МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРИЗОВАНИХ СИСТЕМ ЗАХИСТУ ІНФОРМАЦІЇ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри
_____С.В. Казмірчук
«_____»_____20__р.

На правах рукопису
УДК 004.056:004.657(079.2)

ДИПЛОМНА РОБОТА

ЗДОБУВАЧА ВИЩОЇ ОСВІТИ
ОСВІТНЬОГО СТУПЕНЯ «БАКАЛАВР»

Тема: Програмний модуль захисту конфіденційності даних бази даних підприємства

Виконавець: А.І. Назаренко
Керівник: старший викладач О.В. Дубчак
Нормоконтролер: старший викладач О.В. Дубчак

Київ 2021

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет: Кібербезпеки, комп'ютерної та програмної інженерії

Кафедра: Комп'ютеризованих систем захисту інформації

Освітній ступінь: Бакалавр

Спеціальність: 125 «Кібербезпека»

Освітньо-професійна програма: «Безпека інформаційних і комунікаційних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____С.В. Казмірчук

«__»_____20__р.

ЗАВДАННЯ

на виконання дипломної роботи

здобувача вищої освіти Назаренка Андрія Івановича

1. Тема: *Програмний модуль захисту конфіденційності даних БД підприємства* затверджена наказом ректора від «26» квітня 2021 р. № 652/ст.
2. Термін виконання: з 10.05.2021 р. по 20.06.2021 р.
3. Вихідні дані: база даних підприємства, наявні вразливості баз даних, загрози, існуючі методи автентифікації користувача в базі даних; програмне забезпечення для розробки та тестування модуля автентифікації користувача.
4. Зміст пояснювальної записки: аналіз існуючих способів та методів автентифікації користувача в базі даних; розробка програмного забезпечення, тестування розробленого програмного модуля, верифікація отриманих результатів.

КАЛЕНДАРНИЙ ПЛАН

виконання дипломної роботи

№ п/п	Етапи виконання дипломної роботи	Термін виконанняетапів	Примітка
1.	Уточнення постановки задачі	19.04.2021	*Виконано*
2.	Аналіз літературних джерел	21.04.2021-23.04.2021	*Виконано*
3.	Обґрунтування вибору рішення	25.04.2021	*Виконано*
4.	Збір інформації	26.04.2021-30.04.2021	*Виконано*
5.	Дослідження проблем, вразливості та загроз конфіденційної інформації бази даних	30.04.2021-04.05.2021	*Виконано*
6.	Розробка алгоритму та програмного модуля для забезпечення конфіденційності інформації бази даних	05.05.2021-15.05.2021	*Виконано*
7.	Тестування розробленого програмного модуля	17.05.2021-20.05.2021	*Виконано*
8.	Оформлення презентації	01.06.2021-03.06.2021	*Виконано*
9.	Передзахист в ЕК	04.06.2021	*Виконано*
10.	Перевірка на антиплагіат	04.06.2021-07.06.2021	*Виконано*
11.	Отримання рецензії	10.06.2021-11.06.2021	*Виконано*
12.	Оформлення і друк пояснювальної записки	14.06.2021-15.06.2021	*Виконано*
13.	Підготовка до захисту в ЕК	15.06.2021	*Виконано*

Здобувач вищої освіти А.І. Назаренко

(підпис, дата)

Керівник дипломної роботи О.В. Дубчак

(підпис, дата)

РЕФЕРАТ

Дипломна робота складається зі вступу, двох розділів, загальних висновків, списку використаних джерел, має 52 сторінки основного тексту, 14 рисунків, 1 таблицю. Список використаних джерел містить 21 найменування і займає 3 сторінки. Загальний обсяг роботи 60 сторінок.

Метою роботи є створення прототипного програмного модуля, завданням якого є покращення обробки токенів доступу до системи, в якій обробляється певна кількість інформації.

В роботі вирішені наступні завдання:

- проведено аналіз існуючих загроз конфіденційності інформації, що зберігається в базах даних;
- досліджено наявні способи автентифікації користувачів у базі даних;
- розроблено авторський програмний модуль щодо убезпечення процесу автентифікації користувача;
- вироблено рекомендації щодо використання розробленого програмного забезпечення.

