Харківський національний університет імені В.Н. Каразіна

Факультет комп'ютерних наук

Безпека інформаційних систем і технологій

«Допущено до захисту»

Зав.кафедрою БІСТ

Сватовський І.І._____

«  » червня 2023р.

**Пояснювальна записка**

до кваліфікаційної роботи бакалавра

спеціальність: 125 Кібербезпека

на тему: «Аналіз методів та засобів захисту сучасних баз даних»

оцінка «  »

Голова ЕК

Лемешко О.В._____

Керівник Єсін В.І._____

(прізвище та ініціали/підпис)

Рецензент Бобух В.А._____

(прізвище та ініціали/підпис)

Виконавець студентка групи КБ-41

Нго Кієу Лінь _____

(прізвище та ініціали/підпис)

Харків – 2023

# РЕФЕРАТ

Пояснювальна записка містить 56 сторінок,  5 рисунків, 2 таблиць, 0 додатків, 19 джерел.

*Метою дипломної роботи* є дослідження є дослідження актуальних методів, засобів, сучасних тенденцій та практик забезпечення безпеки баз даних, з наступним порівнянням ефективності різних заходів безпеки для пом'якшення найпоширеніших загроз та вироблення рекомендацій щодо використання наявних методів та засобів захисту сучасних баз даних залежно від умов їх застосування.

Для досягнення мети в роботі вирішуються наступні завдання:

- Визначення основних загроз безпеці актуальних для сучасних баз даних.
- Аналіз заходів безпеки та методів, які використовуються для захисту баз даних.
- Оціка ефективністі різних заходів безпеки.
- Порівняння різних підходів до забезпечення безпеки баз даних.
- Розробка рекомендацій щодо використання наявних методів та засобів захисту сучасних баз даних.

*Об’єктом дослідження* є процесс забезпечення безпеки баз даних.

*Предметом розробки* є методи та засоби захисту сучасних баз даних

*Методи дослідження*: аналітичний метод та порівняльний метод.

Аналітичний метод. Цей метод використовується для розбору та розуміння проблеми баз даних шляхом розкриття їх структури, характеристик та особливостей. Аналітичний метод допомагає виявити ключові аспекти баз даних і проаналізувати їх вплив на процеси і операції.

Порівняльний метод. Цей метод полягає в порівняльні різних методів та засобів баз даних за співними критеріями, такими як продуктивність,

надійність, масштабованість та інші. Порівняльний метод дозволяє оцінити переваги і недоліки різних методів і засобів баз даних і визначити найбільш підходящі для конкретних потреб і вимог.

Використання цих методів дозволяє провести всебічний аналіз сучасних методів і засобів баз даних, виявити їх переваги і недоліки, а також запропонувати рекомендації щодо поліпшення і оптимізації баз даних.

*Результатами* проведеної роботи є порівняння ефективності різних заходів безпеки для пом'якшення найпоширеніших загроз та вироблення рекомендацій щодо використання наявних методів та засобів захисту сучасних баз даних залежно від умов їх застосування. Розгляд особливості захисту баз даних в хмарах.

*Ключові слова*: БАЗИ ДАНИХ, ЗАХИСТ, SQL, ЗАХИСТ ІНФОРМАЦІЇ, SQL-ІН’ЄКЦІЇ, ШИФРУВАННЯ, АУДИТ ТА МОНІТОРИНГ, АВТЕНТИФІКАЦІЯ ТА АВТОРИЗАЦІЯ, ХМАРОВІ БАЗИ ДАНИХ, РЕЗЕРВНЕ КОПІЮВАННЯ ТА ВІДНОВЛЕННЯ.

# ABSTRACT

The explanatory note consists of 56 pages, 5 figures, 2 tables, 0 annexes, 19 sources.

*The aim of the thesis* is to investigate the current methods, tools, trends, and practices for ensuring the security of databases, with a subsequent comparison of the effectiveness of different security measures to mitigate common threats and provide recommendations for the use of existing methods and tools for protecting modern databases based on their application conditions.

To achieve this goal, the following tasks are addressed:

- Identifying the main security threats relevant to modern databases.
- Analyzing security measures and methods used for database protection.
- Evaluating the effectiveness of various security measures.
- Comparing different approaches to database security.
- Developing recommendations for the use of available methods and tools to protect modern databases.

*The subject matter* is the process of ensuring the security of databases.

*The scope of the study* is the methods and tools for protecting modern databases.

*Research methods:* The following methods were applied in this study: analytical method and comparative method.

Analytical method: This method is used to analyze and understand the issues related to databases by examining their structure, characteristics, and peculiarities. The analytical method helps to identify key aspects of databases and analyze their impact on processes and operations.

Comparative method: This method involves comparing different database methods and tools based on common criteria such as performance, reliability, scalability, and others. The comparative method allows for the evaluation of advantages and disadvantages of different database methods and tools and determines the most suitable ones for specific needs and requirements.

The use of these methods allows for a comprehensive analysis of modern database methods and tools, identifying their strengths and weaknesses, and providing recommendations for the improvement and optimization of databases.

*The results* of this study include the comparison of the effectiveness of different security measures to mitigate common threats and the development of recommendations for the use of existing methods and tools to protect modern databases based on their application conditions. The specific aspects of database security in cloud environments are also discussed.

*Keywords:* DATABASES, SECURITY, SQL, INFORMATION PROTECTION, SQL INJECTIONS, ENCRYPTION, AUDIT AND MONITORING, AUTHENTICATION AND AUTHORIZATION, CLOUD DATABASES, BACKUP AND RECOVERY.

# ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ

| | | |
|---|---|---|
| СКБД | - | Система Керування Базами Даних |
| БД | - | База Даних |
| SQL | - | Structured Query Language |
| URL | - | Uniform Resource Locator |
| HTML | - | HyperText Markup Language |
| ОС | - | Операційна Система |
| XSS | - | Cross-Site Scripting |
| DoS | - | Denial of Service |
| DDoS | - | Distributed Denial of Service |
| PCI | - | Payment Card Industry |
| DAM | - | Digital Asset Management |
| GDPR | - | General Data Protection Regulation |
| CCPA | - | California Consumer Privacy Act |
| HIPAA | - | Health Insurance Portability and Accountability Act |
| IRS 1075 | - | Internal Revenue Service |
| SOX | - | Sarbanes-Oxley Act |
| UK DPA | - | UK Data Protection Act |
| SLA | - | Service Level Agreement |
| PaaS | - | Platform as a Service |
| SOC2 | - | Service Organization Control Report 2 |
| FedRAMP | - | Federal Risk and Authorization Management Progam |
| EU GDPR | - | European Union General Data Protection Regulation |

# ВСТУП

У сучасному інформаційному суспільстві бази даних стають центральним елементом багатьох організаційних систем, що зберігають та обробляють великі обсяги даних. Забезпечення безпеки та захисту цих баз даних є критичним завданням, оскільки несанкціонований доступ до них може мати серйозні наслідки, включаючи втрату чутливої інформації, порушення конфіденційності даних, а також фінансові та репутаційні втрати для організації.

У зв'язку з постійним розвитком технологій і появою нових загроз, розробка та впровадження ефективних методів та засобів захисту сучасних баз даних стають все більш актуальними завданнями.

Актуальність теми дослідження. Тема, що розглядається є дуже актуальною в контексті сучасної інформаційної інфраструктури підприємств та організацій, оскільки зростає кількість кібератак і загроз безпеці даних. Зловмисники постійно шукають нові способи вторгнутися в бази даних, крадіжки чутливої інформації, розповсюдження шкідливого програмного забезпечення і шантажування. Захист баз даних є необхідною складовою їх безпеки, оскільки вони містять значну кількість важливої конфіденційної інформації про клієнтів, партнерів, фінансові операції та інші дані.

Дана робота спрямована на виявлення найкращих методов, механізмов, практик та інструментів для захисту даних у різних типах баз даних, включаючи реляційні, NoSQL та NewSQL. Для цього аналізуються різні загрозибезпеки баз даних, досліджуються новітні технології та методи, пов'язані з методами шифрування, контролю доступу, механізмамі автентифікація та деякими іншими способами та інструментами.

Практичне значення теми дослідження полягає в тому, що вона допоможе окремим особам та організаціям розуміти важливість захисту їхніх

баз даних і знайти оптимальні рішення для їхнього застосування. Компанії, які можуть ефективно захистити дані своїх клієнтів, забезпечують більшу довіру і лояльність, що сприяє успіху бізнесу. Урядові установи можуть також скористатися результатами дослідження для розробки політик та нормативних актів, що сприятимуть підвищенню ефективності захисту даних на національному рівні.

Результати дослідження можуть бути корисними для приватних осіб, підприємств та країн, які шукають найкращі підходи до захисту конфіденційної інформації у своїх базах даних.

# 1 ОСНОВИ ЗАХИСТУ БАЗ ДАНИХ

## 1.1 Роль баз даних у сучасному світі

Роль баз даних у сучасному світі є надзвичайно важливою і впливає на різні сфери життя і діяльності. Вони виступають основою для зберігання, керування та обробки великого обсягу інформації, що є ключовим ресурсом для багатьох організацій та інституцій. Ось декілька аспектів, які відображають роль баз даних у сучасному світі:

- Бізнес і економіка. Бази даних є основою для керування бізнес-операціями. Вони дозволяють зберігати і організовувати дані про клієнтів, продукти, транзакції, фінансову інформацію та інше. Бази даних сприяють прийняттю рішень на основі аналітики та допомагають в управлінні ресурсами, оптимізації процесів та плануванні.
- Наука і дослідження. В наукових дослідженнях бази даних використовуються для зберігання та обробки даних, які отримуються в результаті експериментів, спостережень та аналізу. Вони дозволяють науковцям збирати, порівнювати та аналізувати дані, виявляти закономірності і робити нові відкриття.
- Освіта. В установах освіти бази даних використовуються для зберігання студентських даних, розкладів, навчальних матеріалів, результатів тестувань та оцінок. Вони допомагають управляти учбовим процесом, створюють зручні умови для доступу до інформації та підтримують електронну обмін даними між викладачами та студентами.
- Організація і адміністрація. Бази даних є важливим інструментом для зберігання інформації про співробітників, клієнтів, процесів управління та інші аспекти організації. Вони спрощують управління ресурсами, автоматизують бізнес-процеси, підтримують звітність та аналітику.

- Інтернет та електронна комерція. Веб-застосунки та електронні магазини базуються на базах даних для зберігання інформації про товари, замовлення, користувачів та інше. Бази даних дозволяють створювати персоналізовані пропозиції, підтримувати електронну платіжну систему та забезпечувати ефективне функціонування електронної комерції.
- Сфера охорони здоров'я. В медичних установах бази даних використовуються для зберігання медичних записів пацієнтів, результатів лабораторних досліджень, медичних документів та інших даних. Це допомагає медичним працівникам забезпечувати ефективну медичну допомогу, вести моніторинг стану пацієнтів та аналізувати дані для наукових досліджень.
- Транспорт та логістика. Бази даних використовуються для керування даними про рух транспорту, розкладами, маршрутами, вантажами та іншою інформацією, що стосується транспортних систем. Вони дозволяють враховувати потреби клієнтів, планувати маршрути, визначати оптимальні шляхи доставки та підтримувати логістичні процеси.

