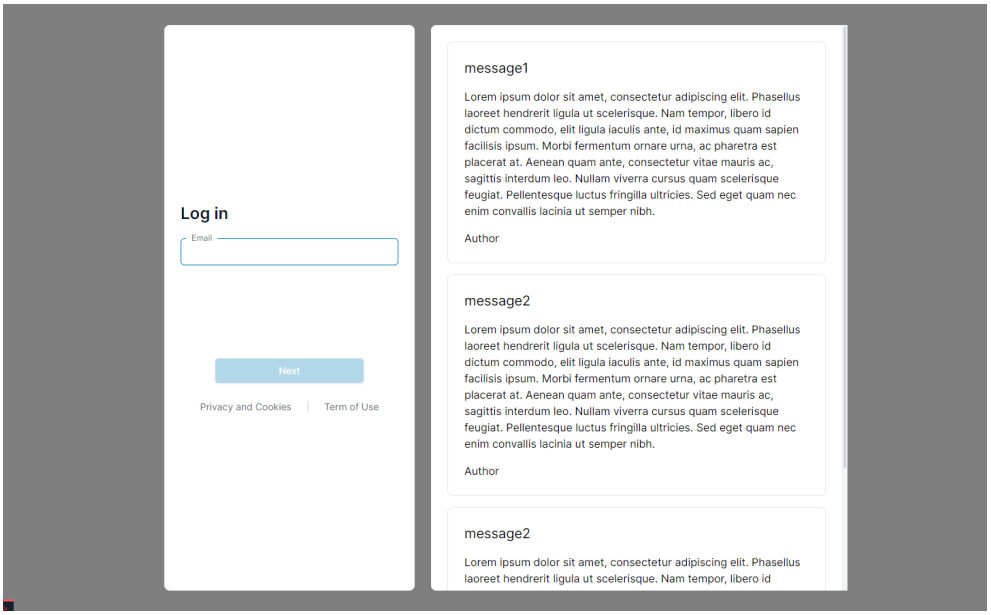


Рисунок 4.1 – Головна сторінка

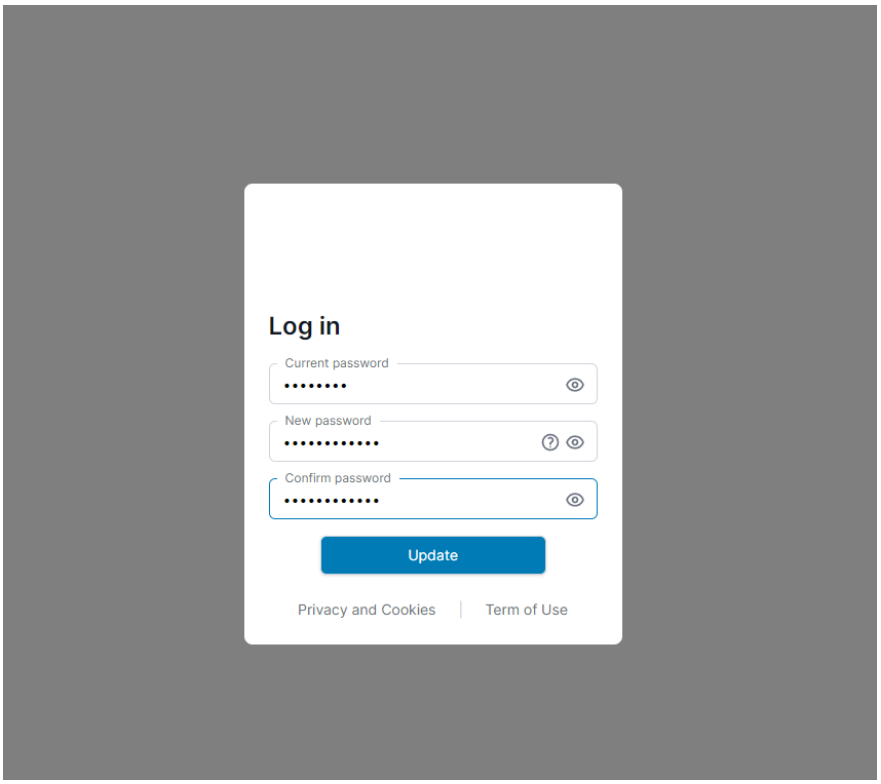
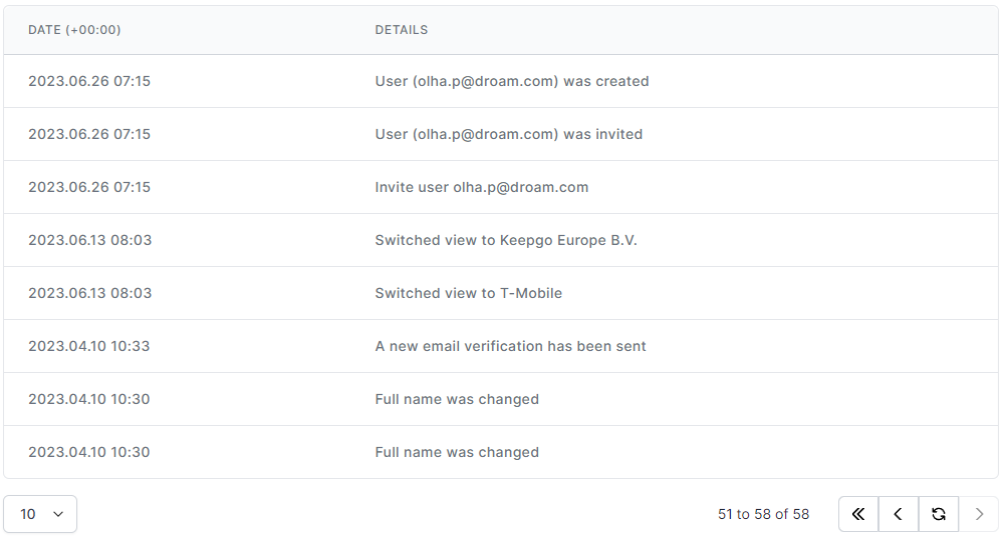


Рисунок 4.2 – Демонстрація функціоналу клієнтів

У кожного клієнта будуть косметичні та функціональні налаштування, пов’язані з його акаунтом, від зміни паролю, аватарів, імені і іншого, до можливості змін налаштувань акаунту та сповіщень.



DATE (+00:00)	DETAILS
2023.06.26 07:15	User (olha.p@droam.com) was created
2023.06.26 07:15	User (olha.p@droam.com) was invited
2023.06.26 07:15	Invite user olha.p@droam.com
2023.06.13 08:03	Switched view to Keepgo Europe B.V.
2023.06.13 08:03	Switched view to T-Mobile
2023.04.10 10:33	A new email verification has been sent
2023.04.10 10:30	Full name was changed
2023.04.10 10:30	Full name was changed

Рисунок 4.3 – Історія логування з системи користувача

Усі дії користувачів створюють запис у таблицях Action та Security Log. Виключні користувачі будуть здатні переглядати логи кожного, у своєму акаунті. Немає виключень, не буде можливості видалити логування через інтерфейс чи API.

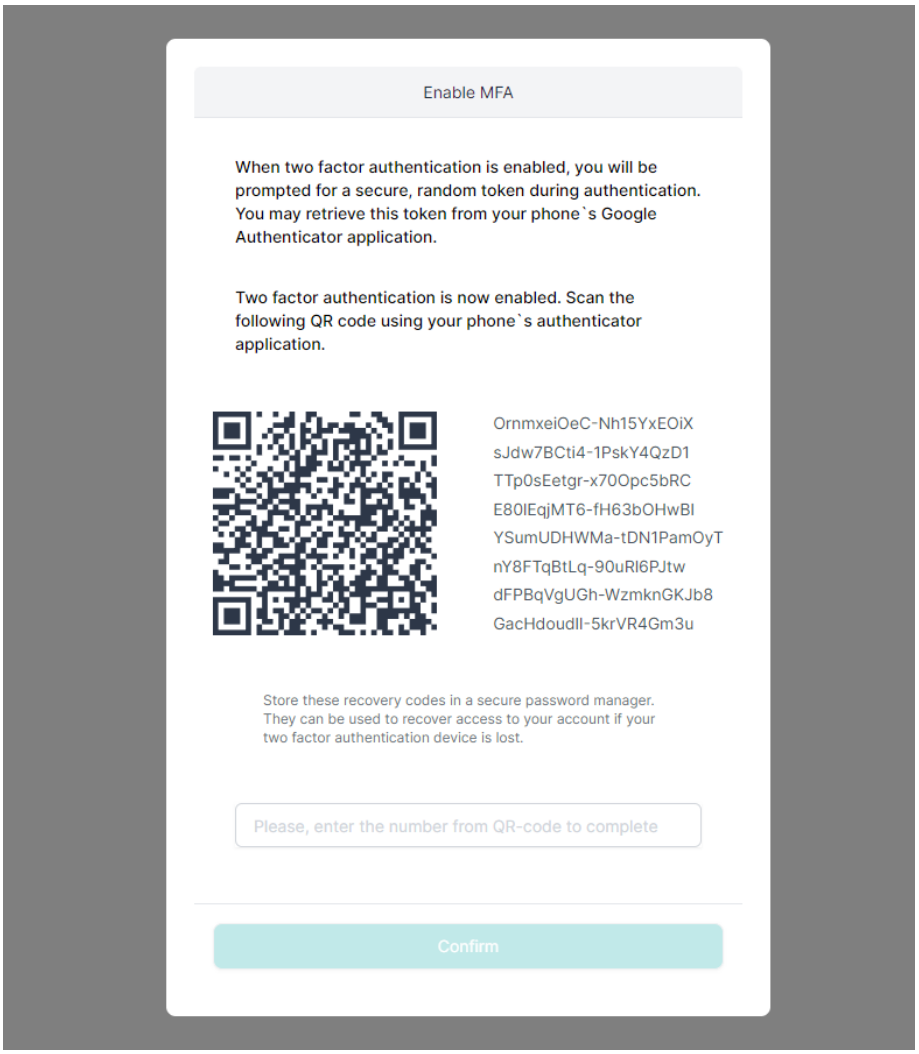


Рисунок 4.4 – вікно генерації 2FA коду та кодів відновлення

За етапами можна розподілити цю функцію таким чином:

Користувач входить в систему, ввівши своє ім'я користувача та пароль. Система надає другу форму автентифікації для користувача. Користувач надає другий елемент. Це може бути код, надісланий на ваш мобільний пристрій, згенерований програмою автентифікації або отриманий з апаратного маркера.

Якщо обидва елементи правильні, доступ дозволено. Поширені способи впровадження 2FA: Текстові повідомлення (SMS): одноразовий код надсилається на мобільний телефон користувача через SMS. Програма автентифікатора: Користувач встановлює програму автентифікації (наприклад, Google Authenticator, Authy), яка генерує код на основі часу.

Перевірка електронної пошти: Код або посилання надсилається на електронну адресу користувача. Hardware Token: фізичний пристрій , який генерує або відображає код. Біометрія: розпізнавання відбитків пальців або обличчя на одному пристрої.

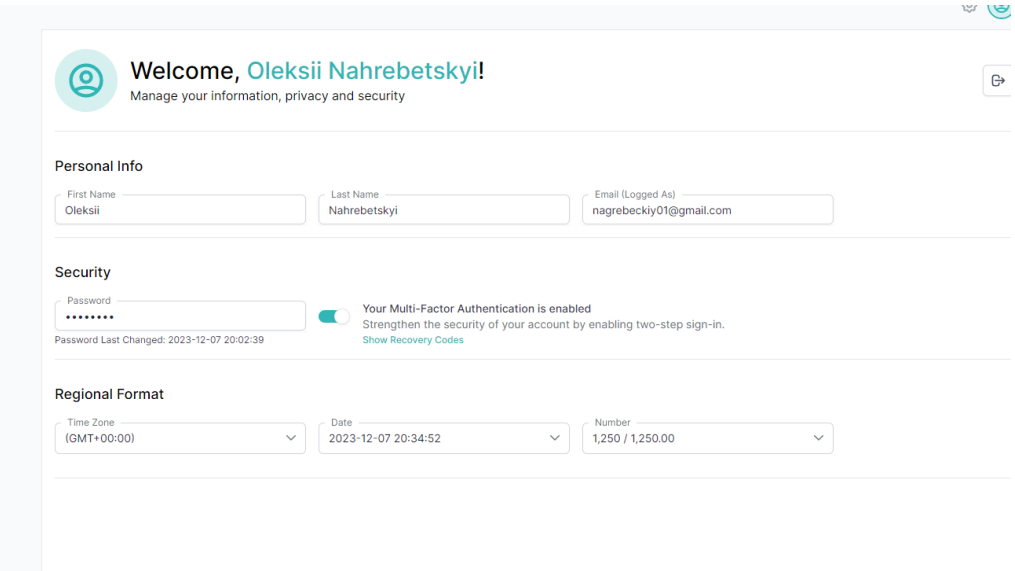


Рисунок 4.5 – Вікно профілю користувача

У даному вікні реалізовано функціонал для змін конфіденційної інформації користувача, окрім цього є ще налаштування для акаунту. Виключно, у власників.

4.2 Економічна ефективність системи

Економічну ефективність інтегрованої системи захисту інформації (ISIS) можна оцінити на основі різноманітних факторів, які враховують як витрати, так і вигоди, пов’язані з її впровадженням.

Важливими аспектами, які слід враховувати при оцінці економічної ефективності CPS, є:

Вартість впровадження: Початкова вартість: Придбання та впровадження інтегрованої системи безпеки, включаючи апаратне забезпечення, програмне забезпечення тощо. Оцініть початкові витрати. Консалтингові послуги. Розглянемо витрати, пов’язані з навчанням працівників роботі з новою системою. Технічне обслуговування та оновлення: варто перевірити поточні витрати на обслуговування системи, оновлення та оновлення. Моніторинг і управління: це витрати, пов'язані з постійним моніторингом, реагуванням на інциденти та управлінням системою.

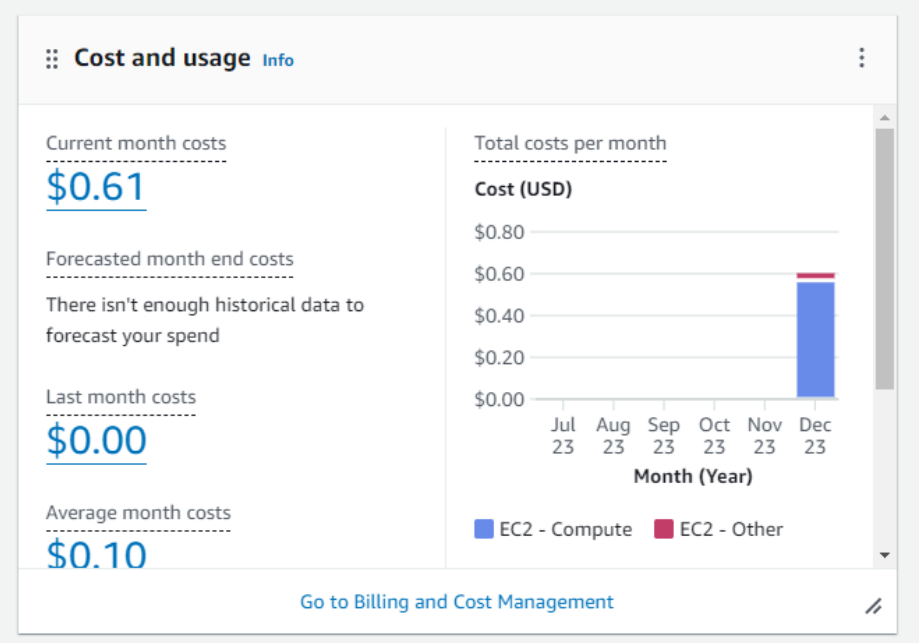


Рисунок 4.6 – статистика економічних затрат від Amazon, дане вікно було зроблено на етапі запуску системи, кошти на підтримку сервера на кінець запуску були 4 долари в місяць

Зменшення кількості інцидентів безпеки: це моніторинг впливу на частоту та серйозність інцидентів безпеки. При правильному застосуванні CSIR зменшує ймовірність і вплив порушень безпеки. Оцінка необхідна і в потенційних

фінансових втратах, яких можна уникнути, запобігши інциденту безпеки. Але у даному проекті зробити подібні заключення досить важко

Продуктивність і операційні переваги:

Економія часу: скільки часу можуть заощадити співробітники, покращивши процеси безпеки та впорядкувавши робочі процеси.

