# ПРИВАТНІСТЬ ТА ЗАХИСТ ПЕРСОНАЛЬНИХ ДАНИХ/ PRIVACY & PROTECTION FROM IDENTITY THEFT

**DOI: 10.18372/2225-5036.30.18618**

# СТРАТЕГІЇ ТА ІННОВАЦІЙНІ ПІДХОДИ ДО ЗАХИСТУ БАЗ ДАНИХ В ЕПОХУ ЗРОСТАЮЧИХ КІБЕРЗАГРОЗ

**Олег Гарасимчук, Оксана Бужович**

*Національний університет «Львівська політехніка»*

**ГАРАСИМЧУК Олег Ігорович,** к.т.н., доц.
*Рік та місце народження:* 1979 рік, м. Бережани, Тернопільська обл., Україна.
*Освіта:* Національний університет «Львівська політехніка», 2001 рік.
*Посада:* доцент кафедри захисту інформації з 2008 року.
*Наукові інтереси:* комплексні системи санкціонованого доступу, генерування псевдовипадкових чисел та послідовностей, генерування пуассонівських імпульсних послідовностей, методи і засоби захисту інформації, проектування комплексних систем захисту інформації, бази даних та знань, сигнальні процесори в системах захисту інформації.
*Публікації:* більше 100 наукових публікацій, серед яких наукові статті, монографії, навчальний посібник, патенти, тези та матеріали доповідей на конференціях.
*E-mail:* oleh.harasymchuk@gmail.com.
*Orcid ID:* 0000-0002-8742-8872.

**БУЖОВИЧ Оксана Сергіївна,** студентка
*Рік та місце народження:* 2003 рік, м. Львів, Україна.
*Освіта:* Національний університет «Львівська політехніка», 2024 рік.
*Посада:* студентка кафедри захисту інформації Національного університету «Львівська політехніка».
*Наукові інтереси:* безпека даних, захист інформації.
*Публікації*: наукові публікації, серед яких наукові статті.
*E-mail:* oksanabuzhovuch4@gmail.com.
*Orcid ID:* 0009-0006-6363-2816.

***Анотація.*** *У сучасному цифровому середовищі, де бази даних відіграють критичну роль у зберіганні та обробці важливої інформації для різних сфер діяльності людства, захист від кіберзагроз стає надзвичайно актуальною задачею. Це відповідно ставить перед організаціями нові вимоги та відповідальність. Сучасні технології не тільки полегшують доступ до даних, але й ставлять під загрозу їх конфіденційність та цілісність. Швидкі та постійно зростаючі виклики кібербезпеки вимагають розробки ефективних стратегій та інноваційних підходів до захисту баз даних, які забезпечують надійність та стійкість баз даних в умовах постійно зростаючих кібератак та порушень безпеки. В роботі детально розглянуті питання організації ефективного захисту інформації, яка зберігається в базах даних. Проаналізовано основні методи захисту інформації в базах даних, зокрема шифрування даних, механізми аутентифікації, контролю доступу та моніторингу активності користувачів. Визначені їх переваги та недоліки, а також можливі наслідки для даних у разі недотримання цих методів захисту. Стаття висвітлює важливість постійного моніторингу та аналізу активності користувачів для вчасного виявлення та реагування на можливі загрози безпеці даних у системі баз даних. Підкреслюється важливість комплексного підходу до захисту, який враховує особливості конкретної організації та дозволяє забезпечити ефективний рівень безпеки даних.*

***Ключові слова:*** *база даних, конфіденційність, цілісність, доступність, захист інформації, загроза, ідентифікація, аутентифікація, авторизація, шифрування, моніторинг, SQL-ін'єкція, IDS/IPS, доступ.*

**Постановка проблеми**

Ми живемо у вік технологій, коли діяльність будь-якої організації супроводжується активним використанням комп'ютерів та пов'язана з обробкою великих обсягів інформації, а доступ до цієї інформації має велике коло осіб. Важливим стає питання своєчасного та повного отримання інформації та обробки складних даних у розподілених системах. Постійно зростає значення збереження цілісності та доступності відкритої інформації, а також забезпечення конфіденційності обмеженого доступу до даних в автоматизованих системах. Поряд із позитивним впливом інформатизації основних сфер суспільного життя, слід враховувати й загрозу вразливості цих сфер перед інформаційними технологіями. Це підкреслює важливість ретельного вибору технологій з урахуванням конкретного об'єкту інформаційної діяльності, його значення для суспільства та можливих наслідків порушення цих технологій. Дані безумовно стали таким же багатством як корисні копалини, виробничі чи людські ресурси, а їх захист від кіберзагроз, різного роду посягань та злочинів стає актуальною та надзвичайно важливою проблемою. За таких обставин зловмисні дії або ж навіть просто некомпетентність якогось співробітника організації можуть нанести непоправні втрати для організації. Тут мова може йти не лише про викрадення, знищення чи спотворення інформації, а навіть елементарне блокування доступу до важливих даних на достатньо тривалий період. Найкращим способом ефективної обробки інформації є база даних. База даних (БД) – це сукупність логічно взаємопов'язаних даних, призначених для задоволення інформаційних потреб її користувачів.

З плином часу та з розвитком новітніх технологій, кіберзлочинність зростає у масштабах та складності, ставлячи під загрозу не лише великі корпорації та урядові установи, а й звичайних користувачів. Таким чином, виникає термінова потреба в розробці та впровадженні стратегій та інноваційних підходів до захисту такої важливої складової інформаційної системи, як бази даних, які забезпечували б їхню недоступність для несанкціонованого доступу, зміни чи видалення. Запобігання кібератакам та ефективний захист інформації стають найважливішими завданнями в умовах постійної еволюції цифрового середовища. У цьому контексті, стратегії безпеки даних постійно переглядаються та адаптуються до нових викликів, включаючи в себе розробку передових систем ідентифікації та аутентифікації, впровадження шифрування даних на всіх рівнях, вдосконалення механізмів виявлення та реагування на інциденти, а також навчання персоналу з питань кібербезпеки.

Захист інформації – це сукупність організаційно-технічних заходів і правових норм, які запобігають нанесенню шкоди інтересам власників інформації. Тривалий час методи захисту інформації розроблялися лише державними органами, а їх впровадження вважалось виключним правом держави. Однак в останні роки, з розвитком комерційної та підприємницької діяльності постійно зростає кількість спроб отримання несанкціонованого доступу до конфіденційної інформації, а проблеми її захисту привернуло увагу багатьох фахівців та експертів в різних країнах. Як наслідок, зросла потреба в захисті конфіденційної, чутливої інформації, яка зберігається у базах даних. Тому захист баз даних – це одна з найскладніших та найбільш актуальних проблем сьогодення.

**Аналіз останніх досліджень та публікацій**

Останні дослідження та публікації в галузі захисту баз даних являють собою важливий напрямок у сучасній інформаційній безпеці. З моменту появи перших баз даних зросла як їхня кількість, так і різноманітність, що призвело до збільшення загроз їхньому захисту. У зв'язку з цим виникає потреба в постійному удосконаленні методів та технологій захисту, а також вивченні нових загроз та вразливостей. Оскільки база даних є важливим корпоративним ресурсом, безпека бази даних є важливою складовою загального плану безпеки інформаційних систем будь-якої організації. Окрім необхідності зберігати та захищати дані для безперебійного функціонування організації, розробники баз даних несуть відповідальність за захист конфіденційності осіб, дані про яких зберігаються. Загалом збиток від порушення захисту БД виходить за межі фактичної втрати або розголошення чутливих, конфіденційних даних і збитку бренду, оскільки організації зазнають значних фінансових витрат, що пов'язані з відновленням та багаторічними претензіями щодо судової відповідальності. Організації, чутливі до ризику повинні бути на крок попереду в безпеці своїх баз даних, щоб захистити свої дані від а безліч зовнішніх і внутрішніх загроз.

Захист бази даних – це набір заходів, спрямованих на забезпечення безпеки, цілісності, доступності і конфіденційності даних, які зберігаються в базі даних. Основні вимоги до захисту даних в БД фактично повністю збігаються з вимогами до захисту даних в комп'ютерних системах та охоплюють як фізичне середовище даних та системних програм так і захист від несанкціонованого доступу зловмисниками чи шкідливими програмами до даних, які обробляються, зберігають чи передаються.