Ці приклади лише дотикаються різноманітних сфер застосування баз даних у сучасному світі. Вони стали необхідним інструментом для ефективного управління даними та сприяють розвитку різних галузей, полегшуючи доступ до інформації, покращуючи прийняття рішень та забезпечуючи ефективну діяльність організацій та суспільства в цілому.

## 1.2 Системи керування базами даних

Система керування базами даних (СКБД) - це комплекс програмних і мовних засобів, необхідних для створення баз даних, підтримання їх в актуальному стані та організації пошуку в них необхідної інформації [13].

Системи керування базами даних (СКБД) можна класифікувати за їхнім типом і функціональністю. Основні типи СКБД включають реляційні, NoSQL, NewSQL СКБД.

- Реляційні СКБД є найпоширенішим типом і організовують дані у вигляді таблиць з рядками і стовпцями. Вони використовують мову запитів SQL для маніпулювання даними.

Використання СКБД має декілька переваг, включаючи підвищену узгодженість та точність даних, спрощення маніпулювання та отримання даних, а також підвищення безпеки даних шляхом контролю доступу та обмежень. Однак впровадження СКБД може бути витратним та вимагати спеціальних знань. Деякі типи СКБД можуть бути менш ефективними для певних типів даних, наприклад, неструктурованих даних.

Загалом, СКБД є найважливішими інструментами для управління даними в сучасному світі. Вони забезпечують надійний та організований спосіб зберігання, обробки та отримання даних, що допомагає організаціям приймати кращі рішення на основі точної інформації.

## 1.3 Криптографія та її роль в захисті баз даних

Криптографія – напрям у криптології, що вивчає основні закономірності, протиріччя, методи, системи та засоби забезпечення конфіденційності, цілісності, дійсності, доступності та спостережливості інформації та ресурсів тощо, ґрунтуючись на криптографічних перетвореннях [1].

Шифрування бази даних є важливим заходом у забезпеченні безпеки і конфіденційності даних. Цей процес використовує криптографію для перетворення звичайних текстових даних у незрозумілий формат, що називається шифротекстом. Шифрування використовує ключ для перетворення даних, а розшифрування може бути виконане лише за допомогою відповідного ключа.

Шифрування бази даних має декілька переваг. Воно дозволяє зберігати конфіденційну інформацію в зашифрованому вигляді, зменшуючи ризик несанкціонованого доступу до даних, якщо база даних стає доступною для зловмисників. Шифрування також може бути використане для виконання вимог щодо безпеки даних, які встановлені регуляторами або стандартами, наприклад, в рамках Загального регламенту про захист персональних даних (GDPR).

Крім шифрування, криптографія також використовується для автентифікації користувачів бази даних. Це означає, що користувачі повинні надати правильне ім'я користувача та пароль для отримання доступу до бази даних. Паролі зберігаються у зашифрованому вигляді, що ускладнює зловмисникам отримання несанкціонованого доступу.

Криптографія також допомагає забезпечити цілісність даних. Вона використовує різні алгоритми і методи для захисту інформації від несанкціонованого доступу і змін. Ось кілька способів, як криптографія допомагає забезпечити цілісність даних:

Шифрування даних: шифрування - це процес перетворення даних у зашифрований вигляд за допомогою спеціальних алгоритмів та ключів. Зашифровані дані неможливо зрозуміти або змінити без знання правильного ключа або пароля. Шифрування допомагає захистити дані від несанкціонованого доступу та незаконних змін.

Цифрові підписи: цифрові підписи використовуються для перевірки цілісності даних та підтвердження їх авторства. Цифровий підпис створюється за допомогою криптографічної функції, яка гарантує, що дані не були змінені після створення підпису. Це дозволяє перевірити, що дані не підроблені і залишаються цілісними.

Хеш-функції. Хеш-функції перетворюють дані в унікальний рядок фіксованої довжини, відомий як хеш-значення або контрольна сума. Будь-яка зміна вихідних даних призведе до зміни хеш-значення. При перевірці

цілісності даних, можна порівняти отримане хеш-значення з спочатку згенерованим, щоб переконатися, що дані залишилися незмінними.

Контрольні суми. Контрольні суми використовуються для виявлення випадкових помилок або змін у даних. Контрольна сума являє собою Числове значення, яке розраховується на основі вмісту даних. При отриманні даних, одержувач може розрахувати контрольну суму і порівняти її з відправленим значенням, щоб перевірити цілісність даних.

Комбіноване використання цих криптографічних методів дозволяє забезпечити цілісність даних, захищаючи їх від несанкціонованого доступу, змін або пошкоджень. Це важливо для підтримки конфіденційності та надійності інформації. Узагальнюючи, шифрування є важливими компонентами забезпечення безпеки баз даних. Вони допомагають зберегти конфіденційність, запобігти несанкціонованому доступу та забезпечити цілісність даних, що є вирішальними аспектами у сучасному цифровому середовищі.

## 1.4 Рівні захисту баз даних

Рівні захисту баз даних – це система заходів та технологій, які застосовуються для забезпечення безпеки інформації в базі даних від несанкціонованого доступу, втручання, пошкодження або втрати даних.

Існує кілька рівнів захисту баз даних, які забезпечують різні рівні безпеки та захисту. Нижче наведено короткий опис кожного рівня захисту баз даних:

Фізичний захист. Цей рівень захисту включає фізичні заходи для забезпечення безпеки серверів баз даних, таких як контроль доступу до приміщень, встановлення систем відеоспостереження, інтелектуальних замків тощо.

Захист мережі. Цей рівень захисту включає заходи для захисту мережі, на якій працює база даних. Він включає захист мережевих точок доступу,

захист від атак на мережу, захист від вірусів та шкідливих програм, інші заходи для забезпечення безпеки мережі.

Захист доступу. Цей рівень захисту включає заходи для забезпечення безпеки входу до системи баз даних. Він включає автентифікацію користувача, контроль доступу до баз даних, шифрування даних, захист від атак переповнення буфера і інші заходи.

Захист даних. Цей рівень захисту включає заходи для забезпечення безпеки даних в базі даних. Він включає шифрування даних, резервне копіювання, забезпечення цілісності даних, моніторинг доступу до даних і інші заходи.

Захист застосунків: цей рівень захисту включає заходи для забезпечення безпеки застосунків, які використовують базу даних. Він включає перевірку безпеки програмного забезпечення, контроль цілісності даних, перевірку вводу даних і інші заходи для забезпечення безпеки застосунків.

Захист на рівні операційної системи: цей рівень захисту включає заходи для забезпечення безпеки операційної системи, на якій працює база даних. Це можуть бути різноманітні конфігураційні параметри, політики безпеки, права доступу до файлів та інших ресурсів операційної системи, захист від шкідливих програм та інших загроз на рівні операційної системи.

Захист на рівні бази даних: цей рівень захисту включає заходи для забезпечення безпеки самої бази даних, такі як настройка прав доступу до об'єктів бази даних, встановлення політик автентифікації та авторизації, захист від SQL-ін'єкцій та інших атак, захист від несанкціонованого змінення даних та інших загроз на рівні бази даних.

Захист даних в режимі передачі: цей рівень захисту включає заходи для захисту даних під час їх передачі через мережу, такі як шифрування даних, захист від атак на мережевий трафік, встановлення політик автентифікації та авторизації для доступу до мережевих ресурсів, захист від віддалених атак та інших загроз під час передачі даних через мережу.

Успішна реалізація заходів на кожному рівні захисту дозволяє забезпечити високий рівень безпеки баз даних і унеможливити несанкціонований доступ до конфіденційної інформації. Крім того, вона дозволяє зменшити ризики втрати даних, внесення помилок в базу даних, втрати даних під час їх передачі та інших проблем, які можуть вплинути на роботу бази даних.

Успішна реалізація заходів на кожному рівні захисту дозволяє забезпечити високий рівень безпеки баз даних і унеможливити несанкціонований доступ до конфіденційної інформації. Крім того, вона дозволяє зменшити ризики втрати даних, внесення помилок в базу даних, втрати даних під час їх передачі та інших проблем, які можуть вплинути на роботу бази даних.

Загальний підхід до захисту баз даних передбачає комплексне застосування заходів на різних рівнях захисту, включаючи захист на рівні фізичної інфраструктури, операційної системи, бази даних, застосунків та мережевого забезпечення. Також важливо забезпечити регулярне оновлення захисту та моніторинг системи на предмет виявлення потенційних загроз і вразливостей.

Найкращою практикою є застосування стандартів безпеки та керування ризиками, таких як ISO/IEC 27001, які допоможуть забезпечити ефективну систему захисту баз даних і зменшити ризики їхнього порушення.

# 2 ВРАЗЛИВОСТІ СУЧАСНИХ БАЗ ДАНИХ

## 2.1 SQL-ін’єкції

Атаки SQL-ін’єкцій прості за своєю природою – зловмисник передає програмі вхідні дані в надії маніпулювати оператором SQL на свою користь. Складність атаки передбачає використання оператора SQL, який може бути невідомий зловмиснику. Програми з відкритим кодом і комерційні програми, що постачаються з вихідним кодом, є більш уразливими, оскільки зловмисник може знайти потенційно вразливі заяви до атаки [14].

Моделювання загроз. Атаки SQL-ін’єкцій дозволяють зловмисникам підробити ідентифікаційні дані, підробити існуючі дані, спричинити проблеми відмови, такі як анулювання транзакцій або зміна балансі, та стати адміністраторами сервер бази даних.

SQL-ін’єкції дуже поширений у програмах PHP і ASP через поширеність старих функціональних інтерфейсів.

Жорсткість атак SQL Injection обмежується навичками та уявою зловмисника, а також, меншою мірою, глибокими контрзаходами захисту, такими як підключення з низькими привілеями до сервера баз даних тощо. Загалом, вважайте SQL-ін’єкцію серйозним ударом [15].

Існує багато відомих вразливостей SQL-ін'єкції, серед яких:

• UNION-атаки. Зловмисники використовують операцію UNION для комбінування даних з двох або більше таблиць із метою отримати чутливу інформацію або змінити результат запиту.

• Blind SQL-ін'єкція. Зловмисники використовують логічні умови для виявлення інформації про базу даних, навіть без прямого виведення результатів запиту.

• Error-based SQL-ін'єкція. Зловмисники використовують спеціально створені запити, які викликають помилки бази даних, що містять корисну інформацію про структуру бази даних або дані.