Операційна ефективність та її можливість підвищення загальної ефективності роботи в результаті більш безпечних і надійних інформаційних систем.

Дотримання нормативних вимог, уникнення штрафних санкцій. Економічні вигоди від уникнення штрафних санкцій і правових наслідків, пов’язаних із порушеннями.

Репутація та довіра клієнтів: цінність захисту бренду та репутації вашої організації на ринку. Довіра клієнтів оцінка впливу посилених заходів безпеки на довіру та лояльність клієнтів.

Масштабованість і гнучкість:: Оцініть здатність системи розширюватися та адаптуватися в міру зростання організації без непропорційних витрат. Уникнення майбутніх витрат: варто перевірити потенційну економію коштів, пов’язану з уникненням майбутнього капітального ремонту системи.

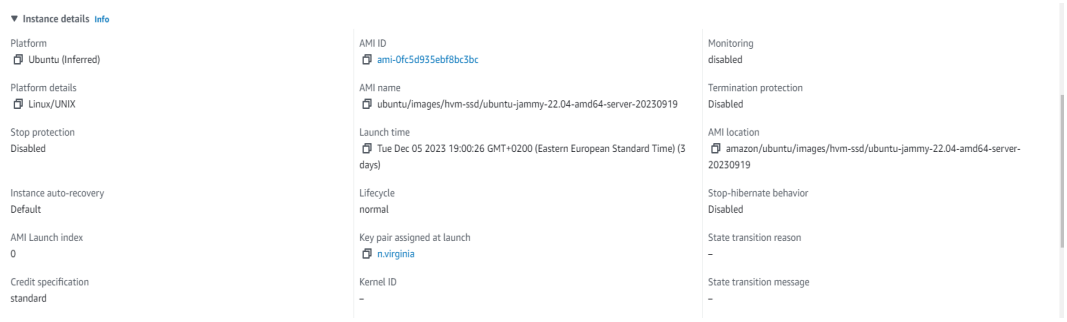


Рисунок 4.7 – статистика та налаштування фізичних можливостей серверів, їх можна масштабувати при необхідності

Загальна вартість володіння (TCO):

Комплексна оцінка: Враховує загальну вартість володіння протягом життєвого циклу системи, включаючи прямі та непрямі витрати.

Рентабельність інвестицій (ROI) від покращень безпеки. Збільшення продажів, залучення клієнтів або можливості для бізнесу. Кількісно визначені переваги пов’язані з впровадженням IISS.

Аналіз витрат і вигод: порівняємо витрати і вигоди: Виконайте аналіз витрат і вигод, щоб зважити витрати і вигоди за певний період часу.

Постійне вдосконалення: Адаптація та вдосконалення: Розгляньте економічну цінність здатності IISS адаптуватися до нових загроз і технологій, уникаючи таким чином частої зміни кадрів. Постійне вдосконалення: Адаптація та вдосконалення: Розглянемо економічну цінність здатності SIMIS адаптуватися до нових загроз і технологій, таким чином уникаючи необхідності частої заміни.

Страхові премії: Вплив на витрати на страхування: Оцініть, чи покращило впровадження IISS управління ризиками та зменшило страхові премії.

Еталонний показник: Еталонний показник промисловості: Порівняйте економічну ефективність IISS із галузевими стандартами та найкращими практиками.

Таким чином, оцінка економічного впливу інтегрованих систем інформаційної безпеки вимагає цілісного підходу, який враховує як понесені витрати, так і отримані вигоди.

Щоб гарантувати, що CSIB продовжуватиме забезпечувати цінність перед обличчям нових проблем кібербезпеки та організаційних змін, необхідно проводити регулярну оцінку та коригування. [36]

Не забудемо, що при оцінці важко визначити мій особистий вклад, на створена самої дипломної роботи та проекту пройшло більше 3 місяців. Не можна зробити еквівалент у годинах часу, для цієї роботи, бо не існує на ринку пропозицій подібного широко спектру профейсійних навичок, що були продемонстровані у роботі.

Якщо відійти від абстракції, для реалізації подібного проекту необхідно мінімум з десяток спеціалістів різних сфер, з досвідом та чітко визначеними задачами розробки. Додати до цього усього щомісячні виплати сервісам послуг, операторів, тощо.

І варто зазначити, що не було реальних практичних кейсів від співробітників, що варто було б оптимізувати, покращити, змінити… А ці пунтки виникають виключно, під час роботи співробітників, які теж хочуть отримати гроші, за свою працю..

У даному сегменті не було оцінено комерційної вигоди проекту, оскільки немає реальної цілі у вигоді. З теоретичної точки зору, дохід міг би припадати на запити рекламодавців, кастомізації функціоналу під конкретні прохання від клієнтів, створення системи “найпопулярніші статі”, пошуку нових ринків збуту подібного.

Оглянувши усі можливі надходження та витрати, даний проект при гідній реалізації функціоналу та капіталовкладень, міг би приносити 140% дохід (40% чистого доходу) щомісяця, від вкладених затрат, без обрахування суми початкового капіталу. Дана оцінка є суб’єктивною та відповідає аналізу ринку за 2023 рік у промислі новинного, текстоописного характеру.

Приблизні перші значні доходи, будуть через 8 місяців, з дня запуску системи та адекватної кількості зацікавлених користувачів у даній системі, що ще додає 2-3 місяці до цього часу. Округлюємо до одного року, з моменту, коли проект приноситиме реальні кошти.

4.3 Перевірка системи. Етичний хакінг

Етичне хакерство (також відоме як етичне хакерство, тестування на проникнення або пентестування) — це метод перевірки рівня захисту від кібератак на комп’ютерну систему, який виконується для оцінки безпеки системи, створений за допомогою схваленого моделювання.

Як воно працює?. Виконується тести для виявлення вразливостей (також відомих як уразливості), які передбачають можливість неавторизованих осіб отримати доступ до функціональних можливостей системи або даних, і міцності, яка дозволяє повну активацію.

Подібні прийоми не нові, є навіть спеціалізовані організації, які займаються цим питанням. Їх перевага в тому, що створені сервіси можуть забезпечити безпеку користувачів.

Однак його корисність досить відносна, оскільки витрачати непотрібні гроші на захист нікому не потрібної інформації – безглуздо.

Нещодавно постачальники програмного забезпечення також заснували власні дослідницькі інститути з цієї теми.

Те ж саме відбувається з військовими компаніями, які виробляють обладнання та техніку. Питання, які виникають під час тестування на проникнення, можуть бути критичними та вимагати негайного вирішення.

Всесвітній економічний форум назвав кібератаки п'ятим за величиною ризиком 2020 року. Оскільки 77% лідерів безпеки передбачають порушення критичної інфраструктури, великим і малим компаніям важливо підготуватися до таких подій. Крім того, очікується, що до кінця 2023 року вартість IT-індустрії досягне 170,4 мільярда доларів, що вказує на швидке зростання фінансових кіберризиків.

Кібератаки можуть статися будь-коли, тому потрібно бути готовим і приймати правильні рішення. Атаки можуть бути як активними, так і пасивними, зсередини або ззовні організації.

Раннє виявлення атак може заощадити гроші компаній і запобігти подальшому доступу до конфіденційної інформації шляхом вимкнення систем і сповіщення відповідних сторін.

Користувач намагаючись знайти корисну для себе інформацію у мережі інтернет, надає доступ зловмиснику до власного ПК, натиснувши на невідоме посилання. Це звичайна ситуація в наш час, а особливо для людей літнього віку, котрі не дуже усвідомлені в планах кібергігієни... У подібних випадках, зазвичай,

спрацьовує антивірус, при переході з гіперпосилання, або завантаженні шкідливого ПО. Подібний захист оминути просто, чи повідомивши на сайті неначе «Ваш антивірус може блокувати завантаження, або уповільнити його, вимкнути антивірус та продовжуйте», створювати – дублікати (невидимки), досить свіжі вірусні продукти, що є дочірніми до відомих програм, проте, ще не занесені до словників антивірусних програм, або ж напівшкідники – трояни, черви і т. д., програми, що є відповідними та мають невеликі зміни на користь хакера. Або ж заархівовані файли, оскільки антивірус перевіряє не лише назву файлу, а й його зміст, стиснені або зашифровані файли – неможливо перевірити на всі 100%, чим і користуються злочинці.

В даному випадку для створеної програми головною ціллю буде захистити конфіденційні дані у БД. Беручи до уваги, що безпосередні гарантії захисту надає сам продукт MySQL, навіть, у випадку його нестачі – зловмисники не можуть використати дані з максимальною ефективністю. В їх розпорядженні буде конфіденційна інформація, проте, вони не зможуть нею скористатися через те, що вони не зберігаються у звичному вигляді.

1) Хакер отримав доступ до ПК, що підключений до мережі, що далі?

Наступним, йому необхідно оминути захист системи самої бази даних і він матиме 2 шляхи. Перший – намагатись відгадати логін та пароль користувача, щоб попередити подібні вчинки усьому персоналу були повідомлені вказівки, щодо їх паролів та логінів, збережено двофакторну аутентифікацію при підозрілих вчинках над БД, використовуються складні паролі довжиною 8 символів з високим символьним діапазоном та лімітом на кількість введень паролів.

Другий спосіб – це приєднатись напряму до локального сервера та атакувати БД напряму. Тут існує декілька можливих атак. Першою є маскування під адміністратора, друге – атака самого сервісу MySQL Server. Проте, як показує практика для організації подібної атаки на сервіс, необхідно витратити значно більше коштів та часу, ніж цінність внутрішньої інформації... Отримання доступу до журналу транзакцій не зможе призвести до достатньої загрози,

оскільки створені токени користувачів прив’язані строго до часу та їх особистий даних. Так, у системі залишено відкритим журнал транзакцій користувачів, проте, він зашифрований ключем RSA – 128, це створено з метою «відбою». Якщо ж сервіс вдалося зламати та йде атака на БД з використанням ПК користувача, адміністратор зможе одразу ж від’єднати ПКП від системи відкривши файл транзакцій та переглянути його на активність. Маскування під адміністраторів можливе лише з унікальними ідентифікатора адміна БД – ключа підключення до бази даних (256 – бітний згенерований ключ) чи використанням журналу автоматичного входу (перевіряє унікальний номер процесора). Нажаль, сервіс MySQL Server, справді має деякі недоліки у плані захисту інформації, проте, обраховуючи цінність інформації, що знаходиться на сервері, використання ще більш сильного захисту є недоцільним. [37].

Існує ще одна послуга аутсорсингу невеликої компанії, яка виконує етичне хакерство для виявлення прогалин у захисті цієї версії програмного забезпечення.

4.4 Проведення пробних атак на створений сервіс за допомогою сторонніх програм

NetLimiter – додаткове програмне забезпечення для моніторингу та передачі трафіку в підключених мережах.

Це шкідливо, оскільки як тільки злочинці отримають доступ до вашого Wi-Fi, вони можуть перехопити та отримати дані, передані на хост вашого провайдера.

Як ми бачимо, справдні є трафік на локальній підмережі, проте, беручи до уваги, що дана система є локальною, щоб виконати ці дії необхідно мати до неї доступ, або ж доступ до ПК у цій сітці, що дає більше проблем для взлому.

NetLimiter часто використовується приватними особами та організаціями для оптимізації продуктивності мережі, ефективного розподілу смуги

пропускання і запобігання перевантаженню мережі певними програмами. Він може бути корисним у різних сценаріях, таких як управління пропускною здатністю в середовищах спільного використання, забезпечення якості обслуговування (QoS) або просто моніторинг і контроль використання Інтернету на персональному комп'ютері.

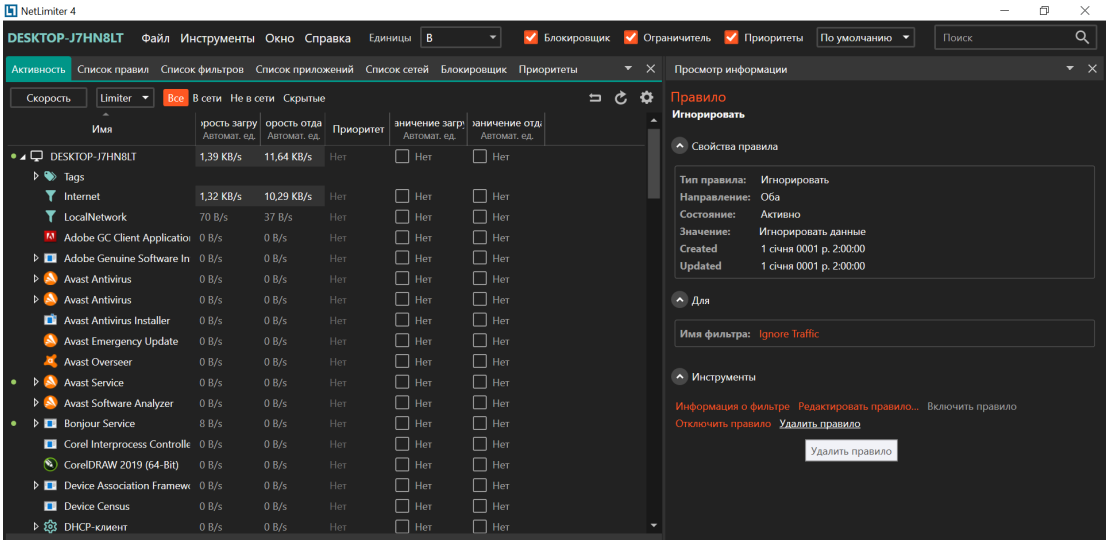


Рисунок 4.8 – вікно програми NetLimiter та дані трафіку

Проведемо ще одну атаку зі словником, намагаючись оминути блокування кількості введень:

Рисунок 4.9 – Спроба атаки словником на сервіс за допомогою програми HashCat

Нам не вдалося, буквально, одразу. Вже після повторного запиту ми отримали помилку 429, відмову в обслуговуванні. Сама система відслідковує подібне та не блокуватиме користувача одразу, а через декілька спроб. Щоб повністю повторити повний цикл, для автоматичного блокування, необхідно набирати повний ліміт запитів протягом 2 годин, що неможливо для справжнього користувача.