Розроблений метод та програмне забезпечення відносяться до галузі інформаційної безпеки і можуть бути використані для підвищення рівня захищеності бази даних підприємства.

Можливі напрямки розвитку цієї роботи пов’язані із розширенням алгоритму програмного забезпечення відповідно до вимог стандартів для підвищення захисту конфіденційності інформації бази даних підприємства.

Ключові слова: кібербезпека, інформаційна безпека, конфіденційність, автентифікація, верифікація, ідентифікація, база даних.

ЗМІСТ

УМОВНІ ПОЗНАЧЕННЯ ТА СКОРОЧЕННЯ:

JWT	- JSON Web Token	- відкритий стандарт для створення токенів доступу;
RPC	- Remote Procedure Call	- виклик віддалених процедур;
SSL	- Secure Socket Layer	- криптографічний протокол;
SQL	- Structured Query Language	- мова структурованих запитів;
VPN	- Virtual Private Network	- віртуальна приватна мережа;
БД	- база даних;	
ДОБ	- достовірна обчислювальна база;	
ІБ	- інформаційна безпека;	
ІС	- інформаційна система;	
НСД	- несанкціонований доступ;	
ПЗ	- програмне забезпечення;	
СКБД	- система керування базою даних;	

ВСТУП

Актуальність. Проведення дослідження провідних підприємств показало, що на даний момент не існує загальної концепції безпеки для система керування базою даних (СКБД). Історично склалось так, що система безпеки баз даних (БД) розвивалась у якості реагування спеціалістів на дії кіберзлочинців. Як наслідок БД підприємства захищена в одному напрямку, але має уразливість перед тим, з чим ще не зіштовхувались спеціалісти з безпеки підприємства.

Оскільки проникнення до бази даних підприємства формується ще на стадії автентифікації, для забезпечення безпечного входу до системи спеціалісти з питань безпеки схиляються до системи, які використовують механізм по обробці автентифікаційних даних між користувачем та сервером, схиляючись до стандарту Json Web Token (JWT), який перевіряє чи є користувач тим, за кого він себе видає.

У наш час, враховуючи науково-технічний прогрес та стрімкий розвиток інформаційних технологій, постає необхідність у забезпеченні безпеки БД. Однак, при широкому використанню інформації у повсякденному житті та у професійній сфері, постає питання захищеності і безпеки таких технологій автентифікації користувача в системі БД.

Метою дипломної роботи є створення прототипного програмного модуля, завданням якого є покращення обробки токенів доступу до системи, в якій обробляється певна кількість інформації.

Для досягнення поставленої мети потрібно виконати наступні завдання:

- провести аналіз існуючих загроз конфіденційності інформації, що зберігається в базах даних;
- дослідити наявні способи автентифікації користувачів у базі даних;
- розробити авторський програмний модуль щодо убезпечення процесу автентифікації користувача;

– виробити рекомендації щодо використання розробленого програмного забезпечення.

Об’єкт дослідження: процес захисту БД, зокрема, автентифікації користувача в базі даних підприємства, що реалізовано згідно стандарту JWT.

Предмет дослідження: існуючі методи та засоби захисту баз даних, протоколи для реалізації програмних засобів модуля автентифікації в базі даних.

Методи дослідження:

– метод порівняльного аналізу існуючих загроз та наявних способів автентифікації в БД;
– метод об’єктно-орієнтованого програмування при створенні модуля автентифікації.

Галузь застосування: створений програмний модуль може бути використаний для зниження ризику ймовірності несанкціонованого доступу в корпоративних програмних застосунках та в системах з високим навантаженням.

РОЗДІЛ 1. БАЗА ДАНИХ – ОБ'ЄКТ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

На сьогоднішній день для зберігання та зручного користування відомими даними у всіх інформаційних системах використовують автоматизовану систему. Це набагато спрощує завдання пошуку і отримання необхідної інформації, або внесення змін до даних, що підвищує швидкість роботи підприємства.

1.1. Основні поняття терміну база даних

Перед початком визначення поняття автоматизованої системи зберігання та обробки інформації варто відзначити, що такі системи в наш час поширюються швидко та мають різні варіації свого існування. Така автоматизована система називається БД [1]. Кожне підприємство використовує таку базу для спрощення та підвищення ефективності роботи у своїй діяльності.