Існує чотири основні категорії атак SQL-ін’єкції на бази даних:

- Маніпуляції SQL
- Введення коду
- Ін'єкція виклику функції
- Переповнення буфера

Маніпуляції SQL. Найбільш поширеним типом атаки з використанням SQL-ін'єкцій є маніпулювання SQL. Зловмисник намагається змінити існуючу інструкцію SQL, додаючи елементи до речення WHERE або розширюючи інструкцію SQL за допомогою операторів SET, таких як UNION, INTERSECT або MINUS. Існують і інші можливі варіанти, але це найбільш значущі приклади [14].

Класична маніпуляція з SQL виконується під час автентифікації при вході в систему. Спрощена веб-програма може перевірити автентифікацію користувача, виконавши наступний запит на рисунку 2.1.1 і перевіривши, чи були повернуті рядки.

```
SELECT * FROM users
WHERE username = 'bob' and PASSWORD = 'mypassword'
```

Рисунок 2.1.1 – Переіврка автентифікації користувача

Зловмисник намагається маніпулювати оператором SQL на рисунку 2.1.2.

```
SELECT * FROM users
WHERE username = 'bob' and PASSWORD = 'mypassword' or 'a' = 'a'
```

Рисунок 2.1.2 – Маніпулювання оператором SQL

Виходячи з пріоритету оператора, пропозиція WHERE є істинною для кожного рядка, і зловмисник отримав доступ до програми.

Введення коду. Атаки введення коду намагаються додати додаткові оператори SQL або команди до існуючого оператора SQL.

Цей тип атаки часто використовується проти програм Microsoft SQL Server, але рідко працює з базою даних Oracle. але рідко працює з базою даних Oracle. Оператор EXECUTE в SQL Server є частою мішенню атак SQL-ін’єкцій – в Oracle немає відповідного оператора [14].

Ін'єкція виклику функції. Ін’єкція виклику функції – це вставка спеціальних функцій у вразливий оператор SQL. Ці виклики функцій можна використовувати для викликів операційної системи або маніпулювання даними в базі даних [14].

Переповнення буфера. Ряд стандартних функцій бази даних чутливі до переповнення буфера, що може бути використано через атаку SQL-ін’єкції в базу даних без виправлень. Відомі переповнення буфера існують у стандартних пакетах бази даних, а також у стандартних функціях бази даних, таких як TZ_OFFSET, TO_TIMESTAMP_TZ, BFILENAME, FROM_TZ, NUMTOYMINTERVAL і NUMTODSINTERVAL.

Атака переповнення буфера з використанням TZ_OFFSET, TO_TIMESTAMP_TZ, BFILENAME, FROM_TZ, NUMTOYMINTERVAL або NUMTODSINTERVAL виконується за допомогою методів впровадження функції, описаних раніше. Використовуючи переповнення буфера за допомогою SQL-ін’єкції, можна отримати віддалений доступ до операційної системи. Широко доступна додаткова інформація про виконання та запобігання атакам переповнення буфера [14].

## 2.2Denial of Service (DoS/DDoS)

Атака відмови в обслуговуванні (DoS) має на меті зробити ресурс (сайт, додаток, сервер) недоступним для тієї мети, для якої він був розроблений.

Існує багато способів зробити послугу недоступною для законних користувачів, серед іншого, маніпулюючи мережевими пакетами, програмуючи, логічно обробляючи вразливості або ресурси, що обробляють вразливості. Якщо служба отримує дуже велику кількість запитів, вона може перестати бути доступною для законних користувачів. Подібним чином служба може зупинитися, якщо буде використана програмна вразливість або те, як служба обробляє ресурси, які вона використовує [3].

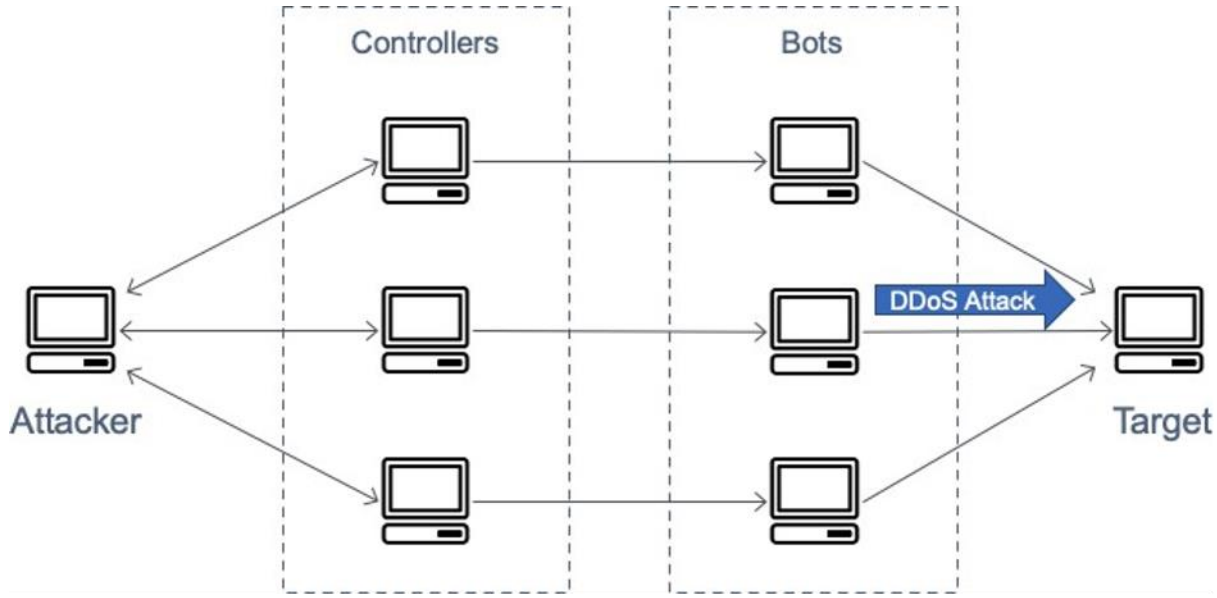


Рисунок 2.2.1 – Приклад схеми DDoS/DoS атаки

Зловмисники постійно змінюють тактику відмови в обслуговуванні (DoS), переміщаючись вгору по стеку від мереж до серверів, а з серверів - на прикладний рівень. В результаті узгоджених зусиль щодо поліпшення захисту мережі та підвищення надійності серверів зловмисники переключають свою увагу на більш легкі цілі: бази даних [2].

Існують різні типи DoS-атак [4]:

- SYN-флуд. У мережах на основі TCP/IP при першому встановленні з'єднання відбувається так зване рукостискання. Під час цього рукостискання відбувається обмін пакетами даних SYN та ACK. При атаці за допомогою SYN-флуд в комп'ютерну систему відправляються так звані SYN-пакети. Замість власної адреси відправника ці пакети містять помилкову IP-адресу, доступ до якої

можливий через Інтернет. Комп'ютерна система, що атакується, намагається відповісти на пакети SYN пакетами SYN-ACK. Однак, оскільки адреса відправника першої посилки була підробленою, система не може зв'язатися з комп'ютером, який хотів встановити з нею з'єднання за цією адресою. Після закінчення певного періоду часу атакована система припинить спроби встановити з'єднання. Якщо велика кількість фальсифікованих пакетів SYN надійде одночасно, комп'ютер, який атакується, використає всі свої можливості підключення, марно намагаючись відправити пакети SYN-ACK, що зробить його повністю недоступним для інших систем.

- Ping-флуд. Ping - це програма, яка перевіряє, чи можна зв'язатися з іншими комп'ютерами в мережі. При переповненні ping зловмисник бомбардує цільовий комп'ютер величезною кількістю так званих пінгів. Комп'ютер змушений задіяти всі свої ресурси, щоб відповісти на ping (за допомогою "pongs"). Залежно від типу та розміру повідомлень, що надходять щосекунди, комп'ютери, що працюють на старих операційних системах, можуть повністю вийти з ладу протягом дуже короткого проміжку часу. У всіх випадках переповнення ping значно знижує продуктивність комп'ютера, що атакується, і, перш за все, Мережі, в якій знаходиться комп'ютер. Атака не тільки призводить до збою системи, але і може виявитися дуже дорогою, якщо плата за мережеве підключення стягується в залежності від обсягу генерованих даних, а не від часу.

Бомбардування поштою. Під час бомбардування поштою зловмисники або надсилають надзвичайно об'ємний електронний лист на цільову адресу, або бомбардують його тисячами повідомлень. Це призводить до засмічення облікового запису електронної пошти. У гіршому випадку сервер електронної пошти сповільнюється або повністю виходить з

ладу. Такого роду поштові атаки можна відносно легко здійснити за допомогою програм, доступних онлайн.

## 2.3Cross Site Scripting (XSS)

Атаки міжсайтових сценаріїв (XSS) - це тип ін’єкцій, при якому шкідливі сценарії вводяться на безпечні та надійні веб-сайти. Атаки XSS відбуваються, коли зловмисник використовує веб-програму для надсилання шкідливого коду, як правило, у вигляді сценарію на стороні браузера, іншому кінцевому користувачеві. Недоліки, які дозволяють цим атакам бути успішними, є досить поширеними і виникають скрізь, де веб-програма використовує введені користувачем дані у вихідних даних, які він генерує, не перевіряючи та не кодуючи їх [8].

Атака міжсайтових сценаріїв (XSS), як правило, не безпосередньо спрямована на бази даних, а скоріше на користувачів додатків або веб-сайтів, які взаємодіють з базами даних.

Міжсайтові сценарії (XSS) атаки відбуваються, коли:

- Дані надходять у веб-програму через ненадійне джерело, найчастіше через веб-запит.
- Дані включаються в динамічний вміст, який надсилається веб-користувачу без перевірки на шкідливий вміст.

Шкідливий вміст, який надсилається веб-переглядачу, часто має форму сегмента JavaScript, але також може включати HTML, Flash або будь-який інший тип коду, який може виконувати браузер. Різноманітність атак на основі XSS майже безмежна, але зазвичай вони включають передачу зловмиснику особистих даних, як-от файлів cookie або іншої інформації про сеанс, перенаправлення жертви на веб-контент, який контролює зловмисник, або виконання інших зловмисних операцій на комп’ютері користувача. під виглядом вразливого сайту.