У даному розділі було протестовано стабільність системи при атаках пов’язаних на стійкість сервера та програмного забезпечення. Платформа витримала псевдоатаки успішно.

ВИСНОВКИ

Повсякчасно системи обміну інформації перетворились у мастодонтів організаційного процесу. Продовжуються вводитись нові методи організацій, розробки, з’являються нові механізми ООП. Будь-яка система включає у собі спеціалістів з різних сфер, від менеджменту, таргетингу та служби підтримки, до юристів, девелоперів, мережників, тощо. Кількість людей - необмежена, тому за повноцінний комплекс не може братися одна людина, не є проблемою низька спеціальноорієнтованість, головною проблемою є фізичні затрати часу та обмеженість бюджетів.

Це програмне забезпечення є результатом знань, отриманих під час розвитку цієї галузі. Через недостатню кількість необхідного часу та складність реалізації окремих систем тут включені не всі набуті знання.

На основі аналізу та системного підходу розроблено та створено середовище, яке забезпечує належний рівень захисту користувачів.

З'ясувати характеристики ринкової пропозиції, розуміння потреб споживачів, рішення та альтернативи, результати випробувань і технічних засобів. Ця робота стала втіленням копіткої праці багатьох фахівців із різних галузей. У результаті ми маємо продукти, які служать послугою захисту наших громадян.

Після аналізу та інтеграції складних систем захисту інформації я покращив не лише свої навички програмування та кіберзахисту, а й мої навички програмного забезпечення, які є досвідом інтеграції проектів у поле реальних робочих процесів.

Для впровадження повноцінного продукту необхідно було поспілкуватися зі співробітниками та керівництвом компанії, реалізувати їхні побажання, провести тестову інтеграцію системи та скоординувати роботу.

Однак неможливо запобігти всім вторгненням, ненавмисним помилкам персоналу, технічним чи інформаційним «розмиванням» і всім злочинним

діям, які можна спробувати вжити заради власної справи, тому ідеальний всеохоплюючий Слід зазначити, що немає повна система захисту.

Кібербезпека – це своєрідний ультиматум проти незаконної діяльності.

Тому що наше життя пронизане технологічними засобами, які не тільки мають високу вартість, але й використовуються для передачі особистої та конфіденційної інформації, банківських послуг і комунікацій.

У даній роботі не реалізовано усіх можливих потужностей комплексної системи захисту, оскільки подібних “країв” не видно і не існує. Система максимально оптимізована для покращень та втілень ідей, можливостей переробок та редизайну, тощо. Усі механізми використовують стандарти ООП та програмних утиліт.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. United States Federal Energy Regulatory Commission. Federal Energy Regulatory Commission Assessment of Demand Response & Advanced Metering. URL: https://www.ferc.gov/legal/staff-reports/12-08-demand-response.pdf (дата звернення: 05.10.2023).

2. Zhao, Jinquan; Huang, Wenying; Fang, Zhaoxiong; Chen, Feng; Li, Kewen; Deng, Yong (2007-06-24). On-Line Voltage Stability Monitoring and Control (VSMC) System in Fujian power grid. 2007 IEEE Power Engineering Society General Meeting. Proceedings, Power Engineering Society General Meeting, 2007. (PDF) (Tampa, FL, USA: IEEE): 1. ISBN 14244-1296-X. doi:10.1109/PES.2007.385975. Загальний огляд.

3. Чинчик Д., Коробейнікова Т., Захарченко С. Методи та засоби комплексного захисту корпоративної мережі. *InterConf*. 2021. №84. С.433-450.

4. Корпань Я.В. Класифікація загроз інформаційній безпеці в комп’ютерних системах при віддаленій обробці даних / Я.В. Корпань // Світ науки і інновацій. – Науковий світ, 2015. – Т. 17, № 2. – С. 39–46.

5. Корпань Я.В. Комплекс методів і засобів захисту інформації у комп’ютерних системах / Я.В. Корпань // Реєстрація, зберігання і обробка даних. – 2015. – вип. 1. – Т. 3 2. – С. 31–35.

6. Локазюк В. М. Засади систем підтримки прийняття рішень на основі комп'ютерних систем та їх компонентів : навч. посіб. / В. М. Локазюк, О. В. Іванов, В. Ю. Тітова ; Хмельниц. нац. ун-т. – Хмельницький : Гонта А.С., 2010. – 338 с.

7. Yih-Fang Huang; Werner, S.; Jing Huang; Kashyap, N.; Gupta, V., "State Estimation in Electric Power Grids: Meeting New Challenges Presented by the Requirements of the Future Grid," Signal Processing Magazine, IEEE , vol.29, no.5, pp.33,43, Sept. 2012 257

8. Тітова В.Ю. Класифікація моделей загроз в комп’ютерних системах/ В.Ю. Тітова, Ю.П. Кльоц, С.О. Савчук. – Вісник Хмельницького національного університету. – №2, 2020 (283). – С. 201-204.

9. *С.Д. Кузнєцов.* Об’ектно-орієнтовані бази даних – основні концепції, організації та їх управління. CIT Forum, 2014 – 551 с.

10. *Стівен Хольцнер.* Ajax Біблія програміста = Ajax Bible. — М.: Діалектика, 2009. — С. 553. — ISBN 978-5-8459-1502-3.

11. Реалізація процесного підходу до керування ризиками інформаційної безпеки в документах NIST [Електронний ресурс] – Режим доступу: https://ela.kpi.ua/bitstream/123456789/23949/1/ITS2017.5.2(9)_09.pdf

12. Методичне забезпечення оцінки ризиків підприємства [Електронний ресурс] – Режим доступу: https://periodicals.karazin.ua/soceconom/article/download/4813/4366/#:~:text

13. Петрова В.Ф. “Методичні основи оцінки ризиків підприємницької діяльності”, Харківський національний університет міського господарства імені О.М. Бекетова [Електронний ресурс] – Режим доступу: http://www.vestnikdnu.com.ua/archive/201154/171-176.pdf

14. Література Методи якісного аналізу підприємницьких ризиків [Електронний ресурс] – Режим доступу: http://www.dut.edu.ua/uploads/l_50_49235071.pdf

15. Risk Management Guide for Information Technology Systems [Електронний ресурс] – Режим доступу: https://csrc.nist.gov/publications/detail/sp/800- 30/archive/2002-07-01

16. Розробка у Laravel для пошуку даних - Wayback Machine.[ЕЛЕКТРОННИЙ РЕСУРС] http://creative-punch.net/articles/php-articles/laravel-tutorials/

17. Прикладна криптологія : системи шифрування : підручник / О. Г. Корченко, В. П. Сіденко, Ю. О. Дрейс. – К. : ДУТ, 2014. – 448 с.

18. Шинкарук О.М. Основи функціонування багатоканальних систем передачі інформації: навч. посіб./ О.М. Шинкарук, Ю.М. Бойко, І.І. Чесановський. – Хмельницький : ХНУ, 2011. – 245с.

19. Кодування джерел інформації та каналів зв’язку: навчальний посібник / [Беркман Л.Н., Бондарчук А.П., Гайдур Г.І., Чумак Н.С.]. – Київ: НННТІ ДУТ, 2018. – 91 с.

20. The Hacker Playbook 3: Practical Guide To Penetration Testing - Peter Kim, США 2018, 2018 ISBN-13: 978-1980901754

21. Laravel Sanctum [Електронний ресурс]: Laravel Sanctum – The PHP Framework For Web Artisans – Режим доступу : https://laravel.com/docs/9.x/sanctum. Назва з екрана.(електронне джерело)

22. PHP Artisam [Електронний ресурс]: - відкрите інтернет джерело для програмування – Режим доступу : https://laravel.com/docs/9.x/artisan

23. Інформаційна надмінність [Електронний ресурс] – Режим доступу - https://jak.koshachek.com/articles/informacijna-nadmirnist.html

24. Efficient MySQL Performance: Best Practices and Techniques - Daniel Nichter – 2021 p. 94-147 c.

25. High Performance MySQL: Proven Strategies for Operating at Scale - Silvia Botros, Jeremy Tinley – 2021 p. – 64 – 88 c.

26. PHP and MySQL for Dynamic Web Sites - Larry Ullman – 2017 p. – 140-148 c.

27. HASH 256 [Електронний ресурс]: Що таке хеші MD5, SHA-1 і SHA-256 і як їх перевірити? Автор: Jorge Stolfi / Wikimedia [Електронний ресурс]: - https://ua.phhsnews.com/articles/howto/what-are-md5-sha-1-and-sha-256-hashes-and-how-do-i-check-them.html

28. [ЕЛЕКТРОННЕ ДЖЕРЕЛО] Найпопулярніші кібератаки 2020-2021 років – Режим доступу - https://10guards.com/ua/articles/the-most-common-types-of-cyber-attacks-in-2021/.

29. Бойко В.В., Савинков В.М. Проектування баз даних із включенням складних багаторівневих запитів. – М.: Финансы и статистика, 2013. – 351 с.

30. PHP, MySQL, & JavaScript All-in-One For Dummies. - Richard Blum – 2018 – 210 – 234 с.

31. PHP and MySQL Recipes - Frank M. Kromann – 2016 р.- 400 с.

32. Атре Ш. Структурований підхдід до організації баз даних. – М.: Фінанси та статистика, 2003. – 320 с.

33. Molinaro, A. (2020). SQL Cookbook: Query Solutions and Techniques for All SQL Users. In R. de Graaf (Ed.), (ст. 140–322).

34. Джерело програмних ресурсів, форум допомоги програмістам [Електронний ресурс] https://habr.com/post/181772/

35. Локазюк В.М. Контроль і діагностування обчислювальних пристроїв та систем. Навч. посібник для вузів. Хмельницький, ТУП, 1996. – 175 с.

36. Меєр М. Теорія реляційних баз даних. – М.: Світ, 1997. – 608 с.

37. Хаббард Дж. Автоматизоване проектування баз даних. – М.: Мир, 2010. – 294 с.

38. Introducing InnoDB Cluster: Learning the MySQL High Availability Stack - Charles Bell – 2018 р.

39. Закон України "Про інформацію". https://zakon.rada.gov.ua/laws/main/2657-12 Закон України "Про доступ до публічної інформації".

40. Закон України "Про захист інформаційно- телекомунікаційних системах". https://zakon.rada.gov.ua/laws/show/80/94-%D0%B2%D1%80

41. За Закон України "Про основні засади забезпечення кібербезпеки України" https://zakon.rada.gov.ua/laws/main/2163-19

42. Закон України "Про електронні документи та електронний документообіг". https://zakon.rada.gov.ua/laws/show/851-15

43. Постанова Кабінету Міністрів України "Про затвердження Правил забезпечення захисту інформації в інформаційних, телекомунікаційних та

інформаційно-телекомунікаційних системах" від 29.03.2006 № 373. https://zakon.rada.gov.ua/laws/main/373-2006-%D0%BF

44. Постанова Кабінету Міністрів України "Про затвердження Загальних вимог до кіберзахисту об'єктів критичної інфраструктури" від 19.06.2010 № 518.

45. Постанова Кабінету Міністрів України "Про затвердження Типової інструкції про порядок ведення обліку, зберігання, використання і знищення документів та інших матеріальних носіїв інформації, що містять службову інформацію" від 19 жовтня 2016 р. № 736.

46. ДСТУ 33960-96. Захист інформації. Технічний захист інформації. Основні положення. http://www.dsszzi.gov.ua/dsszzi/control/uk/publish/article?art id=38911&cat_id=38836 12. ДСТУ 33961-96 Захист інформації. Технічний захист інформації. Порядок проведення робіт. http://www.dsszzi.gov.ua/dsszzi/control/uk/publish/article?artid=38911&cat_id=3 8836

47. Electric Power Research Institute, IntelliGrid Program Архівовано 18 травень 2007 у Wayback Machine. 33. U.S. Department of Energy, Office of Electric Transmission and Distribution, "Grid 2030" A National Vision for Electricity's Second 100 Years Архівовано 21 липень 2011 у Wayback Machine., July 2003

48. Tomoiagă, B.; Chindriş, M.; Sumper, A.; Sudria-Andreu, A.; Villafafila-Robles, R. Pareto Optimal Reconfiguration of Power Distribution Systems Using a Genetic Algorithm Based on NSGA-II. Energies 2013, 6, 1439-1455.

49. Mohsen Fadaee Nejad, AminMohammad Saberian and Hashim Hizam (June 3, 2013). Application of smart power grid in developing countries. 7th International Power Engineering and Optimization Conference (PEOCO) (IEEE). doi:10.1109/PEOCO.2013.6564586.

50. Gridwise History: How did GridWise start?. Pacific Northwest National Laboratory. 200710-30.

51. Patrick Mazza (2005-04-27). Powering Up the Smart Grid: A Northwest Initiative for Job Creation, Energy Security, and Clean, Affordable Electricity. (doc). Climate Solutions. с. 7. Архів оригіналу за 2008-12-30.

52. E-Commerce News: Deals: Utility Companies Plug In to Google PowerMeter. Ecommercetimes.com. Retrieved on 2011-05-14. that PMUs can revolutionize the way power systems are monitored and controlled."»

53. Smart Wire Grid Distributed Power Flow Control. arpa-e.energy.gov. Процитовано 201407-25.

54. Yilu Liu, Lamine Mili, Jaime De La Ree, Reynaldo Francisco Nuqui, Reynaldo Francisco Nuqui (2001-07-12). State Estimation and Voltage Security Monitoring Using Synchronized Phasor Measurement. Research paper from work sponsored by American Electric Power, ABB Power T&D Company, and Tennessee Valley Authority (PDF) (Virginia Polytechnic Institute and State University). CiteSeerX: 10.1.1.2.7959. ”Simulations and field experiences suggest”

55. Wide Area Protection System for Stability (PDF). Nanjing Nari-Relays Electric Co., Ltd. 2008-04-22. с. 2. Архів оригіналу за 2009-03-18.

56. Energy Future Coalition, "Challenge and Opportunity: Charting a New Energy Future," Appendix A: Working Group Reports, Report of the Smart Grid Working Group.

57. Seeba, M., Mäses, S., Matulevičius, R. (2022). Method for Evaluating Information Security Level in Organisations. In: Guizzardi, R., Ralyté, J., Franch, X. (eds) Research Challenges in Information Science. RCIS 2022. Lecture Notes in Business Information Processing, vol 446. Springer, Cham. https://doi.org/10.1007/978-3-031-05760-1_39

58. Stango, Antonietta & Prasad, Neeli & Kyriazanos, Dimitris. (2009). A Threat Analysis Methodology for Security Evaluation and Enhancement Planning. 262-267. 10.1109/SECURWARE.2009.47.

ДОДАТОК А

Копії наукових публікацій

УДК 004.056.53
DOI:

ТІТОВА ВІРА
Хмельницький національний університет
ORCID ID: 0000-0001-8668-4834
e-mail: titovav@khmnu.edu.ua
КЛЬОЦ ЮРІЙ
Хмельницький національний університет
ORCID ID: 0000-0002-3914-0989
e-mail: klots@khmnu.edu.ua
НАГРЕБЕЦЬКИЙ ОЛЕКСІЙ
Хмельницький національний університет
e-mail: nagrebeckiy01@gmail.com

УПРАВЛІННЯ РИЗИКАМИ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ В РОЗПОДІЛЕНИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ ОБРОБКИ ПЕРСОНАЛЬНИХ ДАНИХ

У даній статті авторами було проаналізовано існуючі на сьогоднішній день підходи до управління ризиками в інформаційних системах. За результатами аналізу було зроблено висновки, що більшість підходів не враховують концепції та вимоги різних стандартів інформаційної безпеки, що може викликати недовіру до застосовуваних підходів у експертів, які проводять аналіз ризиків, та ускладнити можливу сертифікацію організації. Для реалізації вдосконаленого підходу до управління ризиками авторами було проаналізовано розподілену інформаційну систему з точки зору інформаційної безпеки компаній, а також були описані особливості об'єкта, який розглядається. Також було побудовано модель загроз згідно з керівними документами, проведено збір експертної інформації для визначення імовірності кожної з ідентифікованих загроз та проведено розрахунок ризиків для системи, що розглядається.