Цілісність даних забезпечує те, що дані в базі даних залишаються точними, цілісними і невідмінними протягом усього їх життєвого циклу. Для забезпечення цілісності бази даних використовуються методи перевірки даних, шифрування, контроль доступу тощо. Доступність означає, що дані в базі даних доступні користувачам у відповідний час. Заходи, спрямовані на забезпечення доступності, включають в себе резервне копіювання даних, моніторинг стану бази даних, встановлення механізмів відновлення в разі виникнення неполадок і масштабування інфраструктури.

Конфіденційність забезпечує, що доступ до даних в базі даних мають лише уповноважені користу-

вачі. Це може бути досягнуто за допомогою шифрування даних, механізмів автентифікації і авторизації, а також шляхом застосування прав доступу до об'єктів бази даних (таких як таблиці, представлення і процедури).

Для забезпечення ефективного захисту даних в БД, необхідно визначити та проаналізувати набір сервісів та їх механізмів, що забезпечують відповідний захист. Перші дослідження в даному напрямку були зумовлені в основному потребами військової сфери, де проблеми захисту загалом є доволі критичною. Подальший стрімкий розвиток баз даних та впровадження їх у всі сфери життєдіяльності людства, їх стратегічне значення для багатьох організацій та постійні загрози кібератак, що можуть привести до несанкціонованого доступу до даних сприяв поширенню захисту БД у різних сферах застосування. На сьогоднішній день, враховуючи значну актуальність даного напрямку, фахівцями з безпеки даних розроблена велика кількість різноманітних підходів та методи щодо захисту БД [1-5].

Якщо розглядати історично, то розвиток методів захисту БД в основному був реагуванням на безпосередню еволюції. Баз даних та відповідні відомі загрози від зловмисників. Як правило нові підходи та методи, що з'являлися орієнтувалися на подолання відомих існуючих загроз та вразливостей, розроблення підходів по організації захищеного доступу та організації захисту даних, що зберігаються.

Одним з актуальних напрямів досліджень є аналіз методів криптографічного захисту даних в базах даних [6-7]. Для забезпечення конфіденційності і цілісності інформації дослідники розробляють нові алгоритми шифрування, які відповідають сучасним вимогам безпеки. Вивчаються методи аутентифікації користувачів і контролю доступу до баз даних, що дозволяє уникнути несанкціонованого доступу та зловживань.

Останніми роками дедалі більше уваги приділяється захисту баз даних в хмарних середовищах [8], де існує ризик несанкціонованого доступу через віртуалізовані середовища та мережеві канали. Дослідження в цій області спрямовані на розробку імунітету до нових видів загроз та забезпечення безпеки даних в умовах розподіленої обробки.

Нарешті, важливим аспектом є розробка стандартів безпеки для баз даних [10-11], які б стандартизували найкращі практики та методи захисту, що сприятиме забезпеченню єдиної методології захисту даних, яка б діяла як у великих корпоративних середовищах, так і в невеликих підприємствах.

Інші різноманітні аспекти безпеки баз даних описані у великій кількості фахових праць як вітчизняних так і закордонних дослідників. Зокрема в [11] підкреслюються особливі проблеми захисту баз даних. Це означає, що чим складніша база даних, тим більше заходів безпеки потрібно для її захисту. У [12] описується база даних як основа будь-якої інформаційної системи. Тому бази даних потрібно підтримувати у високій якості, щоб забезпечити якість і надійність всієї інформаційної системи. У [13] коротко описані конкретні вразливості, які можуть виникнути при використанні сучасних баз даних, а також запропоновано набір правил і дій, які можуть зменшити ризики, пов'язані з порушенням цілісності, конфіденційності та доступності даних. У [14] розглядаються основи баз даних, тобто їх призначення, функції та ролі. Зосереджена увага на основах управління безпекою баз даних та відповідних технологіях, що їх реалізують. У [15] питання безпеки баз даних розглядаються з точки зору безпеки додатків, які підключаються до баз даних. У [16] розглядаються загрози безпеці існуючих баз даних та їх усунення з акцентом на конфіденційність даних. Розглядаються способи розширення ланцюжка аутентифікації користувачів.

Загалом, варто зазначити, що останні дослідження та публікації в галузі захисту баз даних відображають неабияку активність та інноваційний підхід до вирішення проблем інформаційної безпеки. Ці напрацювання не лише підвищують рівень захисту баз даних, а й дозволяють підтримувати високий стандарт безпеки в умовах інформаційного середовища, яке швидко змінюється.

**Мета та постановка завдання**

Постановка завдання полягає в розгляді сучасних методів захисту баз даних в умовах зростаючих кіберзагроз. Основною метою даної роботи є аналіз та огляд різноманітних стратегій та підходів, таких як використання шифрування, аутентифікація та авторизація, оброблення вхідних даних, встановлення рівнів доступу та інших засобів, спрямованих на забезпечення безпеки, цілісності, конфіденційності та доступності даних. Також в рамках статті необхідно проаналізувати виклики та перешкоди, що стоять на шляху ефективного захисту баз даних та запропонувати певні рекомендації щодо вибору оптимальних стратегій захисту та сценаріїв використання даних.

**Виклад основного матеріалу дослідження**

*Основні загрози безпеці баз даних та методи їх уникнення*

Загроза – це будь-яка дія або подія, яка може стати причиною порушення безпеки бази даних або її вмісту. Причиною виникнення загрози може стати людина, явище або збіг обставин. Навмисні загрози здійснюються людьми, що можуть бути авторизованими або неавторизованими користувачами. Світова статистика вказує на те що приблизно 60% витоків інформації з БД стається саме з вини співробітників організації і доволі часто це відбувається не навмисно, а через некомпетентність даних працівників. Проте постійно зростає і кількість атак на БД від зовнішніх зловмисників, яких в першу чергу цікавить наступна інформація, що зберігається в БД: фінансова інформація організації, її внутрішня операційна інформація, персональна інформація співробітників, інформація про партнерів, клієнтів тощо. Загалом бази даних на-

лежать до найбільш скомпрометованих активів і основною причиною можна назвати масове використання БД у переважній більшості організацій і водночас те, що дані організації не приділяють достатньої уваги їх захисту. Врахування відомих загроз та вразливостей БД, їх аналіз, прогнозування та своєчасне виявлення нових, дозволить підготувати відповідні засоби та заходи безпеки і забезпечити надійний захист. Саме тому даному напрямку приділяється велика увага [17-20]. Можемо виділити наступні основні методи захисту баз даних.

*1. Метод захисту – ідентифікація, аутентифікація та авторизація*

Основна вимога безпеки полягає в тому, що ви повинні знати своїх користувачів. Ви повинні ідентифікувати їх, перш ніж ви зможете визначити їхні привілеї та права доступу, а також щоб ви могли перевірити їхні дії з даними. Тому для будь-якої захищеної бази даних обов'язковими є процедури ідентифікації, аутентифікації та авторизації.

Ідентифікація в базах даних – це процес встановлення, хто саме (користувач, процес або програма) намагається отримати доступ до системи або бази даних. Ідентифікація визначає суб'єкта, який намагається здійснити доступ. Це може бути ідентифікація за допомогою імені користувача, ID, електронної адреси тощо.

Аутентифікація в базах даних передбачає перевірку ідентифікованої сутності (наприклад, користувача) для переконання, що вона є тим, за кого вона себе видає, зазвичай за допомогою пароля, біометричних даних, сертифікатів тощо. Автентифікація не надає жодних привілеїв для певних завдань. Вона лише встановлює, що СУБД вірить, що користувач є тим, за кого себе видає, і що користувач вірить, що СУБД також є передбачуваною системою. Автентифікація є необхідною умовою для авторизації. Зловмисники можуть застосовувати різні підходи, наприклад обхід автентифікації, пароль за замовчуванням, підвищення привілеїв, підбір пароля за допомогою грубої сили та атаку веселки, коли вони намагаються скомпрометувати ідентифікацію та автентифікацію користувача.