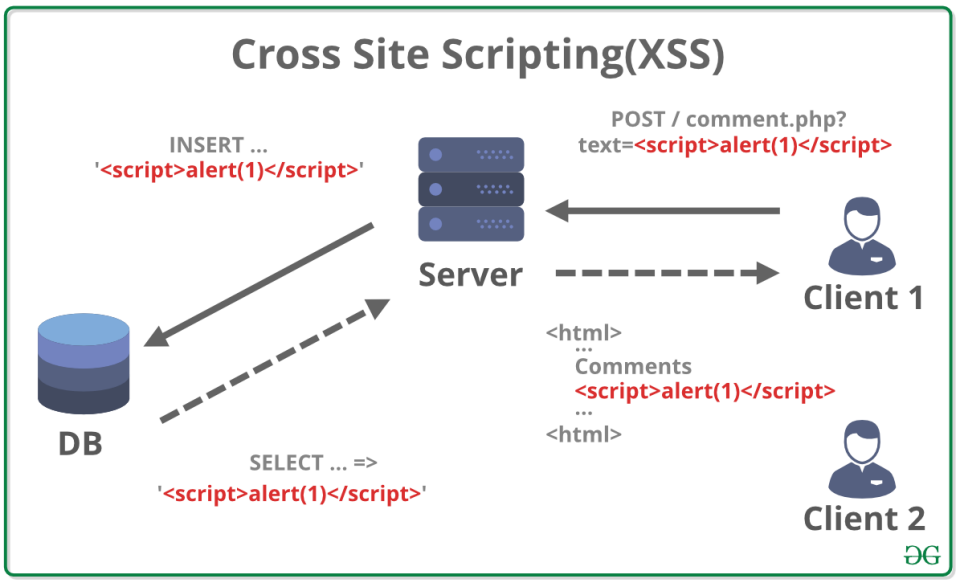


Рисунок 2.3.1 – Схема атаки з використанням міжсайтових сценаріїв

При виконанні XSS-атаки зловмисник може:

- Отримати доступ до конфіденційних даних користувача, таким як логін, пароль, cookie-файли і т. д.
- Маніпулювати веб-сторінкою або додатком для шкідливих цілей, наприклад, перенаправлення на фішингові сайти, виконання небажаних дій від імені користувача і т. д.

Відображені та збережені атаки XSS [8].

Відображені XSS-атаки. Відображені XSS-атаки — це атаки, при яких впроваджений скрипт відображається веб-сервером, наприклад, у повідомленні про помилку, результати пошуку або будь-яку іншу відповідь, яка включає деякі або всі вхідні дані, відправлені на сервер як частину запиту. Відображені XSS-атаки доставляються жертвам за іншим маршрутом, наприклад, у повідомленні електронної пошти або на будь-якому іншому веб-сайті. Коли користувача обманом змушують клацнути шкідливе посилання, відправити спеціально створену форму або навіть просто перейти на шкідливий сайт, впроваджений код переміщається на вразливий веб-сайт, що відображає атаку назад у браузер користувача. Потім браузер виконує код, тому що він прийшов із «довіреного» сервера. Відображений XSS також іноді

називають непостійним або XSS тип I (атака здійснюється через один цикл запиту/відповіді).

Збережені атаки XSS. Збережені атаки – це ті, коли ін’єктований сценарій постійно зберігається на цільових серверах, наприклад у базі даних, у форумі повідомлень, у журналі відвідувачів, у полі коментарів тощо. Потім жертва отримує шкідливий сценарій із сервера, коли він запитує збережений інформації. Збережений XSS також іноді називають постійним XSS або XSS типу II.

## 2.4Слабка автентифікація

Слабка автентифікація [16] — це вразливість, яка може виникнути, коли програма не використовує надійні методи автентифікації або використовує типові або слабкі паролі. Зловмисники можуть скористатися цією вразливістю, щоб отримати неавторизований доступ до програми або бази даних.

Найпростіший спосіб зламати базу даних - це використання ідентифікації авторизованого користувача на цій базі даних.

Ці атаки не обов'язково є складними і можуть бути виконані "скрипт-дітьми", але вони дають хакерам принаймні стільки доступу, скільки має цей конкретний користувач, і, можливо, навіть більше [18].

Процес використання слабкої автентифікації зазвичай включає такі кроки [16]:

- Ідентифікація цілі. Зловмисник визначає цільову програму або базу даних, які вразливі до слабкої автентифікації. Це можна зробити за допомогою розвідки або сканування на відомі вразливості.
- Спроба входу. Зловмисник намагається увійти в програму або базу даних, використовуючи типові або звичайні імена користувачів і паролі. У разі успіху зловмисник отримує доступ до програми або бази даних.
- Атака грубою силою. Якщо типові або загальні паролі не працюють, зловмисник може спробувати атаку грубою силою, яка передбачає

спробу багатьох різних комбінацій імен користувачів і паролів, доки одна не буде успішною.

- Злом паролів. Якщо паролі хешовані або зашифровані, зловмисник може спробувати зламати їх за допомогою інструментів або методів, таких як словникові атаки, райдужні таблиці або атаки грубою силою. Після того, як пароль буде зламано, зловмисник може отримати доступ до програми або бази даних.
- Використовуйте слабкі місця. У деяких випадках зловмисник може виявити слабкі місця в процесі автентифікації, такі як викрадення сеансу, уразливості скидання пароля або атаки соціальної інженерії, щоб обійти автентифікацію та отримати доступ до програми або бази даних. Щоб запобігти слабкій вразливості автентифікації, розробники повинні застосувати політику надійних паролів, наприклад вимогу до складних паролів, термін дії пароля та багатофакторну автентифікацію. Імена користувачів і паролі за замовчуванням слід змінити або вимкнути. Програми слід регулярно перевіряти та перевіряти на вразливості, а журнали доступу слід відстежувати на наявність підозрілої активності.

Недостатня аутентифікація може мати серйозні наслідки, включаючи втрату конфіденційної інформації, порушення даних, крадіжку особистих даних, вплив на бізнес-операції та репутаційний збиток.

## 2.5Атаки шкідливих програм

Атаки шкідливих програм [5] – це тип кібератаки, який передбачає використання шкідливого програмного забезпечення для отримання несанкціонованого доступу, викрадення даних або завдання шкоди системі. Процес атак зловмисного програмного забезпечення зазвичай включає наступні етапи:

- Доставка. Зловмисник доставляє шкідливе програмне забезпечення до цільової системи. Це можна зробити за допомогою різних методів, таких

як вкладення електронної пошти, шкідливі веб-сайти або завантаження зараженого програмного забезпечення.

- Встановлення. Після доставки зловмисного програмного забезпечення його необхідно інсталювати в цільовій системі. Це можна зробити за допомогою різних методів, таких як тактика соціальної інженерії, щоб обманним шляхом змусити користувача запустити зловмисне програмне забезпечення або використати вразливі місця в програмному забезпеченні системи.
- Виконання. Після встановлення зловмисного програмного забезпечення воно виконується в цільовій системі. Тип зловмисного програмного забезпечення визначатиме його поведінку, яка може варіюватися від викрадення даних до завдання шкоди системі.
- Стійкість. Щоб зберегти контроль над системою, зловмисне програмне забезпечення може встановити стійкість, змінюючи налаштування системи або створюючи нові облікові записи користувачів.
- Командування та контроль. Зловмисне програмне забезпечення може з’єднатися з віддаленим сервером командування та керування (C&C), щоб отримати інструкції від зловмисника або надіслати зловмиснику викрадені дані.
- Викрадання даних. Якщо зловмисне програмне забезпечення призначене для викрадення даних, воно може перенести викрадені дані на сервер зловмисника або зберегти їх локально для подальшого отримання.
- Приховати сліди. Щоб замести сліди, зловмисник може спробувати стерти свою діяльність із файлів журналу або змінити дані, до яких він отримав доступ.

Щоб запобігти атакам зловмисного програмного забезпечення, користувачі повинні використовувати антивірусне програмне забезпечення, оновлювати програмне забезпечення за допомогою останніх виправлень

безпеки та бути обережними, відкриваючи вкладення електронної пошти або завантажуючи програмне забезпечення. Розробники повинні дотримуватися методів безпечного кодування та проводити регулярні перевірки безпеки та тестування вразливостей, щоб виявити й усунути потенційні вразливості. Організації повинні впроваджувати заходи безпеки мережі, такі як брандмауери та системи виявлення вторгнень, щоб відстежувати та запобігати атакам шкідливих програм.

## 2.6Проблеми з конфігурацією

Проблеми з конфігурацією [7], [9] є поширеною вразливістю, яка може виникнути, коли системні адміністратори неправильно налаштовують свої системи. Зловмисники можуть використовувати ці вразливості для отримання несанкціонованого доступу до системи, крадіжки даних або пошкодження системи. Процес усунення проблем з конфігурацією зазвичай включає в себе наступні кроки:

- Визначення мети. Зловмисник ідентифікує цільову систему, яка вразлива до проблем з конфігурацією. Це може бути зроблено шляхом розвідки або сканування на наявність відомих вразливостей.
- Виявлення неправильної конфігурації. Зловмисник ідентифікує конкретну неправильну конфігурацію, яка може бути використана для отримання доступу або заподіяння шкоди системі. Це може бути неправильно налаштований брандмауер, слабкий пароль або незахищена Мережева служба.
- Збір інформації. Зловмисник збирає інформацію про цільову систему, щоб визначити версію та конфігурацію програмного забезпечення, що працює в системі. Ця інформація може бути використана для виявлення відомих вразливостей або для розробки користувацьких експлойтів.
- Розробка експлойта. Зловмисник розробляє експлойт, спрямований на конкретну проблему конфігурації, виявлену раніше. Це можна зробити,

змінивши існуючий код експлуатації або розробивши спеціальний код, який використовує вразливість на свою користь.

- Початок атаки. Як тільки експлойт розроблений, зловмисник запускає атаку на цільову систему. Атака може бути пов'язана з отриманням несанкціонованого доступу, крадіжкою даних або заподіянням шкоди системі.
- Замітання слідів. Щоб прикрити сліди, зловмисник може спробувати стерти свої дії з файлів журналів або змінити дані, до яких він отримав доступ.

Щоб запобігти проблемам з конфігурацією, системним адміністраторам слід дотримуватися рекомендацій щодо безпеки, таких як впровадження надійних паролів, використання шифрування та оновлення програмного забезпечення за допомогою останніх виправлень безпеки. Організації також повинні регулярно проводити аудит безпеки та тестування на вразливості для виявлення та усунення потенційних вразливостей. Заходи безпеки мережі, такі як брандмауери та системи виявлення вторгнень, також повинні бути впроваджені для моніторингу та запобігання атакам.

## 2.7Недостатнє шифрування

Недостатнє шифрування є вразливістю, яка може виникнути, коли дані недостатньо захищені шифруванням або використовуються слабкі алгоритми шифрування. Злоумисники можуть використовувати цю вразливість, щоб отримати несанкціонований доступ до конфіденційних даних або змінити дані під час передачі. Процес експлуатації недостатнього шифрування зазвичай включає наступні кроки:

- Визначення цілі. Злоумисник визначає цільову систему, яка має вразливість недостатнього шифрування. Це може бути зроблено за допомогою розвідки або сканування для виявлення відомих вразливостей.