Ключові слова: модель загроз, інформаційна безпека, управління ризиками, розподілені інформаційні системи.

VIRA TITOVA, YURII KLOTS, OLEKSII NAHREBETSKYI
Khmelnitskyi National University

INFORMATION SECURITY RISK MANAGEMENT IN DISTRIBUTED INFORMATION SYSTEMS OF PERSONAL DATA PROCESSING

A distributed information system is a set of interactive software modules that have a single system of users. In fact, any system that jointly processes data between two or more computers is a distributed computing system. It is used to reduce the load on the server and ensure the normal operation of the remote department.

In this work, a distributed information system was developed on the example of a free private company with a staff of 150 employees and an economy and finance department, which includes two automated workplaces that process personal data and a server for storing personal data, which, in turn, are included in the general local network of the enterprise, which has access to the Internet. The authors conducted an analysis of methodological and risk prevention documents, as well as the existing object of information activity from the point of view of information protection.

A threat model has been built, according to regulatory documents. The most likely threats to the selected organization were identified and the attack vector for these threats was determined. At the next stage, expert information was collected to determine the probability of the identified threats and risk calculations were carried out for the systems under consideration. As a result, the goal of the work, namely the implementation of a more advanced

approach to risk management and, accordingly, IS in distributed information systems for personal data processing, was achieved.

Keywords: threat model, information security, risk management, distributed information systems.

Постановка проблеми

Розподілена інформаційна система – це набір інтерактивних програмних модулів, які мають єдину систему користувачів. Фактично будь-яка система, яка спільно обробляє дані між двома або більше комп'ютерами, є розподіленою обчислювальною системою. Вона використовується для зниження навантаження на сервер та забезпечення нормальної роботи віддаленого відділу.

Кожен вузол розподіленої системи має бути незалежним або автономним. Локальна незалежність означає, що вузли розподіленої системи мають рівні права, тобто вважаються рівними. Це означає, що немає потреби викликати так званий центральний вузол або головний вузол для доступу до будь-яких централізованих служб.

Особливості з погляду захисту персональних даних під час їх обробки у розподілених системах полягають у наступному. Оператор при обробці персональних даних зобов'язаний вживати необхідних правових, організаційних та технічних заходів або забезпечувати їх прийняття для захисту персональних даних від неправомірного або випадкового доступу до них, знищення, зміни, блокування, копіювання, розповсюдження персональних даних, а також від інших неправомірних дій щодо персональних даних. Тобто роботу оператора при обробці персональних даних з точки зору інформаційної безпеки можна звести до вирішення задачі управління ризиками інформаційної безпеки.

Огляд існуючих рішень

Питанням аналізу підходів управління інформаційною безпекою (ІБ) присвячено велику кількість наукових праць, більшість із яких або містять велику кількість наявність математичних формул і моделей; або не містять взагалі ніяких математичних складових; або мають схильність в сторону якої-небудь із двох вище наведених груп підходів. Проаналізуємо змістовні аспекти кожної групи [1-3].

Підходи першої групи, як правило, використовують різні розділи вищої математики. В якості ядра підходів вибирають принципи, засновані на теорії імовірності або корисності (надійності), або нечітких множинах, неперервному чи дискретному розподілі, тощо. Роботи, що відносяться до першої групи підходів, досить часто не враховують реальні вимоги організацій, що займаються аналізом ризиків; вимагають від експертів в області ІБ достатньої математичної підготовки, що часто негативно відображається на практиці застосування даних підходів. Друга група підходів у більшій мірі розвинена зарубіжними авторами. Статті авторів із США, Англії мають перш за все рекомендаційний характер для модернізації на основі стандартів ІБ: ISO, BS, які вже зарекомендували себе, не вимагаючи глибоких знань вищої математики [4-6].

Третя група підходів у багатьох випадках поєднує в собі експертні оцінки та оцінки ризиків, що базуються на визначенні їх за наявними статистичними даними. Подібні підходи можна успішно застосовувати в практичній діяльності (не дивлячись на ряд мінусів), так як використання бази статистики дозволяє звести до мінімуму суб'єктивну точку зору експерта на вирішувану задачу і проводити роботу за оцінкою ризиків ІБ-спеціалістів без великого досвіду та кваліфікації.

Формулювання цілей статті

За результатами аналізу можна зробити наступні висновки: більшість підходів не враховують концепції та вимоги різних стандартів ІБ, що може викликати недовіру до застосовуваних підходів у експертів, які проводять аналіз ризиків ІБ, ускладнює можливу сертифікацію організації. Багато підходів, в основах яких лежить мета отримати кількісну оцінку ризиків з використанням математичних формул, моделей, заглиблених в математичні теорії, не мають практичний зв'язок з оцінкою ризиків, реальними вимогами бізнесу. Ряд підходів не забезпечує повного процесу оцінки, управління ризиками ІБ, реалізуючи лише деякі його компоненти. Враховуючи сильні та слабкі сторони існуючих підходів авторами було прийнято рішення реалізації більш досконалого підходу до управління ризиками та, відповідно, ІБ в розподілених інформаційних системах

обробки персональних даних.

Виклад основного матеріалу

Як об'єкт дослідження було обрано довільне приватне підприємство. Штат підприємства складає 150 працівників. У компанії є кілька відділів, кожен з яких здійснює свою діяльність. У цій роботі нас буде цікавити відділ економіки та фінансів, який включає 2 автоматизованих робочих місця, на яких відбувається обробка персональних даних, та сервер для зберігання персональних даних (рис. 1), які в свою чергу включені в загальну локальну мережу підприємства, що має вихід в мережу Internet. Саме на цьому відділі проводилися зазначені дослідження.

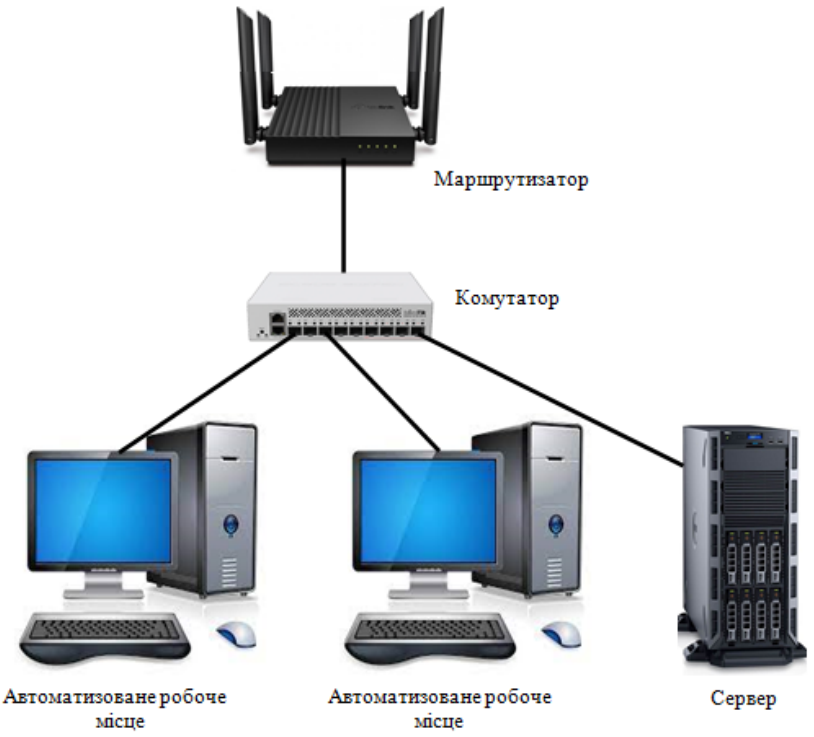


Рис. 1. Схема відділу для дослідження

Діяльність компанії пов'язана з використанням обчислювальної техніки та інформаційних технологій, до неї входить: тестування на відповідність вимогам захищеності від несанкціонованого доступу до інформації; аналіз уразливостей та контроль відсутності недекларованих можливостей у програмному забезпеченні; розробка та впровадження систем захисту інформації, що становить державну таємницю, на основі сертифікованих базових інформаційних захищених комп'ютерних технологій.

До переліку активів організації входить: обладнання – оцінна вартість 750 тис. грн., статутний капітал – 570 тис. грн., персональні дані працівників – оціночна вартість 250 тис. грн., будівлі, споруди – оціночна вартість 1850 тис. грн., програмні продукти – оціночна вартість 600 тис. грн.

Для досягнення поставленої мети в рамках компанії доцільно використовувати метод розрахунку ризиків, представлений в NIST 800-30 [7]:

$$R = P(t) * S, \quad (1)$$

де R – значення ризику; $P(t)$ – ймовірність реалізації загрози ІБ (застосовується якісна та кількісна шкали); S – ступінь впливу загрози на інформаційний актив (вартість активу в якісній та кількісній шкалі).

Визначимо усі можливі загрози інформаційним активам в розподілених інформаційних системах обробки персональних даних (табл.1).

Для розрахунків значення ризиків R необхідно мати значення $P(t)$ – визначимо його шляхом

експертної оцінки за шкалою 1-9. Причому 1 означає, що загрози рівнозначні, 2-9 – що одна загроза є більш імовірною ніж інша у зазначену кількість разів. Отримаємо зведену таблицю порівнянь загроз (табл.2).

Таблиця 1

Модель загроз ІБ

№	Тип загрози		Можливі наслідки
1	Аналіз мережного трафіку		Визначення характеристик мережного трафіку, перехоплення даних, що передаються, в тому числі ідентифікаторів та паролів користувачів
2	Сканування мережі		Визначення протоколів, доступних портів мережних служб, правил формування з'єднань, активних мережних сервісів, ідентифікаторів та паролів користувачів
3	«Парольна» атака		Виконання деструктивних дій, пов'язаних з отриманням несанкціонованого доступу
4	Підміна довіреного об'єкту мережі		Зміна маршруту проходження повідомлень, несанкціонована зміна маршрутно-адресних даних, несанкціонований доступ до мережних ресурсів, нав'язування недостовірної інформації.
5	Нав'язування хибного маршруту		Несанкціонована зміна маршрутно-адресних даних, аналіз та модифікація даних, що передаються, нав'язування хибних повідомлень
6	Впровадження хибного об'єкта мережі		Перехват та перегляд трафіку, несанкціонований доступ до мережних ресурсів, нав'язування недостовірної інформації.
7	Відмова обслуговуванні	Часткове вичерпання ресурсів	Зниження пропускної здатності каналів зв'язку, продуктивності мережних пристроїв, продуктивності серверних додатків
		Повне вичерпання ресурсів	Неможливість передачі повідомлень через відсутність доступу до середовища передачі, відмова у встановленні з'єднання, відмова в наданні сервісів (файловий сервер, пошта, тощо)
		Порушення логічних зв'язків між атрибутами, даними, об'єктами	Неможливість передачі повідомлень через відсутність коректних маршрутно-адресних даних, неможливість отримання послуг в зв'язку з несанкціонованою модифікацією ідентифікаторів, паролів, тощо.
		Використання помилок в програмах	Порушення робото здатності мережних пристроїв

8	Віддалений запуск додатків	Розсилання файлів з деструктивним кодом, вірусне зараження	Порушення конфіденційності, цілісності, доступності інформації
		Переповнення буферу	
		Використання можливостей віддаленого керування системою через приховані програмні/ апаратні закладки або штатними засобами	Приховане керування системою

Таблиця 2

Зведена таблиця порівнянь загроз

	31	32	33	34	35	36	37	38	$\sum$
31	1	1	1	1/7	1/7	1/7	1/5	1/3	**3,96**
32	1	1	1	1/7	1/7	1/7	1/5	1/3	**3,96**
33	1	1	1	1/7	1/7	1/7	1/5	1/3	**3,96**
34	7	7	7	1	1	1	3	3	**30**
35	7	7	7	1	1	1	3	3	**30**
36	7	7	7	1	1	1	3	3	**30**
37	5	5	5	1/3	1/3	1/3	1	1	**18**
38	3	3	3	1/3	1/3	1/3	1	1	**12**
									131,88

Провівши розрахунок вагового коефіцієнту кожної загрози шляхом нормування, отримаємо значення імовірності кожної загрози 31-38.

$P_1(t)$ (31 – Аналіз мережного трафіку) = 0,03; $P_2(t)$ (32 – Сканування мережі) = 0,03; $P_3(t)$ (33 – «Парольна» атака) = 0,03; $P_4(t)$ (34 – Підміна довіреного об’єкту мережі) = 0,23; $P_5(t)$ (35 – Нав’язування

хибного маршруту) = 0,23; $P_6(t)$ (З6 – Впровадження хибного об'єкта мережі) = 0,23; $P_7(t)$ (З7 – Відмова в обслуговуванні) = 0,14; $P_8(t)$ (З8 – Віддалений запуск додатків) = 0,09.

При цьому З1-З6 становлять загрозу для персональних даних працівників та програмних продуктів; З7 – становить загрозу для обладнання та програмних продуктів; З8 – тільки для програмних продуктів. Розрахуємо кількісне значення ризику для кожної загрози.

$R_1 = 25500$ грн.; $R_2 = 25500$ грн.; $R_3 = 25500$ грн.; $R_4 = 195500$ грн.; $R_5 = 195500$ грн.; $R_6 = 195500$ грн.; $R_7 = 189000$ грн.; $R_8 = 54000$ грн.

Ризики, які становлять менше 5% від статутного капіталу відкинемо, як такі, що не несуть суттєвих збитків та з легкістю можуть бути скомпенсовані. В нашому випадку таких ризиків, якими можна було б знехтувати, немає. Для всіх існуючих загроз З1-З8 слід застосувати заходи протії, щоб мінімізувати значення ризиків, які вони несуть. Найбільшої уваги потребують загрози З4-З7 (підміна довіреного об'єкту мережі, нав'язування хибного маршруту, впровадження хибного об'єкта мережі, відмова в обслуговуванні). При цьому вартість засобів та заходів захисту не повинна перевищувати суму ризику, який вони мінімізують.

Висновки

В даній роботі було розглянуто розподілену інформаційну систему на прикладі довільної приватної компанії зі штатом 150 працівників та відділом економіки та фінансів, який включає 2 автоматизованих робочих місця, на яких відбувається обробка персональних даних, та сервер для зберігання персональних даних, які в свою чергу включені в загальну локальну мережу підприємства, що має вихід в мережу Internet.

Авторами було проведено аналіз методичних документів із запобігання ризикам, а також наявного об'єкту інформаційної діяльності з погляду захисту інформації. Побудовано модель загроз, згідно з регламентуючими документами. Було визначено найбільш ймовірні загрози для обраної організації та визначено вектор атак для цих загроз. На наступному етапі було проведено збір експертної інформації для визначення імовірності кожної з ідентифікованих загроз та проведено розрахунок ризиків для системи, що розглядається.