1.1.1. Визначення терміну база даних, її властивості та значення.

БД – це сукупність взаємопов'язаних даних, організована відповідно до певних правил опису, зберігання та маніпулювання, подана у формі, придатній для автоматичного опрацювання, й призначена задовольняти інформаційні потреби користувачів інформації [1].

Для створення БД використовують спеціальне програмне забезпечення, а для встановлення контролю за ними існує СКБД. БД є основою більшої кількості автоматизованих систем обробки інформації.

БД розрізняють за видами [1]:

– бібліографічна БД, що містить бібліографічну інформацію;

– фактографічна БД, інформаційне наповнення якої складається з конкретних фактів і інформації про них;

– реферативна БД, яка містить бібліографічні записи з рефератами або анотаціями;

– лексикографічна БД, в якій подані дані про лексичні одиниці певної лексикографічної моделі та її структуру;

– повнотекстова БД, що містить повні тексти документів.

БД — це деякий набір даних, які пов'язані між собою спільною ознакою або властивістю, та впорядковані, наприклад, за алфавітом. [1] Об'єднання великої кількості даних в єдину базу дає змогу для формування безлічі варіації групування інформації.

Головною перевагою БД є швидкість внесення та використання потрібної інформації. Завдяки спеціальним алгоритмам, які використовуються для БД, можна легко знаходити необхідну інформацію всього за декілька секунд. Також в БД існує певний взаємозв'язок даних: зміна в одному рядку може спричинити зміни в інших рядках — це допомагає працювати з інформацією простіше і швидше.

БД дають змогу зберігати інформацію, що виглядає як зв'язані між собою за змістом таблиці. Саме в БД зберігаються вся необхідна та корисна інформація для функціонування підприємства (клієнтські дані, прайс-лист, список товарів).

Під час програмування сайтів підприємств використовують різні СКБД. До основних СКБД, відносять:

– об'єктно-реляційна СКБД Oracle Database;

– вільна СКБД PostgreSQL;

– СКБД Microsoft SQL Сервер;

– вільна СКБД MySQL. [2]

Такі СКБД відрізняються централізованою обробкою запитів, забезпечують надійність, доступність та безпеку БД. [2] Найбільш популярною системою управління є MySQL, вона надає користувачу зручний доступ для управління БД та підтримує велику кількість таблиць різних типів.

Прагнення виділити загальну частину інформаційних систем (ІС), відповідальну за керування складноструктурованими даними, було основною причиною створення СКБД, без яких успішне ведення бізнесу і керування підприємствами сьогодні практично неможливе. У більшості випадків, коли

говорять про бази даних, мають на увазі деяке автоматизоване сховище інформації різних типів (числового, символьного, логічного та ін.). Але таке визначення не цілком коректне. [2]

Користувачів ІС умовно можна розділити на дві групи: внутрішні і кінцеві. Внутрішні користувачі розробляють ІС і підтримують її функціонування, кінцеві - ті, задля яких і створюється ІС. Групу внутрішніх користувачів складають:

– адміністратор БД;
– системні програмісти;
– прикладні програмісти. [2]

Функції адміністратора БД на стадії розробки та експлуатації ІС різні і тому виконуються різними особами.

На стадії проектування адміністратор БД виступає як ідеолог і конструктор системи, керує роботами зі створення програмного оточення БД. [2]

На стадії експлуатації адміністратор БД — особа, відповідальна за функціонування ІС. Тобто керує режимом використання даних. Основні задачі адміністратора БД при експлуатації — захист даних від руйнування, забезпечення достовірності даних, аналіз ефективності використання ресурсів ІС. [2]

Системні програмісти виконують генерацію СКБД, стежать за її функціонуванням у середовищі операційної системи, розробляють за завданням адміністратора БД програмні компоненти, що розширюють програмне забезпечення СКБД.

Завдання прикладних програмістів полягає у розробці прикладного програмного забезпечення (ПЗ). Для цього їм необхідне знання алгоритмічних і мовних засобів СКБД. Кінцеві користувачі або спілкуються з ІС в інтерактивному режимі, або формулюють свої запити службі адміністратора БД. [2]

1.1.2. Система керування базами даних.