Авторизація в базах даних – це процес визначення, чи має ідентифікована та аутентифікована сутність право на доступ до певних ресурсів чи виконання певних операцій. Авторизація встановлює рівень доступу, який має користувач після успішної ідентифікації та аутентифікації. Це може бути на основі ролей, прав доступу чи інших політик безпеки.

Для забезпечення безпеки баз даних інформація про ідентифікацію дозволяє системі знати, хто користується базою даних, а аутентифікація перевіряє, чи ця особа є дійсною, а авторизація визначає, які дії вона може здійснювати в межах цієї системи.

Відсутність належних механізмів ідентифікації, аутентифікації та авторизації в базі даних може призвести до різних серйозних загроз для безпеки та конфіденційності даних, які в ній зберігаються серед яких можна виділити наступні:

– несанкціонований доступ до даних. Це найбільш розповсюджений та різноманітний вид комп'ютерних порушень. Без відповідних механізмів автентифікації зловмисники можуть отримати доступ до бази даних, навіть якщо вони не мають легітимних прав доступу. Таким чином можна спровокувати витік конфіденційної інформації або зміну або видалення даних;

– маніпулювання даними. Якщо користувачі не ідентифікуються та не аутентифікуються, зловмисники можуть використовувати невідомі акаунти для маніпулювання даними в базі. Негативним наслідком буде спотворення інформації або створення фальшивих записів в базі даних;

– маскарад. Це виконання будь-яких дій одним користувачем від імені іншого користувача, який має відповідні повноваження. Метою маскараду є приписування будь-яких дій іншому користувачеві або присвоєння повноважень і привілеїв іншого користувача. Прикладами реалізації маскараду є: вхід у систему під ім'ям та паролем іншого користувача (цьому маскараду передує перехоплення пароля);

– незаконне використання привілеїв. Кожен користувач БД отримує свій набір привілеїв для роботи з нею, який перевіряється під час авторизації. Несанкціоноване захоплення привілеїв, наприклад, за допомогою маскараду, призводить до того, що порушник зможе виконувати певні дії в обхід системи захисту. Варто зазначити, що незаконне захоплення привілеїв можливе або за наявності помилок у системі захисту, або через недбалість адміністратора при управлінні системою і призначенні привілеїв;

– втрата конфіденційності. Без надійної системи аутентифікації немає гарантії, що тільки уповноважені користувачі отримують доступ до конфіденційних даних. Відсутність системи аутентифікації призведе до витоку конфіденційної інформації, такої як особисті дані користувачів, фінансова інформація тощо;

– втрата цілісності даних. Без надійної системи немає гарантії, що дані не будуть змінені або видалені несанкціонованими користувачами. Небажаними наслідками будуть порушення цілісності даних і спотворення інформації в базі даних;

– втрата доступності. Недостатні механізми ідентифікації та аутентифікації можуть призвести до використання ресурсів бази даних зловмисниками, що може призвести до перевантаження системи або навіть до відмови у обслуговуванні;

– невідомість активності користувачів. Без відповідної ідентифікації та аутентифікації може бути складно відслідковувати дії користувачів в базі даних, що зробить складнішим виявлення та розслідування можливих інцидентів безпеки.

Тому для уникнення цих перерахованих загроз, важливо впровадити надійні механізми ідентифікації, аутентифікації та авторизації, такі як сильні паролі, двофакторна аутентифікація, дискреційна, мандатна чи рольова модель доступу та моніторинг активності користувачів.

Основні механізми ідентифікації, аутентифікації та авторизації:

– паролі. Використання комплексних паролів та їх регулярна зміна є одним із важливих аспектів безпеки в базах даних. Комплексні паролі складаються з комбінації різних видів символів, таких як великі та малі літери, цифри, спеціальні символи. Їх довжина також має бути достатньою, щоб ускладнити процес вгадування, а саме не менше 8 символів. Наприклад, пароль "P@ssw0rd!" є складним через використання різних типів символів та довжину. Кожен користувач бази даних повинен мати унікальний пароль. Це важливо, оскільки використання одного і того ж пароля для різних облікових записів збільшує ризик, що в разі витоку пароля зловмисник отримає доступ до всіх систем, до яких використовується цей пароль. Зміна паролів через певні періоди часу допомагає запобігти можливості злому пароля методом перебору. Частіше за все рекомендується змінювати паролі щонайменше кожні 60-90 днів. Для зручності керування складними та унікальними паролями рекомендується використовувати парольні менеджери. Вони дозволяють зберігати всі паролі в безпечному місці та автоматично генерувати та заповнювати форми входу. Використання комплексних паролів, їх регулярна зміна, визначення мінімальної кількості невдалих спроб входу в систему, обмеження доступу до файлу паролів, а також навчання користувачів безпечному використанню паролів допомагають уникнути несанкціонованого доступу до бази даних через перебір паролів, чи їх перехоплення та інші атаки;

– двофакторна аутентифікація. Двофакторна аутентифікація (2FA) – це метод захисту, який вимагає від користувача не лише введення пароля, але й додаткового фактора ідентифікації для підтвердження своєї особи. Це дозволяє створити додатковий шар захисту для облікового запису і ускладнює завдання зловмисникам, які намагаються незаконно отримати доступ до системи. Перевагою двофакторної аутентифікації 2FA є надання значно вищого рівня захисту, порівняно з традиційною аутентифікацією лише за допомогою пароля. Якщо зловмисники вкрадуть пароль, вони все ще матимуть відсутній другий фактор аутентифікації для входу в обліковий запис. Використання додаткового фактора ускладнює спроби зламати пароль методом перебору. Використання 2FA демонструє високий рівень обережності та забезпечує більшу впевненість у безпеці облікового запису. Принцип роботи двофакторної аутентифікації наступний: користувач вводить свій обліковий запис і пароль для входу в систему. Після успішного введення пароля, система запитує додатковий фактор аутентифікації, такий як одноразовий код, біометричні дані (відбиток пальця, розпізнавання обличчя), фізичний токен або спеціальний мобільний додаток. Після введення додаткового фактора аутентифікації система перевіряє його на вірність. Якщо обидва фактори (пароль і додатковий) вірні, користувач отримує доступ до системи. В якості другого фактору двофакторної аутентифікації можна використати:

а) одноразові коди (OTP). Генерується одноразовий код, який відправляється на мобільний телефон або інші пристрої через SMS або спеціальні додатки;

б) біометричні дані: Використання відбитка пальця, розпізнавання обличчя тощо;

в) фізичні токени: Карти, ключі або інші пристрої, які генерують коди або можуть бути використані для аутентифікації. Існують пасивні токени (токени з пам’яттю), які тільки зберігають, але не обробляють інформацію та активні (інтелектуальні токени);

г) мобільні додатки: Додатки, що генерують одноразові коди або дозволяють підтверджувати аутентифікацію через спеціальний механізм.

Використання механізмів двофакторної аутентифікації дає змогу ефективно забезпечити безпеку облікових записів в базах даних, оскільки надає додатковий шар захисту від несанкціонованого доступу та атак злому;

– дискреційна модель доступу. Формується матриця доступу або списки управління доступом, в яких буде встановлений перелік користувачів та перелік відповідних дозволених їм операцій, щодо кожного з об’єктів БД. Фактично визначається множина дозволених відношень у вигляді ланок “суб'єкт доступу-тип доступу-об’єкт доступу” Існує три способи реалізації дискреційної моделі доступу в БД:

а) добровільне керування доступом при якому вводиться поняття “володіння об’єктами” і права на доступ до об’єктів визначають самі їх власники. Належні власнику привілеї можуть бути передані іншим авторизованим користувачам. СУБД повинна контролювати весь ланцюжок надання привілеїв користувачам із відповідним вказанням хто і як їх надав. Фактично таке керування доступом являє собою децентралізований принцип організації та управління процесом розмежування доступу;

б) примусове керування доступом, що передбачає ведення єдиного централізованого адміністрування доступом. Визначається один довірений суб’єкт, в ролі якого як правило виступає адміністратор і саме він визначає права доступу усіх інших суб’єктів до об’єктів БД. Це централізований принцип організації та управління процесом розмежування доступу, який є менш гнучким та не завжди відповідає потребам користувачів;

в) змішаний принцип організації та управління процесом розмежування доступу, яка власне і є найбільш використовуваним. Згідно даного принципу визначена частина повноважень на доступ до об’єктів