В результаті мету роботи, а саме реалізацію більш досконалого підходу до управління ризиками та, відповідно, ІБ в розподілених інформаційних системах обробки персональних даних, було досягнуто.

Література

1. Методи якісного аналізу підприємницьких ризиків [Електронний ресурс] – Режим доступу: http://www.dut.edu.ua/uploads/l_50_49235071.pdf

2. Методичні основи оцінки ризиків підприємницької діяльності [Електронний ресурс] – Режим доступу: http://www.vestnikdnu.com.ua/archive/201154/171-176.pdf

3. Методичне забезпечення оцінки ризиків підприємства [Електронний ресурс] – Режим доступу: https://periodicals.karazin.ua/soceconom/article/download/4813/4366/#:~:text

4. Реалізація процесного підходу до керування ризиками інформаційної безпеки в документах NIST [Електронний ресурс] – Режим доступу: https://ela.kpi.ua/bitstream/123456789/23949/1/ITS2017.5.2(9)_09.pdf

5. Stango, Antonietta & Prasad, Neeli & Kyriazanos, Dimitris. (2009). A Threat Analysis Methodology for Security Evaluation and Enhancement Planning. 262-267. 10.1109/SECURWARE.2009.47.

6. Seeba, M., Mäses, S., Matulevičius, R. (2022). Method for Evaluating Information Security Level in Organisations. In: Guizzardi, R., Ralyté, J., Franch, X. (eds) Research Challenges in Information Science. RCIS 2022. Lecture Notes in Business Information Processing, vol 446. Springer, Cham. https://doi.org/10.1007/978-3-031-05760-1_39

7. Risk Management Guide for Information Technology Systems [Електронний ресурс] – Режим доступу: https://csrc.nist.gov/publications/detail/sp/800- 30/archive/2002-07-01

References

1. Metody yakisnoho analizu pidpryiemnytskykh ryzykiv [Elektronnyi resurs] – Rezhym dostupu: http://www.dut.edu.ua/uploads/l_50_49235071.pdf

2. Metodychni osnovy otsinky ryzykiv pidpryiemnytskoi diialnosti [Elektronnyi resurs] – Rezhym dostupu: http://www.vestnikdnu.com.ua/archive/201154/171-176.pdf

3. Metodychne zabezpechennia otsinky ryzykiv pidpryiemstva [Elektronnyi resurs] – Rezhym dostupu: https://periodicals.karazin.ua/soceconom/article/download/4813/4366/#:~:text

4. Realizatsiia protsesnoho pidkhodu do keruvannia ryzykamy informatsiinoi bezpeky v dokumentakh NIST [Elektronnyi resurs] – Rezhym dostupu: https://ela.kpi.ua/bitstream/123456789/23949/1/ITS2017.5.2(9)_09.pdf

5. Stango, Antonietta & Prasad, Neeli & Kyriazanos, Dimitris. (2009). A Threat Analysis Methodology for Security Evaluation and Enhancement Planning. 262-267. 10.1109/SECURWARE.2009.47.

6. Seeba, M., Mäses, S., Matulevičius, R. (2022). Method for Evaluating Information Security Level in Organisations. In: Guizzardi, R., Ralyté, J., Franch, X. (eds) Research Challenges in Information Science. RCIS 2022. Lecture Notes in Business Information Processing, vol 446. Springer, Cham. https://doi.org/10.1007/978-3-031-05760-1_39

7. Risk Management Guide for Information Technology Systems [Elektronnyi resurs] – Rezhym dostupu: https://csrc.nist.gov/publications/detail/sp/800- 30/archive/2002-07-01

Комплексна система захисту розподіленого програмного середовища передачі даних

Дипломна робота
студента КБм 22-1
Нагребецького О.В.

Вступ

- Розуміння розподілених програмних систем
- Проблеми в середовищі передачі даних
- Необхідність захисту розподіленого програмного забезпечення
- Основні загрози для розподілених програмних систем
- Розуміння концепції захисту даних
- Сучасні тенденції в захисті даних програмного забезпечення
- Важливість цілісності даних в розподілених системах
- Оцінка ризиків в розподілених програмних системах
- Стратегії захисту розподілених програмних систем
- Практичний приклад: Успішний захист розподіленої програмної системи
- Шифрування даних як захід безпеки
- Роль штучного інтелекту в захисті розподілених систем
- Розуміння фреймворків кібербезпеки
- Вплив GDPR на стратегії захисту даних
- Дослідження майбутнього захисту розподіленого програмного забезпечення
- Впровадження багатофакторної автентифікації
- Роль блокчейну в безпечній передачі даних
- Практичний кейс: Подолання викликів у сфері захисту даних
- Оцінка ефективності заходів захисту
- Висновок та наступні кроки для захисту розподіленого програмного забезпечення

Розуміння розподілених програмних систем

Розподілені програмні системи включають безліч взаємопов'язаних компонентів, розташованих у різних місцях.

Ці системи вимагають надійних заходів безпеки для захисту передачі даних і підтримки цілісності системи.

Розуміння архітектури та протоколів зв'язку має вирішальне значення для забезпечення надійності та безпеки розподілених програмних систем.

Виклики в середовищі передачі даних

1. Забезпечення безпеки та конфіденційності даних під час передачі.

2. Оптимізація швидкості та ефективності передачі даних для великомасштабного розподіленого програмного забезпечення.

3. Винесення програмованого функціоналу у захищене середовище передачі даних на основі базування на AMAZON

Необхідність захисту розповсюджуваного програмного забезпечення

Важливість

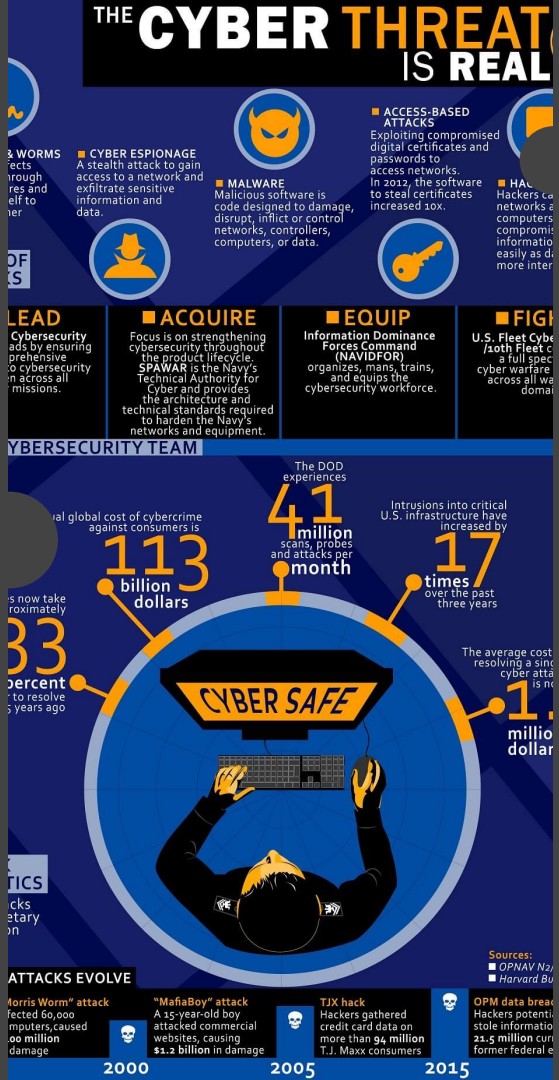
- Забезпечення цілісності та конфіденційності даних у розподілених системах.
- Захист від загроз безпеці та забезпечення відповідності нормам захисту даних.
- Підтримка безпечної передачі даних і зв'язку між розподіленими компонентами.

Виклики

- Керування складністю захисту декількох взаємопов'язаних компонентів.
- Забезпечення узгоджених заходів безпеки в розподілених системах.
- Адаптація до нових загроз і вразливостей безпеки в розподілених середовищах.

Основні загрози для розподілених програмних систем

- Витоки даних та несанкціонований доступ
- Мережеві вразливості та атаки
- Ризики цілісності та конфіденційності даних
- Атаки на відмову в обслуговуванні
- Внутрішньо-системні помилки
- Втрата зв’язку
- Неправильний менедмент у подальшому керуванні та експлуатації



Розуміння концепції захисту даних

Ключові міркування щодо захисту

При захисті передачі даних розподіленого програмного забезпечення такі фактори, як шифрування, контроль доступу, безпечні протоколи зв'язку та перевірка цілісності даних, є вирішальними для комплексної системи захисту.

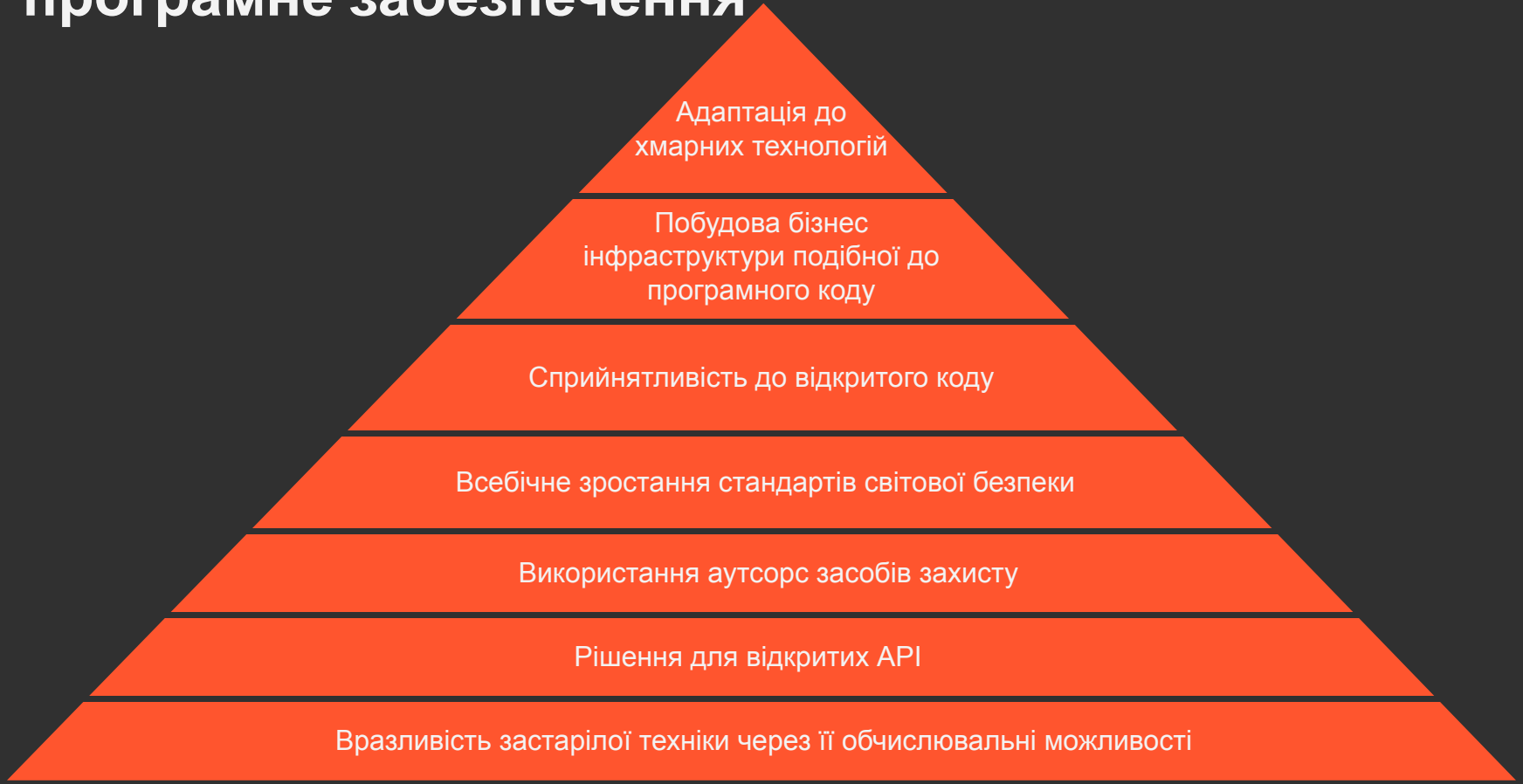
Важливість захисту даних

Захист даних має важливе значення для забезпечення конфіденційності, цілісності та доступності даних під час передачі, зберігання та обробки в розподіленому програмному середовищі.

Виклики у сфері захисту даних

Такі проблеми, як витік даних, дотримання нормативних вимог щодо захисту даних та управління даними в розподілених системах, створюють значні перешкоди для підтримки надійних заходів захисту даних.

Сучасні тенденції у сфері захисту даних про програмне забезпечення



Важливість цілісності даних у розподілених системах

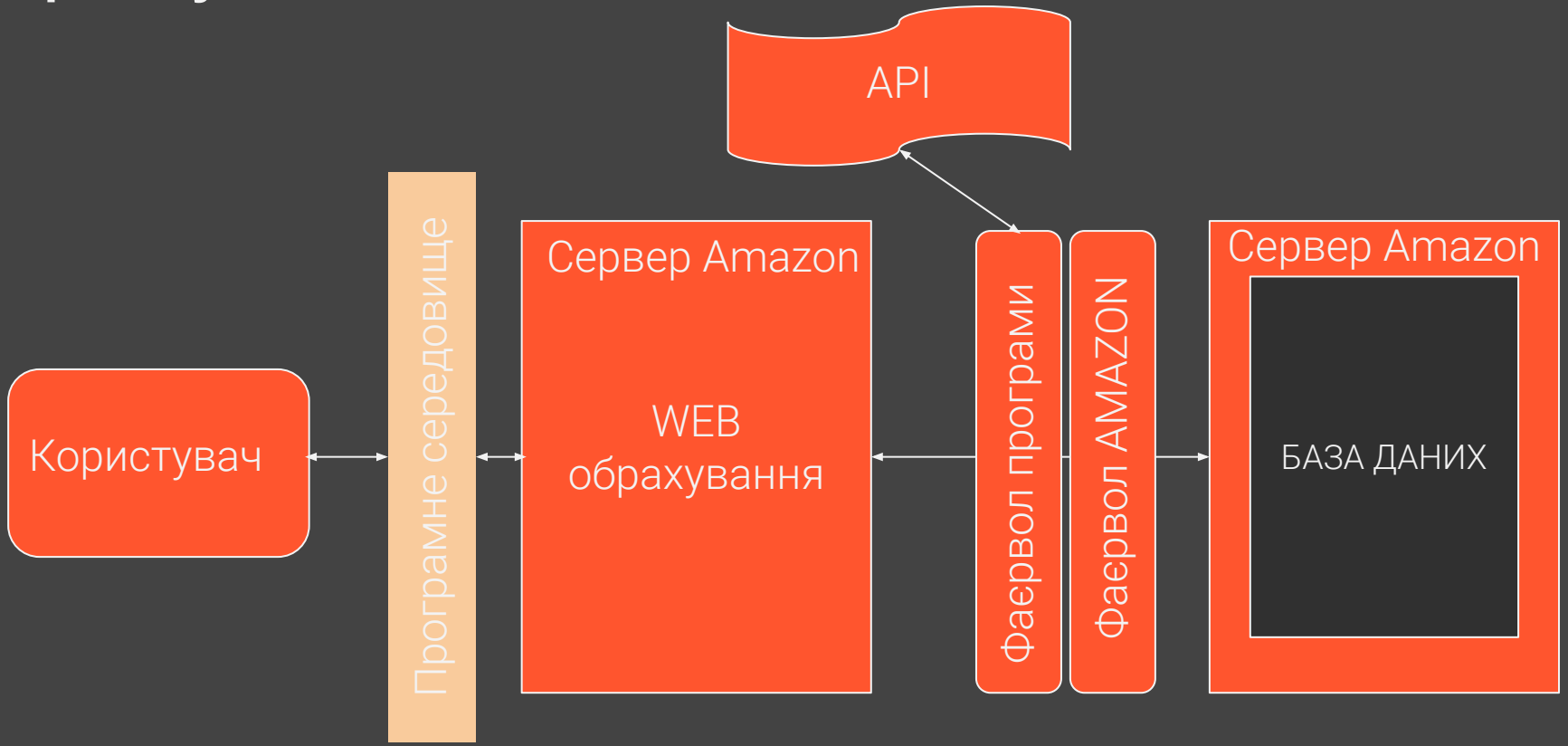
Цілісність даних

- Забезпечує точність, повноту та надійність даних протягом усього їхнього життєвого циклу.
- Запобігає несанкціонованому доступу, фальсифікації або пошкодженню даних під час передачі.
- Підтримує узгодженість і цілісність даних на розподілених вузлах.