- Виявлення слабкості шифрування. Злоумисник визначає конкретну слабкість шифрування, що використовується на цільовій системі. Це може бути слабі ключі шифрування або алгоритм, що вразливий до атак.
- Перехоплення трафіку. Злоумисник перехоплює трафік між цільовою системою та користувачем або між двома системами. Це може бути зроблено за допомогою різних методів, таких як атаки типу "людина посередині" або перехоплення мережевого трафіку.
- Розшифрування трафіку. Якщо шифрування можна легко розшифрувати, злоумисник розшифровує перехоплений трафік, щоб отримати доступ до конфіденційних даних або змінити дані під час передачі.
- Експлуатація даних. Після розшифрування даних злоумисник може використовувати конфіденційні дані для власних цілей або змінювати дані під час передачі.
- Приховання слідів. Щоб приховати свою діяльність, злоумисник може спробувати видалити свої дії з журналів або змінити дані, до яких він мав доступ.

Щоб запобігти вразливостям недостатнього шифрування, організації повинні впроваджувати сильні алгоритми шифрування та використовувати відповідну довжину ключів для захисту конфіденційних даних. Крім того, організації повинні регулярно переглядати свої практики шифрування, щоб переконатися, що вони відповідають останнім стандартам безпеки. Мережевий трафік також повинен бути контрольований на наявність підозрілої активності, а також слід впроваджувати програми освіти та підвищення обізнаності користувачів для запобігання атакам соціального інженерінгу [10].

# 3 МЕТОДИ ТА ЗАСОБИ ЗАХИСТУ СУЧАНИХ БАЗ ДАНИХ

Безпека баз даних – це комплекс організаційно-технічних заходів і правових норм для попередження заподіяння збитку інтересам власника інформації (в тому числі від незаконного використання та шкідливих загроз та атак) [13].

## 3.1Засоби для зменшення вразливостей баз даних

Засоби для зменшення вразливостей реляційних баз даних. Існує кілька засобів, які можна використовувати для зменшення вразливостей реляційних баз даних [11], [12], [13]:

- Засоби моніторингу діяльності бази даних (Database Activity Monitoring, DAM). Засоби DAM моніторять активність бази даних в реальному часі для виявлення та запобігання несанкціонованому доступу, атакам SQL-ін'єкції та іншим підозрілим діям. Ці засоби також надають можливість аудиту баз даних, звітності про відповідність та управління ризиками.
- Сканери вразливостей. Сканери вразливостей можуть ідентифікувати вразливості в системі баз даних та надавати рекомендації щодо усунення. Ці інструменти також можуть використовуватись для сканування баз даних на наявність відомих вразливостей та проблем конфігурації.
- Засоби шифрування та управління ключами. Засоби шифрування використовуються для шифрування конфіденційних даних, збережених у базі даних. Засоби управління ключами дозволяють безпечно керувати ключами шифрування, забезпечуючи доступ лише авторизованим користувачам.
- Засоби управління оновленнями (Patch Management). Засоби управління оновленнями використовуються для забезпечення актуальності базової

системи бази даних за допомогою останніх патчів безпеки, що зменшує ризик використання відомих вразливостей.

- Засоби контролю доступу. Засоби контролю доступу використовуються для управління привілеями та дозволами користувачів, забезпечуючи доступ лише до необхідних даних для виконання їхніх робочих функцій. Ці інструменти також можуть реалізовувати політики паролів та багатофакторну аутентифікацію.
- Засоби резервного копіювання та відновлення. Засоби резервного копіювання та відновлення використовуються для забезпечення можливості відновлення даних у разі порушення безпеки або втрати даних. Регулярні резервні копії також можуть використовуватись для виявлення та відновлення випадків втручання в дані.

Використовуючи ці засоби, адміністратори баз даних та фахівці з безпеки можуть ефективно управляти безпекою своїх реляційних баз даних та зменшити вразливості. Однак важливо зазначити, що жоден окремий інструмент або стратегія не може забезпечити повну безпеку, і для забезпечення комплексної безпеки необхідно застосовувати шаровий підхід, що включає регулярні оцінки безпеки, навчання користувачів та плани реагування на інциденти.

Ситуації, в яких зазвичай використовуються ці засоби:

- Засоби моніторингу діяльності бази даних (DAM) зазвичай використовуються для моніторингу та захисту баз даних від несанкціонованого доступу, атак SQL-ін'єкції та підозрілої активності. Вони особливо корисні в ситуаціях, коли конфіденційні дані зберігаються в базі даних, таких як фінансові або медичні дані.
- Сканери вразливостей зазвичай використовуються для ідентифікації вразливостей у системі баз даних та надання рекомендацій щодо їх усунення. Вони зазвичай використовуються під час впровадження нової

бази даних, при внесенні змін до системи баз даних або після виявлення нових вразливостей.

- Засоби шифрування та управління ключами використовуються для шифрування конфіденційних даних, збережених у базі даних, таких як номери кредитних карток, соціальні номери та особиста інформація про здоров'я. Засоби управління ключами використовуються для безпечного керування ключами шифрування та забезпечення доступу лише авторизованих користувачів до цих ключів.
- Засоби управління оновленнями використовуються для підтримки актуальності системи бази даних за допомогою останніх патчів безпеки, що зменшує ризик використання відомих вразливостей. Вони зазвичай використовуються в систематичному режимі, щоб забезпечити, що система бази даних завжди оновлюється за останніми стандартами безпеки.
- Засоби контролю доступу використовуються для управління привілеями та дозволами користувачів, забезпечуючи доступ лише до необхідних даних для виконання їх робочих функцій. Вони зазвичай використовуються для забезпечення дотримання політик безпеки та запобігання несанкціонованому доступу до бази даних.
- Засоби резервного копіювання та відновлення використовуються для забезпечення можливості відновлення даних в разі порушення безпеки або втрати даних. Вони зазвичай використовуються систематично для регулярного створення резервних копій даних та перевірки їх відновлення.

Кожен з цих засобів використовується для зменшення певного типу вразливостей реляційних баз даних, а їх використання буде залежати від потреб організації в безпеці та характеру даних, збережених у базі даних.

## 3.2Методи для зменшення вразливостей баз даних

Існують кілька методів, які повинні бути вжиті для зменшення вразливостей баз даних [17]:

- Автентифікація та контроль доступу. Слід використовувати надійні засоби автентифікації, такі як багатофакторна автентифікація, щоб забезпечити доступ до бази даних лише авторизованим користувачам. Контроль доступу повинен бути належним чином налаштований, щоб обмежити доступ користувачів лише до необхідних даних та функцій.
- Шифрування. Чутлива інформація повинна бути зашифрована як у спокої, так і під час передачі. Шифрування слід використовувати для захисту даних, що зберігаються в базі даних, а також для даних, що передаються по мережі.
- Регулярне оновлення. Програмне забезпечення бази даних повинно бути оновлене за допомогою останніх патчів безпеки. Регулярне оновлення допоможе зменшити ризик використання відомих вразливостей і успішних атак.
- Моніторинг активності бази даних. Слід вести моніторинг активності бази даних для виявлення та запобігання несанкціонованому доступу та підозрілій активності. Моніторинг може допомогти виявити та запобігти таким атакам, як SQL-ін'єкція та несанкціонований доступ до даних.
- Резервне копіювання та відновлення. Слід регулярно створювати резервні копії бази даних та зберігати їх в безпечному місці. Резервні копії слід регулярно перевіряти, щоб переконатися, що їх можна відновити в разі порушення безпеки або втрати даних.
- Керування конфігурацією. База даних повинна бути належним чином налаштована з точки зору безпеки, а зміни конфігурації повинні бути ретельно контрольовані та перевірені перед впровадженням. Керування конфігурацією допоможе запобігти помилкам конфігурації, які можуть призвести до вразливостей.

- Навчання користувачів. Користувачі, які мають доступ до бази даних, повинні отримати навчання з найкращих практик забезпечення безпеки, таких як ефективне керування паролями та виявлення та повідомлення про підозрілу активність.

Застосовуючи ці заходи, організації можуть краще захистити свої бази даних від вразливостей та зменшити ризик успішних атак. Проте варто пам'ятати, що жоден окремий інструмент або стратегія не може забезпечити повноцінну безпеку, тому необхідно використовувати комплексний підхід, який включає регулярні оцінки безпеки, навчання користувачів та плани реагування на інциденти. А тепер розглянемо основні методі ближче.

## 3.3Автентифікація та авторизація баз даних

Під час автентифікації паролем очікується, що користувачі запам'ятовують та використовують надійні, довгі та складні паролі, що вони запам'ятовують і використовують міцні, довгі і складні паролі та вводять їх, коли це необхідно. Різні паролі повинні використовуватися для різних баз даних, а обмін паролями бази даних заборонено. Однак як адміністратори, так і звичайні користувачі привертаються зручністю і шляхами скорочення, а хакери готові використовувати таку людську поведінку.

Багато типів баз даних підтримує різні способи автентифікації, включаючи паролі, збережені локально в базі даних або в централізованих каталогових службах. Користувачі також можуть бути автентифіковані операційною системою або різними зовнішніми службами аутентифікації, такими як Kerberos, сертифікати з відкритим ключем та RADIUS.

Паролі використовуються для односторонньої автентифікації користувача в базі даних, тоді як Kerberos та сертифікати з відкритим ключем підтримують взаємну автентифікацію, що гарантує, що користувач дійсно підключається до правильної бази даних. Хоча використання паролів зручне, їх легше скомпрометувати порівняно з обліковими даними Kerberos або PKI.

Після успішної автентифікації користувача йому надається доступ до схеми в базі даних, яка складається з таблиць, представлень, індексів та процедур, а також відповідних повноважень через ролі та привілеї. При аутентифікації користувачів за допомогою каталогової служби, вони отримують власну схему бази даних (ексклюзивний мапінг) або відображаються на спільну схему (спільний мапінг) [18].

Метод автентифікації та авторизації баз даних використовується для захисту від наступних вразливостей:

- SQL-ін'єкції. Автентифікація та авторизація допомагають уникнути SQL-ін'єкцій, оскільки правильно налаштована автентифікація та авторизація не дозволяють виконувати шкідливі SQL-запити без належних привілеїв.
- DoS/DDoS атаки. Методи автентифікації та авторизації не є прямими методами для захисту від DoS/DDoS атак, оскільки їх основна мета полягає у контролі доступу. Однак, вони можуть забезпечити захист, вимагаючи аутентифікацію для виконання запитів і обмежуючи кількість запитів, які можуть бути виконані користувачем за певний період часу.
- XSS (Cross Site Scripting). Автентифікація та авторизація не прямо впливають на захист від XSS-атак. Захист від XSS-атак вимагає використання інших методів, таких як фільтрація та екранування введених даних.
- Слабка автентифікація. Метод автентифікації та авторизації баз даних сам по собі спрямований на усунення слабких місць у процесі ідентифікації та контролю доступу до бази даних. Відповідно, він захищає від слабкої автентифікації.
- Атаки шкідливих програм. Метод автентифікації та авторизації баз даних може допомогти уникнути атак шкідливих програм,

забезпечуючи, що лише користувачі з належними привілеями мають доступ до бази даних.