СКБД — сукупність мовних та програмних засобів, призначених для створення, обслуговування та використання БД багатьма користувачами. СКБД

працюють із даними, поданими у вигляді набору записів. Кожен запис має унікальний ідентифікатор (ключ) та набір полів (атрибутів) із заповненими значеннями. Більшість сучасних СКБД використовують реляційну модель БД. Ця модель ґрунтується на використанні поняття відношення, а БД подана у вигляді взаємопов’язаних таблиць із записами. [3]

Основні функції СКБД [3]:

– переклад схеми, яка визначає структуру даних і записана на мові визначення даних (яка застосовується в цій СКБД), у деяке формалізоване представлення із внутрішніми зв’язками, що використовується системою при подальшій роботі з даними;

– створення БД (завантаження даних у базу);

– реалізація запитів користувачів (сформульованих на спеціальній мові, яка прийнята в цій СКБД) на сортування та відбір даних за заданими критеріями;

– використання деякої частини БД, що може супроводжуватися редагуванням та обробкою інформації;

– оновлення деяких частин БД без зміни загальної структури даних;

– забезпечення захисту даних та пріоритетів у їх використанні. [3]

Серед користувачів БД є особлива категорія довірених осіб, які мають право вносити зміни в БД та регулювати доступ до інформації. Ці особи називаються адміністраторами БД. [3]

1.2. Поняття інформаційної безпеки та її зв’язок з базами даних

У сучасних умовах будь-яка діяльність пов'язана з оперуванням великими об'ємами інформації, яке проводиться широким колом осіб, потребує уваги. Захист даних від несанкціонованого доступу (НСД) є одним з пріоритетних завдань при проектуванні будь-якої ІС. Наслідком збільшеного останнім часом значення інформації у роботі підприємства стали високі вимоги до

конфіденційності даних. СКБД, особливо реляційні СКБД, стали домінуючим інструментом в цій області. Забезпечення інформаційної безпеки(ІБ) СКБД набуває вирішальне значення при обранні конкретного засобу забезпечення необхідного рівня безпеки організації в цілому.

Для СКБД важливо три основні аспекти ІС - конфіденційність, цілісність і доступність. Загальна ідея захисту бази даних полягає в проходженні рекомендаціям, сформульованим для класу безпеки в «Критеріях оцінки надійних комп'ютерних систем». [4]

Політика безпеки визначається адміністратором даних. Проте вирішення захисту даних не повинні бути обмежені тільки рамками СКБД. Абсолютний захист даних практично не реалізовується, тому зазвичай задовольняються відносним захистом інформації - гарантовано захищають її на той період часу, поки НСД до неї буде спричиняти негативні наслідки. Розмежування доступу до даних також описується в БД за допомогою обмежень, і інформація про це зберігається в її системному каталозі. Іноді може бути запит додаткової інформації з операційних систем, в оточенні яких працюють сервер БД і клієнт, що звертається до сервера БД. [4]

ІБ – це стан захищеності інформаційного середовища, захист інформації являє собою діяльність щодо запобігання витоку інформації, що захищається, несанкціонованих і ненавмисних впливів на інформацію, що захищається, тобто процес, спрямований на досягнення цього стану. Метою реалізації ІБ будь-якого об'єкта є побудова системи забезпечення ІБ даного об'єкту. [3]

Для побудови та ефективної експлуатації БД необхідно:

– виявити вимоги захисту інформації, специфічні для даного об'єкта захисту;

– врахувати вимоги національного та міжнародного законодавства;

– використовувати напрацьовані практики (стандарти, методології);

– визначити підрозділи, відповідальні за реалізацію та підтримку БД;

– розподілити між підрозділами області відповідальності у здійсненні вимог БД;

– на базі управління ризиками ІБ визначити загальні положення, технічні та організаційні вимоги, складові політики ІБ об'єкта захисту;

– реалізувати вимоги політики ІБ, впровадивши відповідні програмно-технічні засоби і способи захисту інформації в БД;

– реалізувати систему управління ІБ БД підприємства;

– використовуючи систему управління організувати регулярний контроль ефективності БД. [3]

Захист інформації – комплекс заходів, спрямованих на забезпечення найважливіших аспектів ІБ. На сьогоднішній день існують такі аспекти захисту інформації, як [4]:

– цілісність (захист інформації від несанкціонованої модифікації);

– конфіденційність (захист від несанкціонованого ознайомлення з інформацією);

– доступність (забезпечення доступу до інформації, а також можливості її використання).