встановлюється адміністратором, а інша частина безпосередньо самими власниками об'єктів;

– мандатна модель доступу. У цій моделі доступ до даних та інших ресурсів контролюється за допомогою мандатів, які призначаються кожному користувачеві та які визначають його привілеї. Ці мандати можуть бути базовані на ролях, ідентифікаторах користувачів або інших атрибутах. Коли користувач намагається отримати доступ до ресурсу або виконати операцію, його мандат порівнюється з мандатами, необхідними для доступу, і здійснюється відповідна перевірка доступу. Як правило дана модель доступу в основному використовується в спеціальних БД для державних, військових та інших організаціях із жорсткою структурою. Основною метою мандатного розмежування доступу до об'єктів є запобігання витоку інформації з об'єктів з високою міткою конфіденційності в об'єкти з низькою міткою конфіденційності (протидія створенню каналів передачі інформації «зверху вниз»);

– рольова модель доступу. Реалізація ролевої моделі доступу (Role-Based Access Control, RBAC) – це стратегія контролю доступу, яка визначає, які дії можуть виконувати користувачі в системі на основі їхніх ролей або повноважень. У цій моделі доступ до об'єктів бази даних (таких як таблиці, стовпці, процедури тощо) призначається на основі ролей, які надаються користувачам. В рольовій моделі кожному користувачеві призначається одна чи декілька ролей, які відповідають їхнім функціональним обов'язкам або позиціям в організації. Наприклад, в БД можуть бути такі ролі, як "адміністратор", "користувач", "менеджер", "аналітик" тощо. Кожна роль має набір повноважень, які визначають, які операції користувач з цією роллю може виконувати з об'єктами бази даних. Наприклад, адміністратор може мати повний доступ до всіх таблиць і стовпців, тоді як звичайний користувач може мати обмежений доступ лише до певних даних. Адміністратор системи або адміністратор бази даних визначає, які ролі доступні в системі і які користувачі мають ці ролі. Це може бути здійснено через адміністративний інтерфейс або шляхом прямого визначення ролей кожному користувачеві. Система бази даних використовує ролі та повноваження для контролю доступу до об'єктів бази даних. При спробі користувача виконати певну операцію система перевіряє, чи має користувач необхідні повноваження через його роль. Адміністратор системи може змінювати набір повноважень, які мають ролі, а також назначати або забирати ролі у користувачів в залежності від їхніх потреб або зміни в організаційній структурі. Система RBAC дозволяє впровадити гнучкі та динамічні правила розмежування доступу при цьому гарантувати надійний захист шляхом чіткого визначення ролей та відповідних їм прав. Також можна вести аудит доступу, де в журналах доступу реєструється, хто, коли і які операції виконував в базі даних, можна здійснювати моніторинг з метою виявлення підозрілих або несанкціонованих дій.

*2. Метод захисту – захист сервісів, протоколювання та моніторинг активності користувачів*

В сучасних СУБД передбачені механізми резервного копіювання і аудиту як базові складові системи безпеки. Суть резервного копіювання полягає в зберіганні копії БД. При необхідності (несанкціоноване видалення або модифікація БД) з цієї копії відновлюється остання вірна версія БД. Резервне копіювання відбувається із визначеною періодичністю. Наприклад, після жорсткого збою фізична база даних виявляється зруйнованою. Тому для відновлення потрібно мати її резервну копію.

Зазвичай резервне копіювання БД виконується за фактом переповнення журналу транзакцій. Для цього у файлі журналу встановлюється так звана «жовта зона», після досягнення якої запуск нових транзакцій не проводиться. Система чекає на закінчення всіх існуючих транзакцій. Після цього робочі буфери журналу та бази даних виштовхуються у зовнішню пам'ять. Створений таким чином стан бази даних копіюється на резервний носій, а файл журналу очищається. Можливо також створення резервна копія журналу.

При відновленні після жорсткого збою відновлюється стан БД на момент останнього копіювання, а потім за поточним журналом реєстрації повторно виконуються всі транзакції, що успішно завершилися до моменту збою.

Резервне копіювання можливо наступних видів:

– повне – резервуються всі дані;

– диференційоване – здійснюється резервування всіх сторінок даних, змінених з останнього повного резервного копіювання;

– журналу транзакцій – виконується резервування всіх транзакцій у журналі;

– файлу або файлової групи. виконується резервування всіх даних, що містяться у файлі або файловій групі;

– файлове диференційоване – здійснює резервування всіх сторінок даних, модифікованих з моменту останнього резервного копіювання файлу або файлової групи.

Проста модель відновлення ідеально підходить тим базам даних, яким потрібно забезпечення атомарності транзакцій, але не обов'язкова підтримка їх живучості. Журнал зберігатиме транзакції, запис яких до бази даних ще не підтверджено; простір ж, відведений для зберігання всіх інших транзакцій, звільняється для повторного використання. Оскільки журнал транзакцій у цій моделі є лише тимчасовим місцем зберігання, відпадає потреба у його резервуванні. Ця модель відновлення має ряд переваг. По-перше, журнал транзакцій має маленькі розміри, проте розплачуватися за це доведеться втратою всіх транзакцій, виконаних з останнього повного або диференційованого резервування. План відновлення, що базується на простій моделі, дозволяє виконувати повне резервування раз на тиждень, а диференційоване – наприкінці кожного дня тижня. Повна та диференційовані

резервні копії замінюються, коли буде виконано наступне повне резервне копіювання.

Мета реалізації протоколювання та аудиту полягає в забезпеченні підзвітності користувачів та адміністраторів, можливості відтворення послідовності подій та надання інформації для виявлення та аналізу проблем. Розгорнуте або деталізоване протоколювання, хоч і корисним проте може мати негативний вплив на продуктивність сервісів, а також ускладнює процес аудиту, що призводить до зменшення, а не підвищення рівня інформаційної безпеки. Організація інтегрованого протоколювання та аудиту в розподіленій різнорідній системі є складною проблемою.

Перший крок в аудиті доступу – це ведення журналів всіх дій, які виконуються користувачами в базі даних, що включає зміни в даних, запити на вибірку даних, спроби входу в систему, зміни структури бази даних тощо. Інформація, зібрана в результаті аудиту, зазвичай зберігається у спеціальній системі або журнальних файлах для подальшого аналізу та виявлення випадків порушень безпеки. Після того, як журнали доступу створені, проводиться моніторинг активності користувачів для виявлення будь-яких аномальних або підозрілих дій. Наприклад, аналіз надмірної активності, незвичайні запити до БД, спроби несанкціонованого доступу та інші підозрілі патерни. Під час моніторингу активність користувачів аналізується, щоб виявити будь-які підозрілі дії або аномалії, а саме виявлення спроб виконання неавторизованих запитів, зміни даних користувачами без необхідних прав або спроби несправедливого використання облікових записів. Якщо в результаті аудиту доступу виявляються підозрілі дії або порушення безпеки, вживаються відповідні заходи, які можуть включати призупинення облікових записів, видалення несанкціонованих змін або сповіщення відповідних служб безпеки. Маючи повний журнал транзакцій, можна аналізувати, як доступ до даних та їх зміни відбуваються з часом. Це дозволяє виявляти типові зразки доступу та змін і допомагає запобігати витоку інформації, виявляти небезпечні зміни та виявляти підозрілу активність в реальному часі. Процес аудиту доступу є постійним, і він повинен регулярно оновлюватися та вдосконалюватися відповідно до змін в системі та з розвитком нових загроз безпеці, оскільки це допомагає виявляти незвичайну або підозрілу активність та забезпечувати відповідність з вимогами безпеки.

*3. Метод захисту – шифрування*

Шифрування в сфері баз даних є важливим етапом захисту і конфіденційності інформації. Можна застосувати надійні системи захисту БД, проте не можна гарантувати неможливість їх злому, витоку чи перехоплення інформації. Саме тому шифрування може стати останнім рубежем захисту БД, який гарантуватиме, що навіть якщо зловмисник обійде використані системи захисту, то він отримає лише набір безглуздих символів. Шифрування дозволяє перетворювати звичайний текст в незрозумілу форму, яка може бути розкрита тільки за допомогою спеціального ключа. У сфері баз даних шифрування використовується для захисту конфіденційних даних від несанкціонованого доступу. За допомогою шифрування баз даних, навіть якщо зловмисник отримає доступ до файлів чи таблиць, вони не зможуть прочитати конфіденційну інформацію без відповідного ключа. Це забезпечує додатковий рівень безпеки для даних, зменшуючи ризик їх несанкціонованого доступу та зловживання. Деякі СУБД пропонують власні вбудовані засоби шифрування даних, що в них зберігаються, виконуючи таке шифрування в режимі реального часу в момент звертання користувачів до даних.