Розподілені системи

- Дані передаються та обробляються через декілька вузлів та локацій.
- Потребує надійних механізмів для підтримки цілісності та безпеки даних у мережі.
- Виклики включають затримку, збої в мережі та проблеми синхронізації.

Схематичне зображення загальної стратегії захисту проекту



Практичний кейс: Успішний захист розподіленої програмної системи завдяки

Стратегія Захисту Системи

Система була побудована за правилом “можливе лише дозволене”. Функціонал наданий користувачеві є виключним, йому нічого не відомо, що відбувається всередині та як усе працює.

Безпека передачі даних

Використання захищених протоколів, шифрування та перевірки цілісності даних для забезпечення безпеки передачі даних у розподіленій програмній системі.

Шифрування даних як захід безпеки

Узагальнено	Шифрування даних є важливим заходом безпеки, який забезпечує захист конфіденційних даних під час передачі. Воно передбачає кодування даних таким чином, що лише уповноважені особи можуть мати доступ до них і розуміти їх.
Алгоритми шифрування	Було обрано шифрування даних на основі ЕКК (еліптичних кривих) з динамічно сформованим ядром на базі унікальних даних користувачів
Управління ключами	Ефективне управління ключами має важливе значення для шифрування даних. Воно передбачає безпечну генерацію, зберігання та розповсюдження ключів шифрування серед уповноважених осіб, забезпечуючи конфіденційність та цілісність зашифрованих даних.

Роль штучного інтелекту в захисті розподілених систем

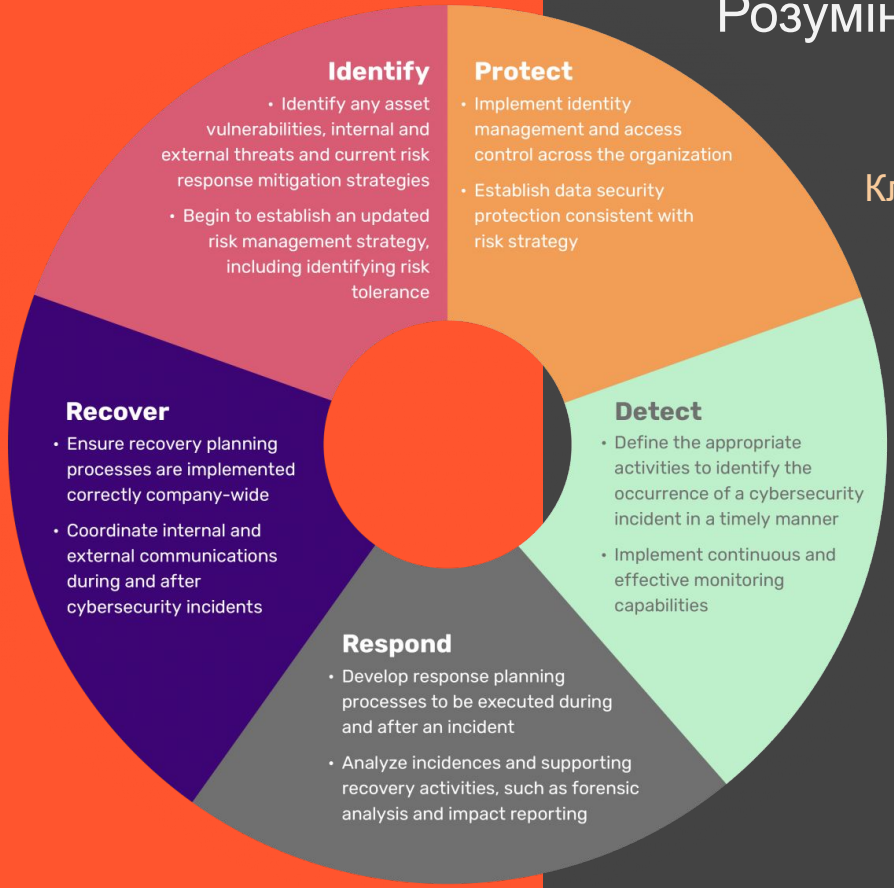
Переваги штучного інтелекту в захисті

- ШІ може виявляти загрози безпеці та реагувати на них у режимі реального часу.
- Він дозволяє проводити предиктивний аналіз для виявлення потенційних вразливостей.
- ШІ може автоматизувати процеси безпеки, зменшуючи кількість людських помилок.

Проблеми впровадження штучного інтелекту

- ШІ потребує великих масивів даних для ефективного навчання та прийняття рішень.
- Існують побоювання щодо етичності використання ШІ в системах безпеки.
- Складність алгоритмів ШІ може створювати проблеми з інтеграцією.

Розуміння основ кібербезпеки

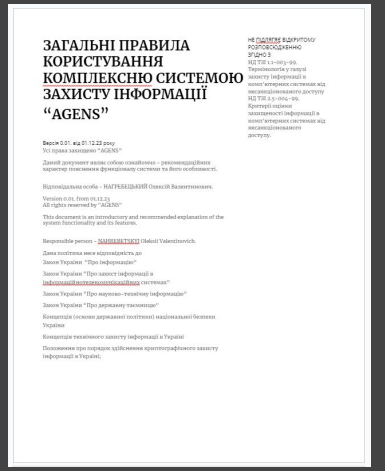


Ключові елементи системи

- Ідентифікація будь-яких дій у системі
- Пошук потенційних слабкостей
- Імплементація систем контролю та логування
- Система моніторингу помилок, несанкціонованого доступу, підозрілих дій користувача

Організація співпраці людини та КСЗД

1. Створена укорочена “Політика Безпеки”
2. Створені “Пам’ятки безпеки” ознайомчого характеру з фунціоналом



ЗАГАЛЬНІ ПРАВИЛА КОРИСТУВАННЯ КОМПЛЕКСНОЮ СИСТЕМОЮ ЗАХИСТУ ІНФОРМАЦІЇ “AGENS”

НЕ ПІДЛЯГАЄ ВІДКРИТОМУ РОЗПОВСЮДЖЕННЮ ЗГІДНО З НД ТЗІ 1.1-003-99. Термінологія у галузі захисту інформації в комп’ютерних системах від несанкціонованого доступу НД ТЗІ 2.5-004-99. Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу.

Версія 0.01. від 01.12.22 року
Усі права захищено “AGENS”

Даний документ являє собою ознайомчо - рекомендаційних характер пояснення функціоналу системи та його особливості.

Відповідальна особа - НАГРЕБЕЦЬКИЙ Олексій Валентинович.

Version 0.01. from 01.12.23
All rights reserved by “AGENS”

This document is an introductory and recommended explanation of the system functionality and its features.

Responsible person - НАГРЕБЕЦЬКИЙ Oleksii Valentinovich.

Дана політика має відповідність до

Закон України “Про інформацію”

Закон України “Про захист інформації в інформаційно-телекомунікаційних системах”

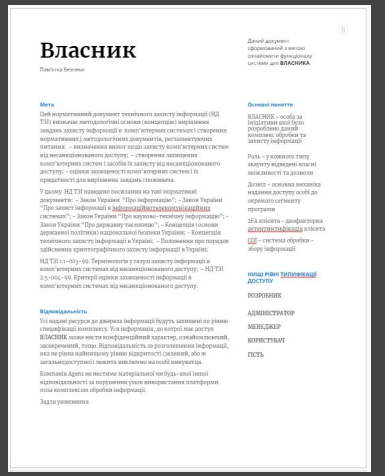
Закон України “Про науково-технічну інформацію”

Закон України “Про державну таємницю”

Концепція (основи державної політики) національної безпеки України

Концепція технічного захисту інформації в Україні

Положення про порядок здійснення криптографічного захисту інформації в Україні;



Власник

Памятка безпеки

Даний документ сформований з метою ознайомити функціоналу системи для ВЛАСНИКА

Мета

Цей нормативний документ технічного захисту інформації (НД ТЗІ) викладає методологічні основи (концепцію) вирішення завдань захисту інформації в комп’ютерних системах і створення нормативних і методологічних документів, регламентуючих питання: - визначення вимог щодо захисту комп’ютерних систем від несанкціонованого доступу; - створення захищених комп’ютерних систем і засобів їх захисту від несанкціонованого доступу; - оцінки захищеності комп’ютерних систем і їх придатності для вирішення завдань споживача.

У цьому НД ТЗІ наведено посилання на такі нормативні документи: - Закон України “Про інформацію”; - Закон України “Про захист інформації в інформаційно-телекомунікаційних системах”; - Закон України “Про науково-технічну інформацію”; - Закон України “Про державну таємницю”; - Концепція (основи державної політики) національної безпеки України; - Концепція технічного захисту інформації в Україні; - Положення про порядок здійснення криптографічного захисту інформації в Україні;

НД ТЗІ 1.1-013-99. Термінологія у галузі захисту інформації в комп’ютерних системах від несанкціонованого доступу; - НД ТЗІ 2.5-004-99. Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу.

Відповідальність

Усі надані ресурси до джерела інформації будуть захищені по рівню специфікації комплексу. Уся інформація, до котрої має доступ ВЛАСНИК може мести конфіденційний характер, ознайомлюючий, закритий, тощо. Відповідальність за розголошення інформації, яка не рівна найвищому рівню відкритості (зелений, або ж загальнодоступний) ложить виключно на особі власника.

Компанія Agens не несемає матеріальної чи буд.- якої іншої відповідальності за порушення умов використання платформи поза комплексом обробки інформації.

Задля уникнення

Основні поняття

ВЛАСНИК – особа за ініціативи якої було розроблено даний комплекс обробки та захисту інформації

Роль - у кожного типу акаунту відмінені власні можливості та дозволи

Дозвіл – основна механіка надання доступу особі до окремого сегменту програми

2FA клієнта – двофакторна аутентифікація клієнта

СОІ – система обробки - збору інформації

НИЖЧІ РІВНІ ТИПІФІКАЦІЇ ДОСТУПУ

РОЗРОБНИК

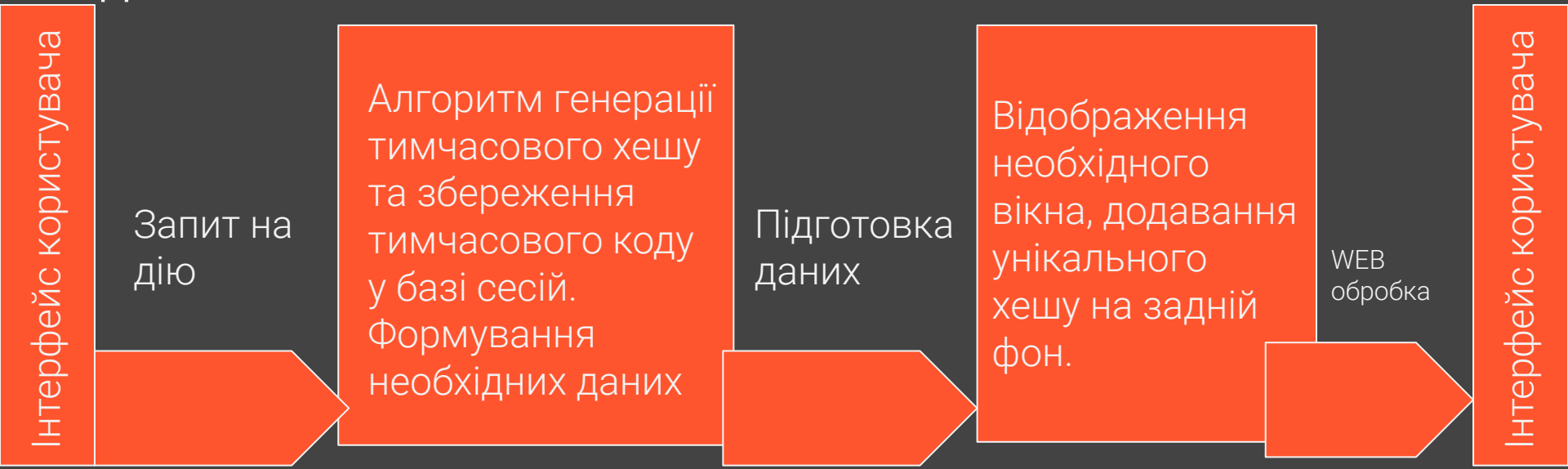
АДМІНІСТРАТОР

МЕНЕДЖЕР

КОРИСТУВАЧ

ГІСТЬ

Метод “Action Key” - це підтвердження дії реального користувача системи. Являється внутрішнім аналогом дії Captcha, його стабільність гарантується іншими підключеними системами.



ЗАГАЛЬНІ ПРАВИЛА КОРИСТУВАННЯ КОМПЛЕКСНОЮ СИСТЕМОЮ ЗАХИСТУ ІНФОРМАЦІЇ “AGENS”

НЕ ПІДЛЯГЯЄ ВІДКРИТОМУ РОЗПОВСЮДЖЕННЮ ЗГІДНО З
НД ТЗІ 1.1-003-99. Термінологія у галузі захисту інформації в комп'ютерних системах від несанкціонованого доступу
НД ТЗІ 2.5-004-99. Критерії оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу.

Версія 0.01. від 01.12.23 року

Даний документ являє собою ознайомчо - рекомендаційних характер пояснення функціоналу системи та його особливості.

Відповідальна особа - НАГРЕБЕЦЬКИЙ Олексій Валентинович.

Version 0.01. from 01.12.23

This document is an introductory and recommended explanation of the system functionality and its features.

Responsible person - NAHREBETSKYI Oleksii Valentinovich.

Дана політика несе відповідність до

Закон України “Про інформацію”

Закон України “Про захист інформації в інформаційнотелекомунікаційних системах”

Закон України “Про науково-технічну інформацію”

Закон України “Про державну таємницю"

Концепція (основи державної політики) національної безпеки України

Концепція технічного захисту інформації в Україні

Положення про порядок здійснення криптографічного захисту інформації в Україні;

Ця політика безпеки сформована AGENS DEVELOPMENT TEAM (далі - “AGENS”, "нас", "наш" або "ми").