- Проблема з конфігурацією. Автентифікація та авторизація баз даних не вирішують безпосередньо проблеми з конфігурацією. Проте вони можуть допомогти уникнути використання недокладних налаштувань аутентифікації та авторизації, які можуть стати джерелом вразливостей.
- Недостатнє шифрування. Автентифікація та авторизація не є методами шифрування даних. Вони не вирішують проблему недостатнього шифрування в базі даних. Для захисту від цієї вразливості необхідно використовувати відповідні методи шифрування даних в базі даних.

## 3.4Шифрування

Автентифікація та авторизація бази даних гарантують, що лише уповноважені користувачі мають доступ до даних, створюючи так звані «задні двері». Однак, якщо зловмисники не можуть отримати доступ шляхом звичайних методів, вони можуть намагатися обійти контроль доступу до бази даних і звернутися до даних іншими способами.

Один з таких способів - перехоплення даних під час їх передачі по мережі, наприклад, між клієнтом і сервером бази даних. Багато внутрішніх мережевих підключень не зашифровані, що дає зловмисникам змогу легко перехопити мережевий трафік і отримати доступ до передаваної інформації.

Інший спосіб для зловмисників - отримання привілейованого доступу до операційної системи і пряме читання файлів бази даних, обходячи контроль доступу до бази даних. Зловмисники також можуть нападати на резервні копії бази даних, які можуть бути збережені на фізичних носіях і доставлені до віддалених місць.

Шифрування є найкращою технікою захисту даних у таких ситуаціях, оскільки воно робить дані нерозбірливими для тих, хто намагається отримати до них прямий доступ. За допомогою шифрування проблема захисту великої

кількості даних зводиться до набагато простішої проблеми захисту ключа шифрування [18].

Типова криптосистема включає:

- Ключ шифрування для зашифрування даних (відкритий текст).
- Алгоритм шифрування, який з використанням ключа шифрування перетворює відкритий тексту шифротекст.
- Ключ розшифрування для розшифрування шифротексту.
- Алгоритм розшифрування для використання ключа розшифрування з шифротекстом та створення початкового відкритого тексту.

Метод шифрування баз даних використовується для захисту від наступних вразливостей:

- SQL-ін'єкції. Шифрування баз даних не є прямим методом для захисту від SQL-ін'єкцій. Хоча шифрування може ускладнити атаку шляхом шифрування введених даних, це не повністю розв'язує проблему SQL-ін'єкцій. Для захисту від SQL-ін'єкцій також потрібно використовувати інші методи, такі як параметризовані запити та коректна обробка введених даних.
- Атаки шкідливих програм. Шифрування баз даних може використовуватись для захисту від певних атак шкідливих програм, але не є універсальним засобом захисту. Шифрування може ускладнити розшифрування та використання даних зловмисниками, але не забезпечує повністю безпеку від усіх видів шкідливих програм. Для захисту від шкідливих програм також потрібно використовувати антивірусне програмне забезпечення, механізми виявлення вторгнень та інші методи захисту.
- Проблеми з конфігурацією. Шифрування баз даних не є прямим методом для вирішення проблем з конфігурацією. Проте воно може бути використано для захисту від несанкціонованого доступу до бази даних у випадку, коли конфігураційні параметри неправильно встановлені. Для

вирішення проблем з конфігурацією бази даних потрібно використовувати правильні методи налаштування і моніторингу.

- Недостатнє шифрування. Метод шифрування баз даних використовується саме для вирішення проблеми недостатнього шифрування. Він дозволяє зашифровувати дані в базі даних, що забезпечує додатковий рівень захисту від несанкціонованого доступу до чутливої інформації. Шифрування баз даних може бути ефективним для запобігання доступу до даних в разі фізичного зламу системи або витоку інформації.

## 3.5Аудит баз даних та моніторинг

У сучасних організаціях існує велика кількість баз даних, що вимагають аудиту та моніторингу дій користувачів і адміністраторів для відповідності вимогам правил та виявлення порушень безпеки. Цей контроль потребує постійного збору та аналізу великого обсягу даних щодо активності, щоб створювати звіти та генерувати сповіщення про незвичайні активності.

Аудит баз даних та моніторинг мережевої активності означає збір вихідних даних аудиту баз даних та SQL-трафіку на мережевому рівні для моніторингу та створення звітів щодо активності баз даних. Аудит баз даних забезпечує реєстрацію дій користувачів та додатків у базі даних, включаючи тих, хто має високі адміністративні привілеї. Брандмауери баз даних моніторять та оцінюють вхідний SQL-трафік на мережевому рівні, виявляють незвичайні дії або операції, які не відповідають політиці безпеки, та блокують SQL-запити, що не відповідають політиці безпеки, перед досягненням бази даних.

Існують три загальні випадки використання аудиту баз даних та моніторингу мережевої активності: впровадження корпоративних вимог щодо безпеки, забезпечення відповідності до регулятивних вимог та проведення форензичного аналізу.

Хоча корпоративні вимоги щодо безпеки можуть варіюватися, вони передбачають аудит активності привілейованих користувачів, події входу до системи, доступ до чутливих даних, моніторинг мережевого трафіку баз даних, запобігання спробам SQL-ін'єкції та інші методи безпеки. Для цього потрібне рішення, яке підтримує аудит баз даних та моніторинг мережі.

Для організацій, які працюють з чутливими даними, можуть бути застосовані регуляторні вимоги, такі як GDPR, CCPA, PCI, HIPAA, IRS 1075, SOX та UK DPA. Ці вимоги вимагають відстеження доступу до чутливих або особистих даних, включаючи доступ до даних у базі даних. Для виконання цих вимог потрібне рішення, яке надає широкий набір готових звітів щодо відповідності цим регулятивним вимогам.

Організаціям потрібна підтримка криміналістичного аналізу у разі порушення безпеки або при спостереженні підозрілих дій [18]. Криміналістичний аналіз передбачає здатність не лише збирати та зберігати великі обсяги записів аудиту та мережевих подій, але й ефективно просуватися в них. Крім того, він повинен підтримувати зберігання та відновлення історичних даних, відстеження змін привілеїв користувачів та змін збережених процедур для полегшення аналізу [18].

Метод аудиту баз даних та моніторингу баз даних використовується для захисту від наступних вразливостей:

- SQL-ін'єкції. Аудит баз даних та моніторинг баз даних можуть допомогти виявляти атаки SQL-ін'єкцій шляхом моніторингу та аналізу запитів до бази даних. Вони дозволяють виявляти незвичну або підозрілу активність, яка може бути пов'язана з SQL-ін'єкціями.
- DoS/DDoS атаки. Аудит баз даних та моніторинг баз даних можуть допомогти виявляти атаки DoS/DDoS шляхом моніторингу трафіку до бази даних. Вони дозволяють виявляти надмірний або незвичайний обсяг запитів до бази даних, що може свідчити про потенційну атаку.

- XSS (міжсайтовий скриптинг). Аудит баз даних та моніторинг баз даних не є безпосередніми засобами захисту від XSS-атак, оскільки ці атаки зазвичай відбуваються на рівні веб-додатків. Однак, шляхом аудиту та моніторингу можна виявляти дії, що вказують на можливість XSS-атаки, наприклад, спроби вставити шкідливий код в базу даних або незвичайний вивід даних.
- Слабка автентифікація. Аудит баз даних та моніторинг баз даних можуть допомогти виявляти проблеми з автентифікацією, шляхом моніторингу вхідних подій та активності користувачів. Вони дозволяють виявляти незвичну або підозрілу активність, що може свідчити про спроби несанкціонованого доступу до бази даних.
- Атаки шкідливих програм. Аудит баз даних та моніторинг баз даних можуть допомогти виявляти атаки шкідливих програм шляхом моніторингу активності користувачів та запитів до бази даних. Вони дозволяють виявляти незвичайну або підозрілу активність, яка може бути пов'язана з діяльністю шкідливих програм.

Щодо проблеми з конфігурацією та недостатнього шифрування, аудит баз даних та моніторинг баз даних самі по собі не є прямими методами для вирішення цих проблем. Однак, вони можуть сприяти виявленню можливих проблем з конфігурацією або недостатнього шифрування, якщо вони виявляють незвичайну або підозрілу активність, що може вказувати на такі проблеми.

### 3.6Резервне копіювання та відновлення

Резервне копіювання та відновлення баз даних є важливою процедурою для забезпечення безпеки та захисту даних. Основні пункти, які варто враховувати при резервному копіюванні та відновленні, включають [19]:

- Захист даних від випадкової втрати. Резервне копіювання даних дозволяє зберегти цінні і важливі дані в разі випадкової втрати,

системних відмов або катастрофи. Відновлення даних з резервних копій допомагає відновити базу даних до попереднього стану.

- Час відновлення. Важливо враховувати час, необхідний для відновлення бази даних. Швидке відновлення є критичним, особливо у випадку непередбачених ситуацій або аварій.
- Регулярність. Резервне копіювання повинно здійснюватися на регулярній основі, щоб забезпечити актуальність даних та зменшити втрату інформації. Частота резервного копіювання може варіюватися залежно від важливості даних та часу, необхідного для створення копій.
- Зберігання резервних копій. Резервні копії баз даних повинні зберігатися в безпечному місці, віддаленому від основного сервера. Це забезпечує захист даних навіть у разі фізичного знищення або випадкового доступу до сервера.
- Тестування відновлення. Важливо періодично перевіряти процес відновлення з резервних копій, щоб переконатися, що дані можуть бути успішно відновлені і база даних повністю функціональна.
- Безпека. При резервному копіюванні та відновленні баз даних слід враховувати аспекти безпеки, такі як зашифрування резервних копій та контроль доступу до них, щоб забезпечити конфіденційність і цілісність даних.

Метод резервного копіювання та відновлення баз даних не використовується безпосередньо для усунення конкретних вразливостей, таких як SQL-ін'єкція, DoS/DDoS, XSS, слабка автентифікація, атаки шкідливих програм, проблема з конфігурацією чи недостатнє шифрування.

Метод резервного копіювання та відновлення баз даних є процедурою, спрямованою на забезпечення збереження та відновлення даних в разі втрати чи пошкодження. Цей метод допомагає відновити базу даних до попереднього стану, заснованого на резервних копіях, які були створені у попередній момент часу. Він дозволяє відновити базу даних, якщо вона стала недоступною або

пошкодженою через технічні або людські помилки, віруси, збої обладнання та інші подібні ситуації. Хоча резервне копіювання та відновлення баз даних може допомогти відновити дані після певних вразливостей, воно не є прямим методом безпеки для усунення таких вразливостей. Щоб захистити базу даних від SQL-ін'єкцій, DoS/DDoS, XSS, слабкої автентифікації, атак шкідливих програм, проблем з конфігурацією та недостатнього шифрування, слід використовувати відповідні методи безпеки, такі як регулярне оновлення програмного забезпечення, використання безпечних кодувань, застосування сильних методів автентифікації та авторизації, належна конфігурація системи, застосування шифрування даних та інші практики безпеки.