Дані можна шифрувати на різних рівнях – від програми до механізму баз даних. Для більшості сучасних СУБД характерним є використання прозорого шифрування, яке відбувається непомітно для користувача, а його програми не змінюються. Такі системи прозорого шифрування можуть бути вбудованими в СУБД або ж бути зовнішніми відносно них.

Шифрування баз даних може також відігравати ключову роль у виконанні вимог щодо дотримання законодавства про конфіденційність даних, таких як Загальний регламент про захист даних (GDPR) в Європейському Союзі [20]. Захищаючи дані шляхом шифрування, організації можуть демонструвати свою здатність забезпечувати конфіденційність та приватність даних своїх користувачів і клієнтів.

Можливі наступні загрози для баз даних, які можуть виникнути без використання шифрування:

– несанкціонований доступ до даних – без шифрування, дані можуть бути доступні для несанкціонованих користувачів, які можуть отримати доступ до бази даних через недоліки в системі або атаки з зовнішнього середовища, що може призвести до витоку конфіденційної інформації;

– спілкування по мережі – під час передачі даних по мережі відбувається ризик перехоплення даних, які можуть бути використані для незаконного доступу або крадіжки інформації. без шифрування, дані можуть бути надіслані у відкритому вигляді, що робить їх вразливими для перехоплення;

– фізичний доступ до бази даних – якщо хакер або зловмисник отримує фізичний доступ до серверів бази даних, він може отримати доступ до даних, що зберігаються на цих серверах, якщо вони не зашифровані, що може допомогти на шляху до викрадення даних або витоку інформації;

– внутрішні загрози – навіть серед авторизованих користувачів можуть бути ті, хто намагається отримати доступ до конфіденційної інформації або вчинити інші шкідливі дії. Шифрування допомагає обмежити ризики, пов'язані з внутрішніми загрозами, навіть якщо користувачі мають доступ до бази даних.

Для захисту баз даних використовуються як симетричні, так і асиметричні шифрувальні алгоритми. У симетричному шифруванні один ключ використо-

вується як для шифрування, так і для розшифрування повідомлень. Це може бути ефективним для захисту баз даних, особливо якщо база велика і вимагає частого доступу. Можливі два варіанти такого симетричного шифрування: потокове та блочне. Популярні алгоритми симетричного шифрування, які можуть бути використані для захисту баз даних, включають:

– AES (Advanced Encryption Standard).AES використовує блочний шифр, де текст розбивається на блоки фіксованої довжини (наприклад, 128 біт), які потім шифруються один за одним. Кожен блок вхідних даних обробляється за допомогою ключа шифрування для створення шифртексту. Одним з головних переваг AES є його висока швидкодія та надійність шифрування при правильному використанні;

– DES (Data Encryption Standard). DES працює на блоках тексту фіксованого розміру (64 біта) та використовує ключ шифрування довжиною 56 біт, що робить його вразливим до атак перебору ключа на сучасних комп'ютерах. Він складається з циклічної комбінації заміщення (substitution) та перестановки (permutation) даних, які повторюються 16 раундів для кожного блоку даних. Це старіший алгоритм, який все ще може використовуватися в деяких випадках, але його використання зазвичай не рекомендується через обмежену довжину ключа.

Асиметричне шифрування використовує пару ключів, один з яких відкритий (для шифрування) і один закритий (для розшифрування). Це забезпечує високий рівень безпеки, але може бути менш ефективним для великих обсягів даних через обчислювальні витрати на операції з ключами. Алгоритми асиметричного шифрування включають:

– RSA (Rivest-Shamir-Adleman). Цей алгоритм є одним з найбільш популярних і застосовується в широкому спектрі застосувань, включаючи захист даних в базах даних. У RSA кожен користувач має два ключі: публічний ключ і приватний ключ. Публічний ключ використовується для шифрування повідомлень або перевірки цифрового підпису, тоді як приватний ключ використовується для розшифрування повідомлень або створення цифрового підпису. Ця асиметрична структура дозволяє безпечно обмінюватися ключами та захищати дані від несанкціонованого доступу;

– DSA (Digital Signature Algorithm). Використовується для цифрового підпису повідомлень та перевірки автентичності даних.

Послуги асиметричного шифрування можуть бути реалізовані на базі симетричних методів, проте тут необхідно мати надійну третю сторону, яка знатиме секретні ключі своїх клієнтів. Щоб компенсувати недоліки симетричних методів шифрування для захисту баз даних зазвичай використовують комбінацію кількох методів. Наприклад, дані можуть бути зашифровані симетрично за допомогою AES, а ключ для розшифрування цих даних може бути зашифрований асиметрично за допомогою RSA, а потім збережений в безпечному місці. Шифрування даних у базі даних є важливим аспектом забезпечення безпеки і конфіденційності інформації. Власне використання шифрування дозволить здійснити ефективний контроль цілісності інформації, оскільки на відміну від традиційних методів використання контрольних сум, які можуть ефективно протистояти лише помилкам випадкового характеру, криптографічні контрольна сума усуває можливості непомітної зміни даних. Для досягнення сильного захисту інформації, слід розглядати шифрування наступних типів даних:

– конфіденційна особиста інформація: імена, адреси, номери телефонів, адреси електронної пошти, соціальні страхові номери та інші особисті ідентифікатори;

– фінансова інформація: дані про кредитні картки, банківські рахунки, транзакції тощо;

– медична інформація: у базах даних медичної інформації можуть зберігатися дані про стан здоров'я, історії хвороб, рецепти та інша конфіденційна інформація про пацієнтів;

– комерційна та корпоративна інформація: інформація про бізнес-операції, клієнтів, партнерів, виробництво та інші комерційні аспекти можуть також містити конфіденційні дані, які потрібно захищати;

– аутентифікаційні дані: імена користувачів, паролі, ключі доступу та інші дані, що використовуються для аутентифікації та авторизації користувачів.

Шифрування цих типів даних допомагає запобігти несанкціонованому доступу до конфіденційної інформації та забезпечує високий рівень захисту даних у базі даних. Отже, використання шифрування для захисту бази даних є критично важливим для забезпечення конфіденційності, цілісності та доступності даних.

*4. Метод захисту - Оброблення вхідних даних*

Оброблення вхідних даних - це процес перевірки, фільтрації та обробки даних, що вводяться користувачами або зовнішніми джерелами перед тим, як вони будуть передані для виконання запитів до бази даних. Метою оброблення вхідних даних є запобігання вразливостям безпеки, таким як SQL-ін'єкції та інші атаки.

*SQL-ін'єкція (SQL Injection)*

SQL-ін'єкція (SQL Injection) є однією з найпоширеніших загроз для баз даних і полягає в тому, що зловмисник використовує неперевірені або неправильно оброблені вхідні дані для виконання шкідливого SQL-коду в базі даних. Недостатнє оброблення вхідних даних дозволяє зловмиснику виконувати SQL-команди, які можуть пошкодити базу даних або витягнути конфіденційну інформацію. Ця технологія особливо небезпечна тим, що будь-хто, хто має доступ до БД або Web-сайту організації, що звертається до БД, і здатний вводити дані в текстові поля, може стати джерелом атак за допомогою ін'єкцій SQL. Успішний експлойт SQL-ін'єкції може зчитувати конфіденційні

дані з бази даних, змінювати дані БД (вставити/оновити/видалити), виконувати операції адміністрування бази даних (такі як завершення роботи СУБД), відновлювати вміст певного файлу, наявного у файлі СУБД, а в деяких випадках навіть видають команди операційній системі. Дана загроза може бути реалізована наступними методами:

– введення SQL-запитів через веб-форми або URL-параметри. Зловмисник може ввести SQL-запит в поля введення веб-форми або як параметр в URL-адресі;

– перехоплення та модифікація SQL-запитів на боці клієнта: Зловмисник може використовувати інструменти для перехоплення та модифікації SQL-запитів, які відправляються з клієнта до сервера, для внесення шкідливого SQL-коду;

– ін'єкція через cookies або HTTP-заголовки: Зловмисник може вставити SQL-код в HTTP-заголовки або cookies, що передаються на сервер, і таким чином викликати виконання шкідливого SQL-коду на стороні сервера;

– використання вразливостей в програмному забезпеченні: Зловмисник може використовувати відомі вразливості в програмному забезпеченні, яке взаємодіє з базою даних, для виконання SQL-ін'єкцій;

– прикріплення шкідливого коду. Додавання термінатора інструкції до іншої інструкції SQL та вмісту введення дозволяє зловмиснику передати програмний код у рядок виконання. Термінатор інструкції SQL - це спеціальний символ або комбінація символів, яка використовується для вказівки кінця окремої SQL-інструкції. У більшості СУБД найпоширенішим термінатором є крапка з комою (;), яка використовується для розділення різних інструкцій SQL у запитах або скриптах.