У процесі своєї діяльності та з метою надання Послуг (як визначено в наших Загальних умовах використання), компанія Agens зобов'язана збирати та обробляти персональні дані своїх користувачів (далі - "Користувачі").

Ця політика конфіденційності, впроваджена компанією "АГЕНС", має на меті надати Користувачам стислий опис та огляд обробки персональних даних, що здійснюється компанією "АГЕНС".

Компанія "АГЕНС" надає особливого значення повазі до недоторканності приватного життя Користувачів та конфіденційності їхніх персональних даних, і тому зобов'язується обробляти дані відповідно до чинного законодавства, зокрема, Закону №. 78-17 від 6 січня 1978 року про інформаційні технології, файли даних і громадянські свободи (далі - "Закон про захист даних"), а також Регламенту (ЄС) 2016/679 Європейського парламенту і Ради від 27 квітня 2016 року про захист фізичних осіб у зв'язку з обробкою персональних даних і про вільний рух таких даних (далі - "GDPR").

ВАЛІДАЦІЯ

Функціонал валідації на інтернет-платформі грає ключову роль у забезпеченні безпеки, надійності та коректності введених даних користувачами. Основні завдання валідації включають перевірку формату, правильності та відповідності даних певним критеріям. Валідація може стосуватися різних типів даних і областей платформи, таких як реєстраційна інформація, текстові поля, фотографії, документи тощо.

Користувачі гарантовано отримають помилку валідації та побачать її у інтерфейсі. Якщо дані були введені вами коректно але ВИ (Користувач) нездатні пройти даний пункт захисту системи з будь-яких причин, зверніться до служби підтримки.

ПАРОЛІ

Використовуйте сильні паролі, які містять комбінацію великих і малих літер, цифр і спеціальних символів. Не використовуйте легко вгадувані дані, такі як ім'я, дата народження чи "пароль".

Використовуйте унікальні паролі для різних інтернет-сервісів і акаунтів. Це запобігає можливості компрометації всієї вашої інформації, якщо один пароль стає відомим.

Змінюйте паролі періодично, навіть якщо немає ознак компрометації вашого облікового запису.

ПРИЗНАЧЕННЯ

Надання рекомендацій роботи для **УСІХ** користувачів системи

ПІДТРИМУВАНІ МОВИ РОЗПОВСЮДЖЕННЯ

Українська

ВЕРСІЯ

v. 0.01 01.12.23

Двофакторна аутентифікація (2FA)

Ввімкніть двофакторну аутентифікацію, дане вікно ви можете знайти у ВАШОМУ персональномі вікні налаштувань.. Це надає додатковий шар захисту, вимагаючи додатковий крок для входу.

Перевірка URL-адрес)

Переконайтеся, що ви користуєтеся безпечними і правильними URL-адресами під час введення конфіденційної інформації, особливо при введенні фінансових даних.

РОЗПОВСЮДЖЕННЯ ДАНИХ ІНТЕЛЕКТУАЛЬНОЇ ВЛАСНОСТІ

Дана платформа створена з метою розповсюдження текстовими даними у простір відкритий для відвідувачів системи, у тому числі незареєстрованими гостями. У випадку, якщо ви бажаєте виділити окрему групу читачів, ВИ матимете змогу обрати таке налаштування. Якщо дані, були розповсюджені методами платформи, а ВИ не планували цього робити, МИ не нестимо відповідальність за ВАШІ налаштування. Функціонал та відповідні механізми розповсюдження - гарантовані. У випадку копіювання, збереження, зміни, видалення, чи будь-яких інших дій, поза межами функціоналу комплексу, або ж діями окремих користувачів, компанія не нестиме відповідальності за ваші дані, якщо не були використані методи чи функціональність комплексу.

ПРОСИМО ОЗНАЙОМИТИСЬ ДО ПОЧАТКУ РОБОТИ З ПЛАТФОРМОЮ ТА СЛІДКУВАТИ ЗА ЗМІНАМИ У ЦЬОМУ ДОКУМЕНТІ.

Менеджер

Пам’ятка безпеки

Даний документ сформований з метою ознайомити функціоналу системи для **МЕНЕДЖЕРА**

Мета

Цей нормативний документ технічного захисту інформації (НД ТЗІ) визначає методологічні основи (концепцію) вирішення завдань захисту інформації в комп'ютерних системах і створення нормативних і методологічних документів, регламентуючих питання: - визначення вимог щодо захисту комп'ютерних систем від несанкціонованого доступу; - створення захищених комп'ютерних систем і засобів їх захисту від несанкціонованого доступу; - оцінки захищеності комп'ютерних систем і їх придатності для вирішення завдань споживача.

У цьому НД ТЗІ наведено посилання на такі нормативні документи: - Закон України “Про інформацію”; - Закон України “Про захист інформації в інформаційнотелекомунікаційних системах”; - Закон України “Про науково-технічну інформацію”; - Закон України “Про державну таємницю"; - Концепція (основи державної політики) національної безпеки України; - Концепція технічного захисту інформації в Україні; - Положення про порядок здійснення криптографічного захисту інформації в Україні;

НД ТЗІ 1.1-003-99. Термінологія у галузі захисту інформації в комп'ютерних системах від несанкціонованого доступу; - НД ТЗІ 2.5-004-99. Критерії оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу.

Відповідальність

Усі надані ресурси до джерела інформації будуть захищені по рівню специфікації комплексу. Уся інформація, до котрої має доступ **МЕНЕДЖЕР,** може нести конфіденційний характер, ознайомлюючий, засекречений, тощо. Відповідальність за розголешення інформації, яка не рівна найнищому рівню відкритості (зелений, або ж загальнодоступної) лежить виключно на особі винуватця.

Компанія Agens не нестиме матеріальної чи будь-якої іншої відповідальності за порушення умов використання платформи поза комплексом обробки інформації.

Задля уникнення

Основні поняття

Менеджер - особа яка має доступ та права у межах акаунту у комплексі обробки та захисту інформації

Роль - у кожного типу акаунту відведені власні можливості та дозволи

Дозвіл - основна механіка надання доступу особі до окремого сегменту програми

2FA клієнта - двофакторна аутентинтифікація клієнта

COI - система обробки - збору інформації

НИЩІ РІВНІ ТИПИФІКАЦІЇ ДОСТУПУ

КОРИСТУВАЧ

ГІСТЬ

Прив’язаність до проекту (проектів)

Дана політика відповідає умовам проекту (проектів) від девелопмент відділу Agens, несе виключно ознайомчий та рекомендаційний характер без прямих зобов'язань. Дану пам’ятку можна використовувати на усіх відповідних посадах, пов’язаних з комплексною системою захисту інформації Agens. За невиконання умов, відповідальність лягає на особу, котра свідомо відмовилась від ознайомлення з даною пам’яткою, або ж свідомо допустила ігнорування пунктів даного документу

З пам’яткою ознайомлений (ознайомлена)

ПІБ

ДАТА

ПІДПИС

ЗАБОРОНЯЄТЬСЯ

- ☐ СВІДОМЕ НАДАННЯ ДОСТУПУ ДО **СОІ** НЕВІДОМИХ СТОРІН. У даному пункті означено розуміння передачі особистих паролів, логів, критографічних ключів, котрі можуть відкрити доступ до інформації системи
- ☐ НАДАННЯ ДОЗВОЛУ ВИЩЕ ЗА НЕОБХІДНЕ. У випадку порушення відповідності даних через діяльність осіб, котрі мали доступ до редагування, видалення, копіювання, тощо, хоча не мали його мати, відповідальність лягатиме на особоу, котра надала доступ до подібного функціоналу
- ☐ ПОШИРЕННЯ КОНФІДЕНЦІЙНОЇ ІНФОРМАЦІЇ З ПЛАТФОРМИ НЕПЕРЕДБАЧЕНИМИ У НІЙ ШЛЯХАМИ. У випадку розповсюдження, копіювання, збереження, тощо, інформації чи будь-якого іншого характеру інтелектуальної власності, відповідальність за такі дії лягатиме на особу винуватця.
- ☐ ОБМЕЖЕННЯ ФУНКЦІОНАЛУ ЧИ ЗКИДАННЯ МЕТОДІВ АУТЕНТИФІКАЦІЇ. У даній платформі реалізовані методи, з метою полегшити роботу осіб користувачів. У випадку зкидання 2ФА, паролю чи інших системних обмежень для користувача (акаунту), відповідальність лягатиме на особу виконавця

Усі інші пункти зазначені у документі загального користування платформою.

Програміст

Пам’ятка безпеки

Даний документ сформований з метою ознайомити функціоналу системи для **ПРОГРАМІСТА**

Мета

Цей нормативний документ технічного захисту інформації (НД ТЗІ) визначає методологічні основи (концепцію) вирішення завдань захисту інформації в комп'ютерних системах і створення нормативних і методологічних документів, регламентуючих питання: - визначення вимог щодо захисту комп'ютерних систем від несанкціонованого доступу; - створення захищених комп'ютерних систем і засобів їх захисту від несанкціонованого доступу; - оцінки захищеності комп'ютерних систем і їх придатності для вирішення завдань споживача.

У цьому НД ТЗІ наведено посилання на такі нормативні документи: - Закон України “Про інформацію”; - Закон України “Про захист інформації в інформаційнотелекомунікаційних системах”; - Закон України “Про науково-технічну інформацію”; - Закон України “Про державну таємницю"; - Концепція (основи державної політики) національної безпеки України; - Концепція технічного захисту інформації в Україні; - Положення про порядок здійснення криптографічного захисту інформації в Україні;

НД ТЗІ 1.1-003-99. Термінологія у галузі захисту інформації в комп'ютерних системах від несанкціонованого доступу; - НД ТЗІ 2.5-004-99. Критерії оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу.

Відповідальність

Усі надані ресурси до джерела інформації будуть захищені по рівню специфікації комплексу. Уся інформація, до котрої має доступ **ПРОГРАМІСТ** може нести конфіденційний характер, ознайомлюючий, засекречений, тощо. Відповідальність за розголешення інформації, яка не рівна найнищому рівню відкритості (зелений, або ж загальнодоступної) лежить виключно на особі винуватця.

Компанія Agens не нестиме матеріальної чи будь-якої іншої відповідальності за порушення умов використання платформи поза комплексом обробки інформації.

Основні поняття

ПРОГРАМІСТ - особа з прямим доступоп до методів обробки інформації та прямого допуску до баз даних

Роль - у кожного типу акаунту відведені власні можливості та дозволи

Дозвіл - основна механіка надання доступу особі до окремого сегменту програми

2FA клієнта - двофакторна аутентинтифікація клієнта

COI - система обробки - збору інформації

НИЩІ РІВНІ ТИПИФІКАЦІЇ ДОСТУПУ

МЕНЕДЖЕР

КОРИСТУВАЧ

ГІСТЬ

Прив’язаність до проекту (проектів)

Дана політика відповідає умовам проекту (проектів) від девелопмент відділу Agens, несе виключно ознайомчий та рекомендаційний характер без прямих зобов'язань. Дану пам’ятку можна використовувати на усіх відповідних посадах, пов’язаних з комплексною системою захисту інформації Agens. За невиконання умов, відповідальність лягає на особу, котра свідомо відмовилась від ознайомлення з даною пам’яткою, або ж свідомо допустила ігнорування пунктів даного документу

З пам’яткою ознайомлений (ознайомлена)

ПІБ

ДАТА

ПІДПИС

ЗАБОРОНЯЄТЬСЯ

- [ ] СВІДОМЕ НАДАННЯ ДОСТУПУ ДО **СОІ** НЕВІДОМИХ СТОРІН. У даному пункті означено розуміння передачі особистих паролів, логів, критографічних ключів, котрі можуть відкрити доступ до інформації системи
- [ ] НАДАННЯ ДОЗВОЛУ ВИЩЕ ЗА НЕОБХІДНЕ. У випадку порушення відповідності даних через діяльність осіб, котрі мали доступ до редагування, видалення, копіювання, тощо, хоча не мали його мати, відповідальність лягатиме на особоу, котра надала доступ до подібного функціоналу
- [ ] РОЗГОЛОШЕННЯ МЕХАНІЗМІВ РОБОТИ. Оскільки особа ВЛАСНИК, повинна мати розуміння функціоналу платформи, вона отримає відповідні ознайомчі матеріали, дана інформація являється “КРИТИЧНОЮ” і не надана з метою публічного розголосу, копіювання, використання поза межами проекту чи у відкритих проектах даної системи.
- [ ] ПОШИРЕННЯ КОНФІДЕНЦІЙНОЇ ІНФОРМАЦІЇ З ПЛАТФОРМИ НЕПЕРЕДБАЧЕНИМИ У НІЙ ШЛЯХАМИ. У випадку розповсюдження, копіювання, збереження, тощо, інформації чи будь-якого іншого характеру інтелектуальної власності, відповідальність за такі дії лягатиме на особу винуватця.
- [ ] ОБМЕЖЕННЯ ФУНКЦІОНАЛУ ЧИ ЗКИДАННЯ МЕТОДІВ АУТЕНТИФІКАЦІЇ. У даній платформі реалізовані методи, з метою полегшити роботу осіб користувачів. У випадку зкидання 2ФА, паролю чи інших системних обмежень для користувача (акаунту), відповідальність лягатиме на особу виконавця
- [ ] ПРЯМА РОБОТА З БАЗОЮ ДАНИХ. Забороняється копіювання, збереження, розповсюдження, будь-яка інша взаємодія з даними, збереженими у базі, без реальної необхідності та узгодження задачі з власником фірми чи відповідальної особи

Усі інші пункти зазначені у документі загального користування платформою.

Адміністратор

Пам’ятка безпеки

Даний документ сформований з метою ознайомити функціоналу системи для **АДМІНІСТРАТОРА**

Мета

Цей нормативний документ технічного захисту інформації (НД ТЗІ) визначає методологічні основи (концепцію) вирішення завдань захисту інформації в комп'ютерних системах і створення нормативних і методологічних документів, регламентуючих питання: - визначення вимог щодо захисту комп'ютерних систем від несанкціонованого доступу; - створення захищених комп'ютерних систем і засобів їх захисту від несанкціонованого доступу; - оцінки захищеності комп'ютерних систем і їх придатності для вирішення завдань споживача.

У цьому НД ТЗІ наведено посилання на такі нормативні документи: - Закон України “Про інформацію”; - Закон України “Про захист інформації в інформаційнотелекомунікаційних системах”; - Закон України “Про науково-технічну інформацію”; - Закон України “Про державну таємницю"; - Концепція (основи державної політики) національної безпеки України; - Концепція технічного захисту інформації в Україні; - Положення про порядок здійснення криптографічного захисту інформації в Україні;

НД ТЗІ 1.1-003-99. Термінологія у галузі захисту інформації в комп'ютерних системах від несанкціонованого доступу; - НД ТЗІ 2.5-004-99. Критерії оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу.

Відповідальність

Усі надані ресурси до джерела інформації будуть захищені по рівню специфікації комплексу. Уся інформація, до котрої має доступ **АДМІНІСТРАТОР** може нести конфіденційний характер, ознайомлюючий, засекречений, тощо. Відповідальність за розголешення інформації, яка не рівна найнищому рівню відкритості (зелений, або ж загальнодоступної) лежить виключно на особі винуватця.

Компанія Agens не нестиме матеріальної чи будь-якої іншої відповідальності за порушення умов використання платформи поза комплексом обробки інформації.