Отже, резервне копіювання та відновлення баз даних не призначено для вирішення конкретних вразливостей безпосередньо, але може слугувати важливою складовою стратегії безпеки та відновлення даних.

## 3.7Порівняння ефективності різних методів та засобів безпек

Нижче представлена таблиця 3.1, яка показує чи ефективні різні методи захисту для пом'якшення найпоширеніших загроз баз даних чи ні.

Таблиця 3.1 – Порівняння ефективності різних методів безпек

| Метод безпеки | SQL-ін'єкції | DoS/DDoS | XSS | Слабка автентифікація | Атаки шкідливих програм | Проблеми з конфігурацією | Недостатнє шифрування |
|---|---|---|---|---|---|---|---|
| Автентифікація та контроль доступу | + | + | + | + | + | + | - |
| Шифрування | + | + | + | - | + | - | + |
| Регулярне оновлення | + | - | + | - | + | - | + |
| Моніторинг активності бази даних | + | - | - | - | - | - | - |
| Резервне копіювання та відновлення | + | + | + | + | + | - | - |
| Керування конфігурацією | - | - | - | + | - | - | - |
| Навчання користувачів | - | - | - | + | - | - | - |

У цій таблиці "+" позначає, що метод безпеки ефективний у боротьбі з певною загрозою, а "-" означає, що метод не є ефективним для даної загрози.

Нижче представлена таблиця 3.2, яка показує чи ефективні різні засобі захисту для пом'якшення найпоширеніших загроз баз даних чи ні.

Таблиця 3.2 – Порівняння ефективності різних засобів безпек

| Засіб безпеки | SQL-ін'єкції | DoS/DDoS | XSS | Слабка автентифікація | Атаки шкідливих програм | Проблеми з конфігурацією | Недостатнє шифрування |
|---|---|---|---|---|---|---|---|
| DAM (Database Activity Monitoring | + | + | + | + | + | - | - |
| Сканер вразливостей | + | - | + | - | + | - | + |
| Інструменти шифрування та управління ключами | + | + | + | - | - | - | + |
| Інструмент управління оновленням | + | - | + | - | - | + | - |
| Інструменти контролю доступу | + | - | + | + | + | + | - |
| Інструменти резервного копіювання та відновлення | + | + | + | + | + | - | - |

У цій таблиці "+" позначає, що засіб безпеки ефективний у боротьбі з певною загрозою, а "-" означає, що засіб не є ефективним для даної загрози.

## 3.8Вироблення рекомендацій щодо використання наявних методів та засобів захисту

Грунтуючись на порівнянні ефективності різних засобів безпеки, можна надати наступні рекомендації по використанню доступних засобів і методів для забезпечення безпеки баз даних залежно від умов їх застосування:

Автентифікація та контроль доступу:

- Рекомендовано використовувати автентифікацію з багатофакторним підходом для захисту від SQL-ін'єкцій та слабкої автентифікації.
- Використовуйте інструменти контролю доступу для забезпечення обмеженого доступу до бази даних та виявлення неправомірних дій.

Шифрування:

- Рекомендовано використовувати інструменти шифрування та управління ключами для захисту від недостатнього шифрування та несанкціонованого доступу.
- Застосовуйте шифрування для захисту конфіденційної інформації, яка передається між базою даних та додатками.

Регулярне оновлення:

- Рекомендується встановлювати оновлення бази даних, системного програмного забезпечення та інструментів для захисту від вразливостей та атак шкідливих програм.
- Використовуйте інструменти оновлення та сканери вразливостей для виявлення та усунення потенційних загроз.

Моніторинг активності бази даних:

- Рекомендується встановити систему моніторингу активності бази даних для виявлення некоректних дій, спроб несанкціонованого доступу та аномальної активності.
- Використовуйте інструменти моніторингу для реагування на вразливості та потенційні атаки.

Резервне копіювання та відновлення:

- Рекомендується регулярно створювати резервні копії бази даних та забезпечувати можливість відновлення даних в разі втрати або пошкодження.
- Використовуйте інструменти резервного копіювання та відновлення для автоматизації процесу створення та відновлення резервних копій.

Керування конфігурацією:

- Рекомендується забезпечити належне керування конфігурацією бази даних для запобігання проблемам з конфігурацією та неправильним налаштуванням.
- Використовуйте інструменти керування конфігурацією для контролю версій, виявлення змін та забезпечення відновлення налаштувань.

Навчання користувачів:

- Рекомендується навчати користувачів про найкращі практики з безпеки бази даних та свідоме використання доступу до неї.
- Організуйте навчальні програми та своєчасно оновлюйте користувальницьку документацію з питань безпеки.

Зверніть увагу, що рекомендації можуть змінюватися в залежності від конкретних вимог та ситуацій, тому рекомендується проводити аналіз загроз та вибирати відповідні заходи забезпечення безпеки.

# 4 ОСОБЛИВОСТІ ЗАХИСТУ БАЗ ДАНИХ У ХМАРАХ

Хмарні бази даних можуть бути вразливими перед різними векторами загроз, які відрізняються від тих, що існують у on-premises (база даних розташована на місці (власному) у користувача, організації або підприємства, а не в хмарному середовищі). Наприклад, зловмисники можуть намагатися отримати доступ до бази даних шляхом прямого нападу на мережу хмарного провайдера або через мережі інших клієнтів. Також ви повинні довіряти адміністраторам хмари та залежати від їхніх політик безпеки [18].

## 4.1Унікальні загрози для баз даних у хмарах

Унікальні загрози для баз даних у хмарах включають [18]:

– Злам доступу до облікових записів. Зловмисники можуть намагатися зламати облікові записи користувачів, які мають доступ до хмарної бази даних. Це може статися через атаки на слабкі паролі, використання перехоплення сеансів або вразливостей автентифікації.

– Вразливості мережевої інфраструктури. Хмарні бази даних можуть бути піддаються атакам на мережеву інфраструктуру хмарного середовища, такі як атаки на віртуальні мережі, мережеві протоколи або недостатньо захищені мережеві комунікації.

– Несанкціонований доступ до фізичних ресурсів. Хмарні середовища мають фізичну інфраструктуру, яка підтримує роботу баз даних. Зловмисники можуть намагатися отримати несанкціонований доступ до фізичних ресурсів, таких як сервери, зберігання даних або мережеві пристрої, щоб здійснити атаку на базу даних.

– Ризик втрати контролю над даними. У хмарних середовищах, де ресурси можуть бути спільними для кількох клієнтів, існує ризик втрати контролю над даними. Це може включати неправильну ізоляцію даних між клієнтами,

помилкове надання дозволів на доступ до даних або можливість зловмисника отримати доступ до чужих даних через слабкість в системі розподілу ресурсів.

– Недостатня захищеність даних під час передачі. Під час передачі даних між хмарною базою даних та користувачами можуть виникати загрози безпеки. Це включає можливість перехоплення чутливих даних під час передачі через мережу або використання незахищених механізмів комунікації.

– Ризик збитку або втрати даних. У випадку виникнення проблеми або випадкової помилки в хмарній інфраструктурі, може виникнути ризик збитку або втрати даних. Це може бути спричинено неправильними резервними копіями, відмовою обладнання або помилками в процесах обслуговування.

## 4.2Безпека в хмарі

Хмарна інфраструктура зазвичай має переваги перед локальними розгортаннями. Деякі з цих переваг включають [18]:

Професійні адміністратори безпеки. Хмарні провайдери надають спеціалізованих адміністраторів безпеки, які володіють знаннями та досвідом у сфері захисту даних. Вони використовують найкращі практики та технології для виявлення та запобігання загрозам.

Прив'язка до угод рівня обслуговування. У хмарних послугах зазвичай встановлюються угоди рівня обслуговування (SLA), які гарантують певний рівень безпеки та доступності. Це включає моніторинг, реагування на інциденти та резервне копіювання даних.

Фокус на процеси. Хмарні провайдери акцентують увагу на розробці та впровадженні надійних процесів безпеки. Це включає використання автоматизованих інструментів для виявлення загроз, моніторингу активності та вдосконалення систем безпеки.

Ізоляція ресурсів. Хмарні провайдери забезпечують ізоляцію ресурсів між різними клієнтами. Це допомагає запобігти несанкціонованому доступу до даних і зменшити ризик витоку інформації.

Захист даних. У хмарних базах даних зазвичай застосовуються різноманітні заходи захисту даних, такі як шифрування даних у спокої та під час передачі, контроль доступу та моніторинг активності бази даних.

Загалом, в хмарних розгортаннях безпека баз даних зазвичай є кращою за розгортаннями на місці. Однак, варто пам'ятати, що безпека даних є спільною відповідальністю між клієнтом і хмарним провайдером, і обидві сторони повинні приділяти належну увагу заходам безпеки для забезпечення захисту даних.

## 4.3Спільна відповідальність за безпеку

Безпека баз даних у хмарному середовищі є спільною відповідальністю між клієнтом і хмарним провайдером. Кожна сторона має свою роль у забезпеченні безпеки даних. Основні аспекти загальної відповідальності за безпеку включають [18]:

Клієнтська відповідальність. Клієнт повинен відповідно налаштовувати та керувати своїми базами даних у хмарному середовищі. Це включає встановлення правильних налаштувань безпеки, управління привілегіями користувачів, шифрування даних та встановлення контролю доступу.

Заходи безпеки хмарного провайдера. Хмарні провайдери повинні забезпечувати безпеку інфраструктури та послуг, які вони надають. Це включає використання надійних мережевих захисних механізмів, контроль доступу до фізичної і віртуальної інфраструктури, моніторинг безпеки та реагування на інциденти.

Спільні заходи безпеки. Клієнти і хмарні провайдери повинні спільно працювати над захистом даних. Це включає обмін інформацією про загрози та вразливості, спільне визначення та впровадження безпечних практик, а також спільне навчання та надання підтримки з питань безпеки.

Загальна відповідальність за безпеку передбачає активну співпрацю між клієнтом і хмарним провайдером для забезпечення надійного захисту даних у хмарному середовищі.

Нижче на рисунку 4.3.1 приклад моделі спільної відповідальності PaaS.