Більшість вразливостей SQL-ін'єкцій можна виявити за допомогою різних сканерів веб-вразливостей, наприклад Burp Suite, OWASP ZAP. Також впровадження SQL можна виявити вручну, використовуючи наступні методи:

– відправка символу одинарної лапки ' та пошук помилок;

– відправка логічних умов, наприклад OR 1=1 і OR 1=2, та пошук відмінностей в відповідях додатку;

– відправка корисних даних, що призначені для запуску тимчасових затримок при виконанні запиту SQL, та пошук відмінностей в часі, які потрібні для відповіді.

*Приклади реалізації SQL-ін'єкцій*

Приклад 1. На сайті є форма для пошуку товарів по назві. Частина PHP-коду пошуку товару може виглядати наступним чином:

```
$name_tovar = $_GET["name_tovar"];
$sql = "SELECT * FROM Tovar WHERE tovar_name = " . $name_tovar;
```

Після того, як користувач введе назву товару, наприклад "ноутбук" то SQL інтерпретує даний запит як:

```
SELECT * FROM Tovar WHERE tovar_name = ноутбук
```

Але якщо користувач введе у поле для пошуку товару не просто слово "ноутбук" а "ноутбук; DROP TABLE Tovar;", то SQL інтерпретує даний запит як:

```
SELECT * FROM Tovar WHERE tovar_name = ноутбук; DROP TABLE Tovar;
```

Термінатор інструкції (;) завершує закладену програмістом операцію, після чого СУБД розглядає текст, що триває, як наступну інструкцію в пакеті. Завершальна лапка могла б призвести до помилки виконання, проте ця проблема просто вирішується шляхом додавання маркера коментаря (- для MS SQL Server). В результаті реалізації атаки є видалення таблиці Tovar з бази даних. Дана реалізація є можливою, оскільки більшість систем управління базами даних можуть виконувати декілька команд одночасно.

Приклад 2. Реалізація наступної атаки є можливою, якщо передати умову, результат якої завжди істинний (true), щоб дані виводились, незважаючи нінащо. Наприклад на сайті є форма входу, де користувачам необхідно вказати свої дані, а саме логін та пароль:

```
$username = $_POST["username"];
$password = $_POST["password"];
$sql = "SELECT * FROM Users WHERE username = \"" . $username . "\" AND password = \"" . $password . "\"";
```

Якщо користувач введе коректні дані, наприклад логін "root" і пароль "root", то SQL проінтерпретує даний запит як:

```
SELECT * FROM Users WHERE username = "root" AND password = "root"
```

Але якщо користувач введе логін і пароль наступним чином:

Логін - "invalid_login" OR "1"="1", Пароль - "invalid_password" OR "1"="1".

SQL інтерпретує це як:

```
SELECT * FROM Users WHERE username = "invalid_login" OR "1"="1" AND password = "invalid_password" OR "1"="1"
```

Результатом реалізації атаки буде вивід всіх користувачів бази даних, оскільки "1"="1" завжди істинне, незалежно від того, який логін і пароль буде введений користувачем.

*Методи захисту від SQL-ін'єкцій*

Щоб запобігти SQL-ін'єкціям, важливо правильно обробляти та перевіряти всі вхідні дані, використовувати параметризовані запити або ORM (Object-Relational Mapping) і використовувати механізми захисту від ін'єкцій, такі як функції очищення даних та використання параметрів запиту. Можна виділити наступні шляхи запобігання SQL-ін'єкцій:

– перевірка та фільтрація вхідних даних. Необхідно перевіряти всі вхідні дані на відповідність очікуваним форматам та обмеженням. Наприклад, якщо очікується введення числа, потрібно перевірити, що дані є числовими. Необхідно відфільтровувати та вилучати всі неприпустимі символи або конструкції,

такі як рядкові маркери або коментарі. Також необхідно обмежити число символів, що вводяться;

– використання параметризованих запитів. Використання параметризованих запитів, коли виконуються запити до бази даних. Це означає передачу параметрів окремо від самого запиту, що унеможливлює використання введених даних як частини SQL-коду;

– Object-Relational Mapping (ORM). ORM - це технологія, яка дозволяє працювати з об'єктами програми, а не безпосередньо з базою даних. ORM-фреймворки автоматично генерують SQL-запити на основі операцій з об'єктами програми. ORM може автоматично виконувати очищення та екранування вхідних даних перед їх використанням в SQL-запитах, забезпечуючи додатковий захист від ін'єкцій;

– функції очищення даних. Функції очищення можуть вилучати рядкові символи, які можуть бути використані для створення SQL-ін'єкцій, такі як апострофи чи подвійні лапки;

– використовувати збережені процедури та функції;

– ретельно визначити права доступу, щоб інструкції не мали дозволів на запуск DDL інструкцій;

– вимикати видачу необроблених помилок.

Проблемам SQL-ін'єкцій варто приділяти підвищену увагу. Якщо ваша програма передбачає введення даних з Інтернету, а ви не передбачили дії, спрямовані проти потенційних ін'єкцій, руйнація бази даних стане лише питанням часу. Наслідки SQL-ін'єкції можуть бути серйозними, включаючи втрату конфіденційної інформації, видалення або модифікацію даних, недоступність веб-сайту або додатку, а також потенційні проблеми з обробкою платежів та іншими фінансовими операціями.

*5. Метод захисту - системи виявлення та запобігання вторгнень*

Системи запобігання та виявлення вторгнень (Intrusion Prevention and Detection Systems, IDS/IPS) для баз даних є важливими компонентами в інформаційній безпеці будь-якої організації. Вони спеціально розроблені для виявлення аномальних або шкідливих дій, що відбуваються в базі даних, і для запобігання можливим загрозам. Без систем виявлення та запобігання вторгнень у базах даних існує ризик різноманітних загроз, які можуть призвести до серйозних проблем з безпекою даних, а саме:

– SQL-ін'єкція, суть якої ми розглянули вище;

– витік даних. Якщо зловмисник здобуде несанкціонований доступ до бази даних, він може скопіювати конфіденційну інформацію та використати її в шахрайських цілях;

– неавторизований доступ. Без систем контролю доступу зловмисники можуть отримати доступ до бази даних і виконувати різноманітні дії, включаючи зміну даних, видалення або використання їх у шахрайних цілях;

– DoS-атака. Атаки DoS можуть перевантажити базу даних запитами, що призводить до відмови в обслуговуванні для легітимних користувачів. Це може призвести до припинення роботи бізнес-процесів, що ґрунтується на даних.

Системи виявлення вторгнень (IDS) – аналізують трафік бази даних, щоб виявити потенційно шкідливі дії. Вони використовують різні методи, такі як аналіз відхилень від типової поведінки, підписи загроз (сигнатури) та аналіз аномалій, щоб виявити вторгнення. Наприклад, якщо система виявить спробу SQL-ін'єкції або нелегітимний доступ до бази даних, вона сповістить адміністратора про це.