Основні поняття

АДМІНІСТРАТОР - особа з наданими йому особливими правами та доступом до конфіденційної інформації

Роль - у кожного типу акаунту відведені власні можливості та дозволи

Дозвіл - основна механіка надання доступу особі до окремого сегменту програми

2FA клієнта - двофакторна аутентинтифікація клієнта

COI - система обробки - збору інформації

НИЩІ РІВНІ ТИПИФІКАЦІЇ ДОСТУПУ

МЕНЕДЖЕР

КОРИСТУВАЧ

ГІСТЬ

Прив’язаність до проекту (проектів)

Дана політика відповідає умовам проекту (проектів) від девелопмент відділу Agens, несе виключно ознайомчий та рекомендаційний характер без прямих зобов'язань. Дану пам’ятку можна використовувати на усіх відповідних посадах, пов’язаних з комплексною системою захисту інформації Agens. За невиконання умов, відповідальність лягає на особу, котра свідомо відмовилась від ознайомлення з даною пам’яткою, або ж свідомо допустила ігнорування пунктів даного документу

З пам’яткою ознайомлений (ознайомлена)

ПІБ

ДАТА

ПІДПИС

ЗАБОРОНЯЄТЬСЯ

- [ ] СВІДОМЕ НАДАННЯ ДОСТУПУ ДО **СОІ** НЕВІДОМИХ СТОРІН. У даному пункті означено розуміння передачі особистих паролів, логів, критографічних ключів, котрі можуть відкрити доступ до інформації системи
- [ ] НАДАННЯ ДОЗВОЛУ ВИЩЕ ЗА НЕОБХІДНЕ. У випадку порушення відповідності даних через діяльність осіб, котрі мали доступ до редагування, видалення, копіювання, тощо, хоча не мали його мати, відповідальність лягатиме на особоу, котра надала доступ до подібного функціоналу
- [ ] РОЗГОЛОШЕННЯ МЕХАНІЗМІВ РОБОТИ. Оскільки особа ВЛАСНИК, повинна мати розуміння функціоналу платформи, вона отримає відповідні ознайомчі матеріали, дана інформація являється “КРИТИЧНОЮ” і не надана з метою публічного розголосу, копіювання, використання поза межами проекту чи у відкритих проектах даної системи.
- [ ] ПОШИРЕННЯ КОНФІДЕНЦІЙНОЇ ІНФОРМАЦІЇ З ПЛАТФОРМИ НЕПЕРЕДБАЧЕНИМИ У НІЙ ШЛЯХАМИ. У випадку розповсюдження, копіювання, збереження, тощо, інформації чи будь-якого іншого характеру інтелектуальної власності, відповідальність за такі дії лягатиме на особу винуватця.
- [ ] ОБМЕЖЕННЯ ФУНКЦІОНАЛУ ЧИ ЗКИДАННЯ МЕТОДІВ АУТЕНТИФІКАЦІЇ. У даній платформі реалізовані методи, з метою полегшити роботу осіб користувачів. У випадку зкидання 2ФА, паролю чи інших системних обмежень для користувача (акаунту), відповідальність лягатиме на особу виконавця
- [ ] ПРЯМА РОБОТА З БАЗОЮ ДАНИХ. Забороняється копіювання, збереження, розповсюдження, будь-яка інша взаємодія з даними, збереженими у базі, без реальної необхідності та узгодження задачі з власником фірми чи відповідальної особи

Усі інші пункти зазначені у документі загального користування платформою.

Менеджер

Пам’ятка безпеки

Даний документ сформований з метою ознайомити функціоналу системи для **МЕНЕДЖЕРА**

Мета

Цей нормативний документ технічного захисту інформації (НД ТЗІ) визначає методологічні основи (концепцію) вирішення завдань захисту інформації в комп'ютерних системах і створення нормативних і методологічних документів, регламентуючих питання: - визначення вимог щодо захисту комп'ютерних систем від несанкціонованого доступу; - створення захищених комп'ютерних систем і засобів їх захисту від несанкціонованого доступу; - оцінки захищеності комп'ютерних систем і їх придатності для вирішення завдань споживача.

У цьому НД ТЗІ наведено посилання на такі нормативні документи: - Закон України “Про інформацію”; - Закон України “Про захист інформації в інформаційнотелекомунікаційних системах”; - Закон України “Про науково-технічну інформацію”; - Закон України “Про державну таємницю"; - Концепція (основи державної політики) національної безпеки України; - Концепція технічного захисту інформації в Україні; - Положення про порядок здійснення криптографічного захисту інформації в Україні;

НД ТЗІ 1.1-003-99. Термінологія у галузі захисту інформації в комп'ютерних системах від несанкціонованого доступу; - НД ТЗІ 2.5-004-99. Критерії оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу.

Відповідальність

Усі надані ресурси до джерела інформації будуть захищені по рівню специфікації комплексу. Уся інформація, до котрої має доступ **МЕНЕДЖЕР** може нести конфіденційний характер, ознайомлюючий, засекречений, тощо. Відповідальність за розголешення інформації, яка не рівна найнищому рівню відкритості (зелений, або ж загальнодоступної) лежить виключно на особі винуватця.

Компанія Agens не нестиме матеріальної чи будь-якої іншої відповідальності за порушення умов використання платформи поза комплексом обробки інформації.

Основні поняття

МЕНЕДЖЕР - особа з наданими йому особливими правами та доступом до інформації

Роль - у кожного типу акаунту відведені власні можливості та дозволи

Дозвіл - основна механіка надання доступу особі до окремого сегменту програми

2FA клієнта - двофакторна аутентинтифікація клієнта

COI - система обробки - збору інформації

НИЩІ РІВНІ ТИПИФІКАЦІЇ ДОСТУПУ

МЕНЕДЖЕР

КОРИСТУВАЧ

ГІСТЬ

З пам’яткою ознайомлений

(ознайомлена)

ПІБ

ДАТА

ПІДПИС

Прив’язаність до проекту (проектів)

Дана політика відповідає умовам проекту (проектів) від девелопмент відділу Agens, несе виключно ознайомчий та рекомендаційний характер без прямих зобов'язань. Дану пам’ятку можна використовувати на усіх відповідних посадах, пов’язаних з комплексною системою захисту інформації Agens. За невиконання умов, відповідальність лягає на особу, котра свідомо відмовилась від ознайомлення з даною пам’яткою, або ж свідомо допустила ігнорування пунктів даного документу

ЗАБОРОНЯЄТЬСЯ

- [ ] СВІДОМЕ НАДАННЯ ДОСТУПУ ДО **СОІ** НЕВІДОМИХ СТОРІН. У даному пункті означено розуміння передачі особистих паролів, логів, критографічних ключів, котрі можуть відкрити доступ до інформації системи
- [ ] НАДАННЯ ДОЗВОЛУ ВИЩЕ ЗА НЕОБХІДНЕ. У випадку порушення відповідності даних через діяльність осіб, котрі мали доступ до редагування, видалення, копіювання, тощо, хоча не мали його мати, відповідальність лягатиме на особоу, котра надала доступ до подібного функціоналу
- [ ] РОЗГОЛОШЕННЯ МЕХАНІЗМІВ РОБОТИ. Оскільки особа ВЛАСНИК, повинна мати розуміння функціоналу платформи, вона отримає відповідні ознайомчі матеріали, дана інформація являється “КРИТИЧНОЮ” і не надана з метою публічного розголосу, копіювання, використання поза межами проекту чи у відкритих проектах даної системи.
- [ ] ПОШИРЕННЯ КОНФІДЕНЦІЙНОЇ ІНФОРМАЦІЇ З ПЛАТФОРМИ НЕПЕРЕДБАЧЕНИМИ У НІЙ ШЛЯХАМИ. У випадку розповсюдження, копіювання, збереження, тощо, інформації чи будь-якого іншого характеру інтелектуальної власності, відповідальність за такі дії лягатиме на особу винуватця.
- [ ] ОБМЕЖЕННЯ ФУНКЦІОНАЛУ ЧИ ЗКИДАННЯ МЕТОДІВ АУТЕНТИФІКАЦІЇ. У даній платформі реалізовані методи, з метою полегшити роботу осіб користувачів. У випадку зкидання 2ФА, паролю чи інших системних обмежень для користувача (акаунту), відповідальність лягатиме на особу виконавця
- [ ] ПРЯМА РОБОТА З БАЗОЮ ДАНИХ. Забороняється копіювання, збереження, розповсюдження, будь-яка інша взаємодія з даними, збереженими у базі, без реальної необхідності та узгодження задачі з власником фірми чи відповідальної особи

Усі інші пункти зазначені у документі загального користування платформою.

Завідувачу кафедри кібербезпеки
к.т.н., доц. Кльоцу Ю.П.
Нагребецького Олексія Валентиновича
ПІБ здобувача вищої освіти

Студента ФІТ, 2 курсу, групи КБм-22-1

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у хмельницькому національному університеті» від 31.08.2023, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений. Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

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

8.12.2023
дата

підпис

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 1.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 9%**

	Документ		Сумарний збіг по Базі Даних	
ID: 122219 Назва: Комплексна система захисту розподіленого програмного середовища передачі даних Додано в БД: 2023-12-08 Автора: Нагребецький О.В. Керівники: Тітова В. Ю. Консультанти: Опоненти:	Символи	Лексеми	Символи	Лексеми
	76799	1176	828 (1%)	15 (1%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

UNICHECK
by Turnitin

Ім'я користувача:
Кафедра кібербезпеки

ID перевірки:
1015985526

Дата перевірки:
08.12.2023 19:50:20 EET

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
08.12.2023 19:59:27 EET

ID користувача:
100008300

Назва документа: **Нагребецький на антиплагіат**

Кількість сторінок: 73 **Кількість слів:** 11912 **Кількість символів:** 94127 **Розмір файлу:** 2.19 MB **ID файлу:** 1015666331

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

7.08%
Схожість

Найбільша схожість: 5.22% з джерелом з Бібліотеки (ID файлу: 1011309100)

6.51% Джерела з Інтернету	59	Сторінка 75
5.63% Джерела з Бібліотеки	40	Сторінка 75

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи	1
Підозріле форматування	12 сторінок

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ

КАФЕДРИ КІБЕРБЕЗПЕКИ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:
Назва: Комплексна система захисту розподіленого програмного середовища

Автор: Нагребецький Олексій Валентинович
Спеціальність: 125 – Кібербезпека
Освітня програма: освітньо-професійна
Науковий керівник: Тітова Віра Юріївна, к.т.н, доцент
Після аналізу звіту подібності зроблено такий висновок:

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

Підтвердження:

Оригінальність тексту роботи за результатами перевірки системою Unichek складає 92,92%, оригінальність тексту роботи за результатами перевірки системою Anti-Plagiarism v-15.257 складає 99,0%.

Згідно з Положенням про систему забезпечення академічної доброчесності у ХНУ (https://khmnu.edu.ua/wp-content/uploads/normatyvni-dokumenty/polozhennya/pro-systemu-zabezpechennya-akademichnoyi-dobrochesnosti.pdf) така авторська робота, обсяг оригінального тексту у відсотках до загального обсягу матеріалу в якій складає 90-100 %, визнається роботою з високою унікальністю тексту.

Керівник роботи _____ В.Ю. Тітова

Гарант ОП _____ В.Ю. Тітова

Завідувач кафедри кібербезпеки _____ Ю. П. Кльоц

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

освітньо-кваліфікаційного рівня «магістр»

Студент Нагребецький Олексій Валентинович

Тема: «Комплексна система захисту розподіленого програмного середовища передачі даних»

Галузь знань 12 «*Інформаційні технології*» Спеціальність 125 «*Кібербезпека*»
Освітня програма «Кібербезпека»

Обсяг дипломної роботи освітньо-кваліфікаційного рівня «магістр»:
кількість листів креслень - ; кількість сторінок записки 83;

1. Короткий зміст КР та прийнятих рішень У кваліфікаційній роботі запропоновано комплексний підхід до захисту розподіленого програмного середовища передачі даних. Розглянуті існуючі рамки та стандарти кібербезпеки, якими організації керуються при створенні надійних механізмів захисту. Критично оцінюються такі рамки, як NIST Cybersecurity Framework, ISO/IEC 27001 та CIS Controls, досліджуються їхні сильні сторони, обмеження та застосовність у різних контекстах. Досліджуються нові технології: від штучного інтелекту і машинного навчання для виявлення загроз до блокчейну для захисту транзакцій. Критично розглядаються виклики та етичні міркування, пов'язані з впровадженням цих технологій.

2. Висновок про відповідність КР завданню Кваліфікаційна робота у повній мірі відповідає поставленому завданню як в теоретичній так і у практичній частині роботи.

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У вступі подана загальна характеристика та актуальність поставленої задачі. У першому розділі було досліджено предметну область у сфері побудови комплексного захищеного середовища, а саме: інформаційні ресурси, як об'єкт захисту інформації, інформаційні загрози та методи протидії ним, інформаційна безпека комерційних гігантів, програмна складова захищеного середовища обробки інформації. У другому розділі було розглянуто передпроєктне планування та облаштовування складових проекту, організаційні залежності системи. У третьому розділі проведено синтез захищеного комплексу обробки даних, створено захищену базу даних, імплементовано методи захисту для Laravel, реалізовано метод Action Token. Четвертий розділ присвячено тестуванню комплексної системи захисту інформації, опису основного функціоналу системи, оцінці економічної ефективності, проведенню пробних атак на створений сервіс за домопогою сторонніх програм.

4. Позитивні сторони кваліфікаційної роботи полягають у тому, що реалізований програмний продукт надає користувацький інтерфейс, повноцінне віддалене робоче середовище, функціонал обміну даних, робота за базою та оновленнями на сервері. Дана реалізація включає у собі сучасні практичні реальні можливості доступних технологій захисту та передачі інформації. Використання технологій PHP Laravel, можливостей зручних типів UI (Tailwind) та систем сповіщень на базі соціальних мереж максимально наближені до реальних проектів комплексних систем захисту розподіленого програмного середовища передачі даних.

5. Негативні сторони кваліфікаційної роботи: у роботі не наведено порівняльний аналіз існуючих рішень комплексної інформаційної безпеки розподілених програмних середовищ передачі даних з запропонованими автором рішеннями.

6. Оцінка графічного оформлення та пояснювальної записки роботи. Оформлення всіх матеріалів кваліфікаційної роботи є якісним, здійснене з дотриманням актуальних стандартів та інституційних положень ХНУ. Пояснювальна записка відповідає нормам щодо її оформлення як за структурою, так і за представленням і форматуванням матеріалу.

7. Відгук про роботу в цілому В загальному кваліфікаційна робота заслуговує позитивної оцінки. Весь матеріал кваліфікаційної роботи структурований, чіткий та послідовний. Усі розділи роботи послідовні та логічні, що дозволяє чітко розуміти викладений матеріал в рамках тематики кваліфікаційної роботи.

8. Інші зауваження -

9. Оцінка дипломної роботи Розглянувши позитивні та негативні сторони представленої кваліфікаційної роботи, можна зробити висновок, що робота заслуговує оцінки «добре».

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи) Завідувач кафедри автоматизації, комп'ютерно-інтегрованих технологій та робототехніки, доктор технічних наук, професор Мартинюк Валерій Володимирович.

« 8 » грудня 2023 року. _____ (підпис)