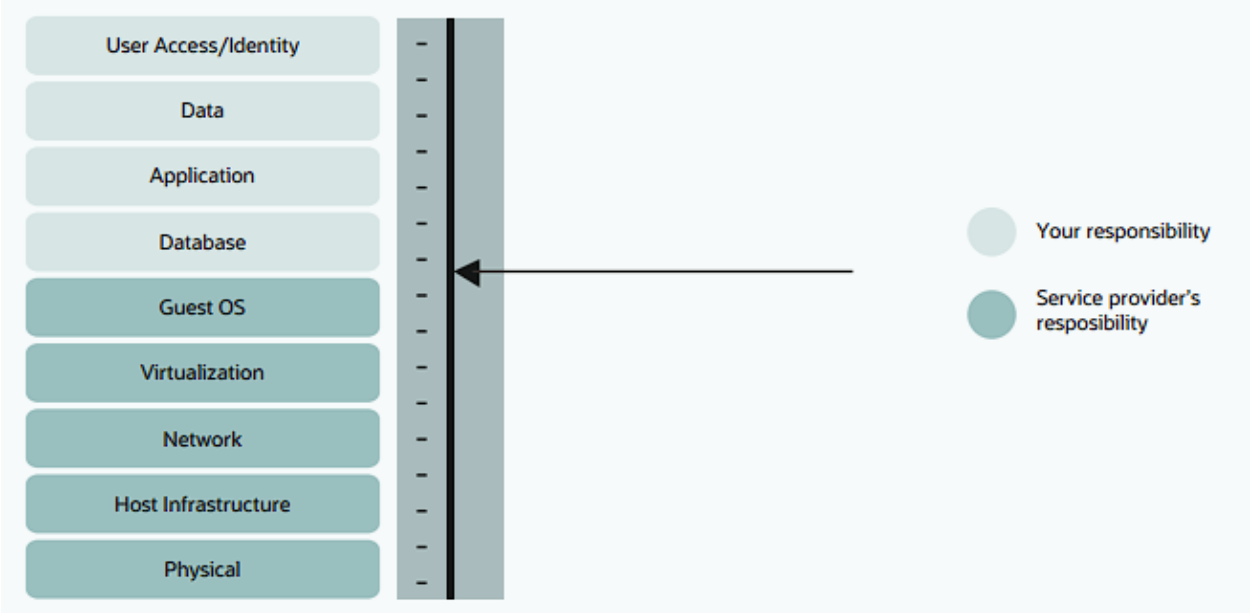


Рисунок 4.3.1 - Модель спільної відповідальності для PaaS

## 4.4Рекомендації щодо захисту хмарних баз даних

Використовувані хмарні сервіси баз даних визначають, коли і як проводити періодичну оцінку безпеки в рамках зобов'язань організації щодо забезпечення безпеки. Необхідно включити перевірку, щоб підтвердити, що постачальник хмарних послуг також виконує свою частину роботи. Оцінки можуть бути процедурними, такими як запит звітів про перевірку третьою стороною (наприклад, SOC2, FedRAMP, EU GDPR або інші перевірки безпеки), або тактичними, такими як періодична перевірка відкритих портів, рівнів виправлень тощо. Тепер розглянемо завдання, які не відносяться до хмарного провайдера. Можливості та методології, описані раніше в цій книзі, продовжують забезпечувати захист бази даних у хмарі. Ось кілька рекомендацій загального характеру та те, як ви можете отримати допомогу в цьому [18]:

- Шифрування. Не розміщуйте незашифровані дані в хмарі. Це ризик, якому ваша організація не повинна піддаватися. Можна автоматично шифрувати хмарні бази даних, що працюють в хмарній інфраструктурі, щоб забезпечити безпеку.
- Мінімізація даних. Не залишайте конфіденційні дані в системах тестування та розробки. Визнайте конфіденційність даних у вашій базі даних і знеособлюйте ці дані за допомогою методів маскування або використовуйте повністю штучні набори даних, які не становлять загрози безпеці. Легко можна знеособити дані в системах тестування і розробки (включаючи заміну ваших даних штучними наборами даних) з використанням відповідних інструментів.
- Контроль доступу. Реалізуйте розподіл обов'язків. Плануйте так, ніби користувачі були скомпрометовані, і зловмисник отримає доступ до середовища, де зберігаються ваші дані. Якщо немає областей безпеки для захисту конфіденційних даних, ніщо не завадить зловмиснику вкрасти їх. Оцініть рівень ризику користувачів вашої бази даних за допомогою функції оцінки користувачів.
- Моніторинг діяльності. Створюйте записи аудиту для всіх важливих дій та відстежуйте журнал аудиту. Переконайтеся, що засоби аудиту організації або хмарного провайдера створюють повідомлення про незвичайні дії. Можна переглядати і включати рекомендовані політики аудиту у вашій базі даних за допомогою відповідних інструментів. Потім можна збирати аудиторські записи та надавати заздалегідь визначені звіти, які допоможуть вам перевірити наявність підозрілої діяльності.

# ВИСНОВОК

Аналіз методів та засобів захисту сучасних баз даних є важливим завданням у світі, де цифрові дані стають все більш цінними та вразливими перед загрозами. У роботі проведено детальний аналіз основ захисту баз даних, вразливостей, методів та засобів захисту, а також особливостей захисту баз даних у хмарних середовищах.

Перш за все, висвітлено основи захисту баз даних, включаючи їхню роль у сучасному світі, системи керування базами даних, а також роль криптографії в їхньому захисті. Розглянуті рівні захисту баз даних, від фізичного рівня до рівня дозволів та доступу, демонструють необхідність комплексного підходу до безпеки.

Проаналізовані вразливості сучасних баз даних, зокрема SQL-ін'єкції, атаки Denial of Service (DoS/DDoS), Cross Site Scripting (XSS), слабка автентифікація, атаки шкідливих програм, проблеми з конфігурацією та недостатнє шифрування. Ці вразливості підкреслюють необхідність ефективного захисту баз даних від різноманітних атак.

Для забезпечення безпеки баз даних були розглянуті різні методи та засоби. Зокрема, було розглянуто засоби для зменшення вразливостей баз даних, які включають в себе використання параметризованих запитів, мінімізацію повноважень та обмеження прав доступу. Також були проаналізовані методи для зменшення вразливостей, такі як моніторинг та виявлення вторгнень, застосування патчів та вдосконалення процесу конфігурації.

Автентифікація та авторизація баз даних є важливими аспектами захисту, і їхні методи були розглянуті в цьому аналізі. Шифрування, аудит баз даних та моніторинг, резервне копіювання та відновлення також є важливими засобами для забезпечення безпеки даних.

Досліджено особливості захисту баз даних у хмарних середовищах, включаючи унікальні загрози, переваги безпеки у хмарі, спільну відповідальність за безпеку та рекомендації щодо захисту хмарних баз даних. З урахуванням поширення хмарних рішень, важливо розуміти специфіку захисту даних у цьому контексті та використовувати відповідні методи та засоби.

Аналіз методів та засобів захисту сучасних баз даних вказує на необхідність постійного вдосконалення практик безпеки та використання найкращих практик для захисту цінних даних. Забезпечення безпеки баз даних є важливою складовою інформаційної безпеки і потребує постійної уваги та підтримки для забезпечення конфіденційності, цілісності та доступності даних.

У цьому аналізі було враховано різноманітні аспекти захисту баз даних, проте важливо зазначити, що безпека є динамічним процесом, і нові загрози постійно з'являються. Тому дослідження та вдосконалення методів та засобів захисту баз даних має бути постійним процесом, що адаптується до змінних умов і загроз.

Завершуючи, враховуючи значення даних у сучасному світі, безпека баз даних є критично важливою. Із зростанням кількості та складності загроз, розуміння методів та засобів захисту баз даних є необхідним для підтримки безпеки і забезпечення довіри до систем обробки даних. Застосування рекомендацій щодо використання наявних методів та засобів захисту є важливим кроком у забезпеченні безпеки баз даних та запобіганні можливим порушенням безпеки в майбутньому.

# ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ


1. Криптологія. Теорія. Практика. Застосування. URL: https://dut.edu.ua/uploads/l_1886_59996057.pdf (дата звернення: 07.05.2023).
2. Dealing with Database Denial of Service. URL: https://cdn.securosis.com/assets/library/reports/Database_DoS.pdf (дата звернення: 31.05.2023).
3. Denial of Service. URL: https://owasp.org/www-community/attacks/Denial_of_Service (дата звернення: 06.05.2023).
4. Denial-of-Service (DoS) and Distributed Denial-of-Service (DDoS). URL: https://www.bsi.bund.de/EN/Themen/Verbraucherinnen-und-Verbraucher/Cyber-Sicherheitslage/Methoden-der-Cyber-Kriminalitaet/DoS-Denial-of-Service/dos-denial-of-service_node.html (дата звернення: 31.05.2023).
5. Qamar A., Karim A., & Chang, V. Mobile malware attacks: Review, taxonomy & future directions. Future Generation Computer Systems, 2019. 887-909 с.
6. Стасишин В. М., Стасишина Т. Л. Бази даних: технології доступу. URL: https://urait.ru/viewer/bazy-dannyh-tehnologii-dostupa-516927#page/1 (дата звернення: 20.05.2023).
7. Abdel rahman A. M., Rodrigues J. J., Mahmoud M. M., Saleem, K., Das A. K., Korotaev V., & Kozlov S. A. Software-defined networking security for private data center networks and clouds: Vulnerabilities, attacks, countermeasures, and solutions. International Journal of Communication Systems, 2021.
8. Cross Site Scripting. URL: https://owasp.org/www-community/attacks/xss/ (дата звернення: 31.05.2023).
9. Malik M., & Patel, T. Database security-attacks and control methods. International Journal of Information, 2016. 175-183 с.

10. Carroll M., Van Der Merwe A., & Kotze P. Secure cloud computing: Benefits, risks and controls. In 2011 Information Security for South Africa, 2011. (pp. 1-9). IEEE.
11. Aliero M. S., & Ghani I. A component based SQL injection vulnerability detection tool. In 2015 9th Malaysian software engineering conference (MySEC), 2015. (pp. 224-229). IEEE. 11.
12. Yaseen Q. Mitigating insider threat in relational database systems. University of Arkansas, 2012. 13 с.
13. Kritikos K., Magoutis K., Papoutsakis M., & Ioannidis, S. A survey on vulnerability assessment tools and databases for cloud-based web applications, 2019. 25 с.
14. Stephen Kost. An Introduction to SQL Injection Attacks for Oracle Developers. URL: https://www.integrigy.com/files/Integrigy_Intro_Oracle_SQL_Injection_Attacks.pdf (дата звернення: 31.05.2023).
15. SQL Injection. URL: https://owasp.org/www-community/attacks/SQL_Injection (дата звернення: 31.05.2023).
16. Arkko J., & Nikader P. Weak authentication: How to authenticate unknown principals without trusted parties, 2004. 5-16 с.
17. Lawai B, Adesoji A, Adekunle, S. Contemporary Control Measures for Mitigating Threats and Vulnerabilities to organizational Databases, 2022.
18. Securing the Oracle Database. URL: https://download.oracle.com/database/oracle-database-security-primer.pdf (дата звернення: 31.05.2023).
19. Backup and Restore of SQL Server Database – Microsoft. URL: https://download.microsoft.com/download/0/F/B/0FBFAA46-2BFD-478F-8E56-7BF3C672DF9D/Backup%20and%20Restore%20of%20SQL%20Server%20Databases.pdf (дата звернення: 01.06.2023).