Системи запобігання вторгненням (IPS) - ці системи більш активні, ніж IDS, оскільки вони не тільки виявляють вторгнення, а й намагаються автоматично їх зупинити. Наприклад, якщо в системі виявляється SQL-ін'єкція, IPS може автоматично заблокувати IP-адресу або сесію, з якої здійснюється атака.

Для запобігання та виявлення вторгнень можна також застосовувати:

– моніторинг доступу. Це важлива функція для систем запобігання вторгненням до баз даних. Вона дозволяє встановлювати права доступу до бази даних для користувачів та здійснювати моніторинг їх дії. Наприклад, система може сповіщати про намагання доступу до чутливих даних, якщо це здійснюється з облікових записів, які не мають на це прав;

– шифрування даних. Шифрування даних в базі даних може захистити їх від несанкціонованого доступу, навіть якщо зловмисник здобуде фізичний доступ до файлів бази даних. Системи IDS/IPS можуть також здійснювати моніторинг спроби розшифрувати дані або виявляти несправний доступ до зашифрованих ресурсів.

Дані функції ми розглядали вище.

Існує кілька видів систем виявлення та запобігання вторгнень (IDS/IPS) для баз даних, кожен з яких має свої особливості та методи роботи. Ось деякі з найпоширеніших:

– системи базовані на сигнатурах. Вони використовують заздалегідь визначені сигнатури або шаблони атак для виявлення вторгнень. Якщо трафік відповідає відомим атакам, система сповіщає адміністратора про потенційну загрозу;

– системи базовані на аномаліях: Ці системи аналізують типову поведінку користувачів та баз даних і сповіщають про будь-які аномальні дії або відхилення від звичайних патернів, що можуть свідчити про вторгнення;

– гібридні системи. Вони поєднують підхід базований на сигнатурах і базований на аномаліях для отримання більш точних та надійних результатів виявлення вторгнень.

Види систем запобігання вторгненням (IPS):

– блокування на основі сигнатур. Система IPS може блокувати трафік, який відповідає відомим атакам або сигнатурам;

– системи на основі правил. Вони встановлюють правила доступу до бази даних і можуть блокувати або обмежувати доступ до певних областей бази даних згідно з цими правилами;

– динамічне адаптивне блокування. Ці системи можуть автоматично реагувати на зміни у середовищі та на нові загрози, блокуючи або обмежуючи доступ до бази даних в реальному часі.

.

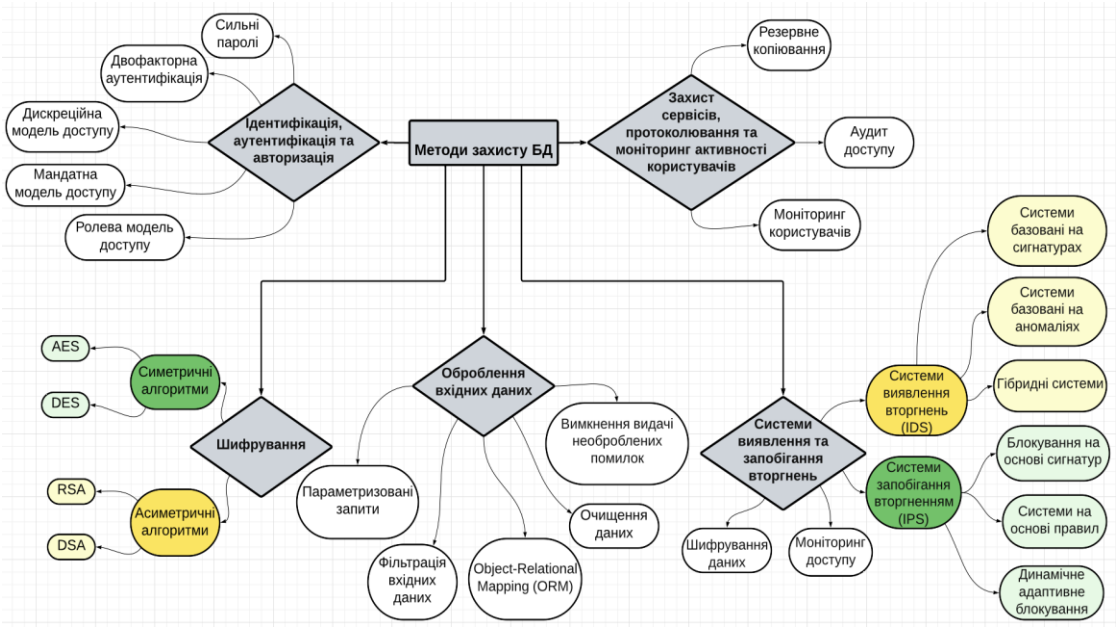


Рис. 1. Структура методів захисту та підходи їх реалізації

Таблиця 1

Характеристика методів захисту

| **Метод захисту** | **Ознаки захисту** |
|---|---|
| Ідентифікація, аутентифікація та авторизація | 1.Встановлення міцних паролів для аутентифікації користувачів і адміністраторів системи. Паролі повинні бути достатньо складними і довгими, а також періодично змінюватись.<br>2.Використання додаткового шару безпеки, де крім пароля користувач також підтверджує свою особу за допомогою додаткового фактора, такого як SMS-код, мобільний додаток або фізичний токен.<br>3.Встановлення чітких прав доступу до різних об'єктів і ресурсів в базі даних для користувачів і ролей. |
| Захист сервісів, протоколювання та моніторинг | 1.Регулярне створення резервних копій даних і їх зберігання в безпечному місці.<br>2.Слідкування за активністю користувачів в системі БД, виявлення незвичайних чи підозрілих дій, таких як невдали спроби авторизації, доступ до незвичайних об'єктів або обсягів даних.<br>3.Запис подій і дій, що відбуваються в системі БД, у спеціальний журнал аудиту, що дозволяє виявляти витоки даних, атаки або порушення політик безпеки, а також допомагає в розслідуванні інцидентів. |
| Шифрування | 1.Захист даних шляхом перетворення їх у незрозумілу форму (шифротекст) за допомогою алгоритмів шифрування, таких як AES, RSA тощо, що дозволяє зберігати дані безпечно навіть у випадку несанкціонованого доступу до БД. |
| Оброблення вхідних даних | 1.Перевірка вхідних даних на відповідність очікуваному формату, типу і діапазону значень. Наприклад, перевірка правильності формату електронної пошти або обмеження числових значень до допустимих меж.<br>2.Видалення або екранування потенційно небезпечних символів або команд з вхідних даних для запобігання SQL-ін'єкціям, XSS-атакам та іншим видам атак, що використовуються для впровадження коду.<br>3.Використання параметризованих запитів в SQL та інших мовах програмування для передачі параметрів окремо від запиту. |
| Системи виявлення та запобігання вторгнень | 1.IDS/IPS аналізують мережевий трафік на предмет аномалій або підозрілих дій, таких як надмірна кількість запитів до БД або незвичайні патерни доступу.<br>2.IDS виявляють вторгнення або спроби несанкціонованого доступу до БД, використовуючи сигнатури атак або аномальні поведінки, що вказують на потенційні загрози.<br>3.IPS може надавати активний захист шляхом блокування підозрілого трафіку або введення додаткових заходів безпеки для запобігання успішним атакам на БД. |

Робота таких систем полягає в постійному моніторингу трафіку баз даних, виявленні аномалій або відхилень від звичайної поведінки, а також у вжитті заходів для запобігання чи припинення атак. Для цього вони можуть використовувати різні методи аналізу даних, машинного навчання та інші технології. Основною метою цих систем є забезпечення безпеки баз даних шляхом виявлення та блокування потенційно шкідливої діяльності, що може стати загрозою для конфіденційності, цілісності та доступності даних. Підбиваючи підсумки (табл. 1) можемо виділити основні методи захисту бази даних, підходи їх реалізації та основні ознаки захисту (рис. 1).

**Висновки.** Бази даних є надзвичайно важливими компонентами сучасних інформаційних систем, які зберігають величезний обсяг конфіденційної і критично важливої інформації для різних організацій і підприємств. Проте, разом зі зростанням цінності цих даних, збільшується й загроза їхнього порушення. Міжнародне законодавство щодо захисту персональної інформації постійно вдосконалюється і стає все більш жорстким. Відповідальність за недоторканність конфіденційних відомостей покладається на організації, які їх збирають в процесі своєї діяльності. Причому в залежності від галузі та типу інформаційних активів, нормативні вимоги можуть істотно відрізнятися. Щоб бути конкурентними на ринку, українським компаніям необхідно відповідати цим стандартам, вкладати більше фінансових ресурсів в забезпечення захисту БД. Ефективний захист баз даних вимагає комплексного підходу, який включає в себе розгляд та впровадження основних методів захисту, а також прогнозування та своєчасне виявлення нових.

Однією з основних загроз для баз даних є несанкціонований доступ до них. Цьому можна запобігти за допомогою ідентифікації, аутентифікації та авторизації. Ці методи дозволяють перевіряти, хто саме намагається отримати доступ до системи, та чи має користувач на це право. Ще однією загрозою є можливість перехоплення або зламу шляхом атак на самі дані. Шифрування даних дозволяє забезпечити їх конфіденційність та неприступність для сторонніх осіб навіть у разі їхнього викрадення. Системи виявлення та запобігання вторгнень є ще одним важливим аспектом захисту баз даних. Вони дозволяють вчасно виявляти та реагувати на спроби несанкціонованого доступу або вторгнень. Нарешті, обробка вхідних даних включає в себе перевірку та фільтрацію вхідних запитів для запобігання SQL-ін'єкціям та іншим атакам, спрямованим на бази даних.

Усі ці методи захисту мають свої основні характеристики, такі як ефективність, складність впровадження, вартість та можливість адаптації до конкретних потреб організації. Враховуючи ці аспекти, організації повинні створити комплексну стратегію захисту баз даних, яка буде оптимальною для їхніх потреб та можливостей. Тільки такий підхід дозволить забезпечити надійний захист цінних даних в сучасному світі інформаційних технологій.

**Список літератури**

[1]. E. Bertino and R. Sandhu, "Database security - concepts, approaches, and challenges," in IEEE Transactions on Dependable and Secure Computing, vol. 2, no. 1, pp. 2-19, Jan.-March 2005, doi: 10.1109/TDSC.2005.

[2]. A. Mousa, M. Karabatak and T. Mustafa, "Database Security Threats and Challenges," 2020 8th International Symposium on Digital Forensics and Security (ISDFS), Beirut, Lebanon, 2020, pp. 1-5, doi: 10.1109/ ISDFS49300.2020.9116436.

[3]. B. Kumar and M. Hamed Said Al Hasani, "Database security — Risks and control methods," 2016 First IEEE International Conference on Computer Communication and the Internet (ICCCI), Wuhan, China, 2016, pp. 334-340, doi: 10.1109/CCI.2016.7778937.

[4]. Олег Дейнека, Олег Гарасимчук. Дослідження проблем класифікації та безпечного зберігання даних // Безпека інформації. 2023. Т. 29, № 2. С. 147-153.

[5]. Oleg Deineka, Oleh Harasymchuk. The challenges and strategies of storing large volumes of data in the modern world // Ukrainian Information Security Research Journal // Vol. 25 No. 4 (2023), pp. 197-207.

[6]. Ларченко, М. . (2022). Сучасні проблеми криптографічного захисту баз даних. Технічні науки та технології, (3(29), 102-113. doi: 10.25140/2411-5363-2022-3(29)-102-113.

[7]. Shmueli, Erez & Vaisenberg, Ronen & Elovici, Yuval & Glezer, Chanan. (2010). Database encryption. ACM SIGMOD Record. 38. 29. doi: 10.1145/1815933.1815940.

[8]. Shcherbinina Ye. Безпека бази даних і вивчення методів шифрування даних в хмарному сховищі / Ye. Shcherbinina, B. Martseniuk, A. Filonenko // Системи управління, навігації та зв'язку. Збірник наукових праць. Полтава: ПНТУ, 2020. Т. 3 (61). С. 104-106. doi: 10.26906/SUNZ.2020.3.104.

[9]. Database Security Standard. Version 1.3// Los Angeles County Information Technology Standards. Developed by: Application Security Engineering Team. Revision Date: 07/11/17.

[10]. Security Standard – Database Management System (SS-005). Version 2.0/ Chief Security office. Date: 14/06/23.

[11]. E. Burtescu, "Database Security, Attacks and Control Methods", Journal of Applied Sciences and Technology, pp. 449-453, 2009.

[12]. J. Juma, and D. Makupi, "Understanding Database Security Metrics: A Review", Mara International Journal of Scientific & Research Publications, vol. 1, no. 1, pp. 40-48, 2017.

[13]. В. А. Пєвнєв, "Безпека баз даних: загрози та превентивні заходи", Сучасні інформаційні системи, т. 2, № 1, С. 69-72, 2018, doi: 10.20998/2522-9052.2018.1.13.

[14]. P. Paul, and P. S. Aithal, "Database Security: An Overview and Analysis of Current Trend", International Journal of Management, Technology, and Social Sciences (IJMTS), vol. 4, no. 2, pp. 53-58, 2019, doi: https://dx.doi.org/10.2139/ssrn.3497728.

[15]. R. A. Teimoor, "A Review of Database Security Concepts, Risks, and Problems'', UHD Journal of Science and Technology, vol. 5, no. 2, pp. 38-46, 2021, doi: 10. 21928/uhdjst.v5n2y2021. pp. 38-46.

[16]. A. Mousa, M. Karabatak, and T. Mustafa, "Database Security Threats and Challenges", in Proc. 8th International Symposium on Digital Forensics and Security (ISDFS), Remote/ Online, 2020, pp. 1-5, doi: 10.1109/ISDFS49300.2020.9116436.

[17]. Pevnev, V., & Kapchynskyi, S. (2018). Database security: threats and preventive measures. Advanced In formation Systems, 2(1), 69-72. doi:10.20998/2522-9052. 2018.1.13.

[18]. Wang, Yong & Xi, Jinsong & Cheng, Tong. (2021). The Overview of Database Security Threats' Solutions: Traditional and Machine Learning. Journal of Information Security. 12. 34-55. doi:10.4236/jis.2021.121002.

[19]. Wang, Y. , Xi, J. and Cheng, T. (2021) The Overview of Database Security Threats' Solutions: Traditional and Machine Learning. Journal of Information Security, 12, 34-55. doi: 10.4236/jis.2021.121002.

[20]. https: // zakon.rada.gov.ua/laws/show/984_ 008-16#Text.

**УДК 004.6**

***Harasymchuk O., Buzhovych O. Strategies and innovative approaches to database protection in the age of growing cyber threats***

***Abstract.*** *In today's digital environment, where databases play a critical role in storing and processing important information for various spheres of human activity, protection against cyber threats becomes an extremely urgent task. This accordingly places new demands and responsibilities on organizations. Modern technologies not only facilitate access to data, but also threaten its confidentiality and integrity. The rapid and ever-growing challenges of cyber security require the development of effective strategies and innovative approaches to database protection that ensure the reliability and resilience of databases in the face of ever-increasing cyber-attacks and security breaches. The work deals in detail with the organization of effective protection of information stored in databases. The main methods of information protection in databases are analyzed, in particular data encryption, mechanisms of authentication, access control and monitoring of user activity. Their advantages and disadvantages are defined, as well as the possible consequences for data in case of non-compliance with these protection methods. The article highlights the importance of constant monitoring and analysis of user activity for timely detection and response to possible data security threats in the database system. The importance of a comprehensive approach to protection, which takes into account the specifics of a specific organization and allows to ensure an effective level of data security, is emphasized.*

***Keywords:*** database, privacy, integrity, availability, information protection, threat, identity, authentication, authorization, encryption, monitoring, SQL Injection, IDS/IPS, access.

**Гарасимчук Олег Ігорович**, к.т.н., доцент кафедри захисту інформації Національного університету «Львівська політехніка».

**Oleh Harasymchuk**, Ph.D., Associate Professor at the Department of Information Security, National University "Lviv Polytechnic"

**Бужович Оксана Сергіївна**, студентка спеціальності «Кібербезпека та захист інформації» Національного університету «Львівська політехніка».

**Oksana Buzhovych**, student of the "Cybersecurity" specialty of the National University "Lviv Polytechnic".

Отримано *13 березня 2024 року*, затверджено редколегією *1 квітня 2024 року*