Відповідно, до даних аспектів захисту інформації, виділяють наступні загрози інформації:

– загрози цілісності (знищення та модифікація інформації);

– загрози доступності (блокування доступу до інформації);

– загрози конфіденційності (НСД, витік та розголошення інформації). [4]

Система називається безпечною, якщо вона, використовуючи відповідні апаратні і програмні засоби, керує доступом до інформації так, що тільки належним чином автентифіковані особи або ж діють від їхнього імені процеси отримують право читати, писати, створювати і видаляти інформацію. [5]

Очевидно, що абсолютно безпечних систем не існує, і тут мова йде про надійну систему у сенсі «система, якій можна довіряти». Система вважається надійною, якщо вона з використанням достатніх апаратних і програмних засобів забезпечує одночасну обробку інформації різного ступеня секретності групою

користувачів без порушення прав доступу. Основними критеріями оцінки надійності є:

- політика безпеки;
- гарантованість. [6]

Політика безпеки, будучи активним компонентом захисту (включає в себе аналіз можливих загроз і вибір відповідних заходів протидії), відображає той набір законів, правил і норм поведінки, яким користується конкретна організація при обробці, захисту та поширенні інформації. Вибір конкретних механізмів забезпечення безпеки системи здійснюється відповідно до сформульованої політики безпеки. [6]

Гарантованість, будучи пасивним елементом захисту, відображає міру довіри, яка може бути надана архітектурі та реалізації системи (іншими словами, відображає, наскільки чітко визначено та обрано механізми, що забезпечують безпеку системи). [6]

У надійній системі повинні реєструватися всі події, що відбуваються і стосуються безпеки (повинен використовуватися механізм підзвітності протоколювання, що доповнює аналізом заповненої інформації, тобто аудитом). [6]

При оцінці ступеня гарантованості, з якою систему можна вважати надійною, центральне місце займає достовірна обчислювальна база. Достовірна обчислювальна база (ДОБ) являє собою повну сукупність захисних механізмів комп'ютерної системи, яка використовується для втілення в життя відповідної політики безпеки. Надійність ДОБ залежить виключно від її реалізації та коректності введених даних (наприклад, даних про благонадійність користувачів, які визначаються адміністрацією). Кордон ДОБ утворює периметр безпеки. Компоненти ДОБ, що знаходяться всередині цього кордону, повинні бути надійними (отже, для оцінки надійності комп'ютерної системи досить розглянути тільки її ДОБ). Від компонентів, що знаходяться поза периметром безпеки не потрібно надійності. Однак це не повинно впливати на безпеку системи. [4]

Так як зараз широко застосовуються розподілені системи обробки даних, то під «периметром безпеки» розуміється межа володінь певного підприємства, у підпорядкуванні якої знаходиться ця система. Контроль допустимості виконання суб'єктами певних операцій над об'єктами, тобто функції моніторингу, виконується ДОБ. При кожному зверненні користувача до програм або даних, монітор перевіряє допустимість даного звернення (узгодженість дії конкретного користувача зі списком дозволених для нього дій). Реалізація монітора звернень називається ядром безпеки, на базі якої будуються всі захисні механізми системи. Ядро безпеки має гарантувати власну незмінність. [4]

Використання БД підприємствами та організаціями є оптимальним методом для роботи з достатньо великими обсягами інформації.

Опираючись на дані зі статистики (рис. 1.1.) в останні роки кількість витоків даних дуже велика та неухильно зростає. Статистика 2020 року показує, що за цей період було зламано більше записів, ніж за минулі 15 років. Якщо станом на 2018 рік кількість зламів досягала 55 мільйонам, то вже у 2021 році таких зламів вже більше ніж 101 мільйон. На число зломів сильно вплинули спеціальні віруси-вимагачі, у яких основною метою були лікарні та інші організації охорони здоров’я. Через такі атаки великим і дрібним компаніям довелося призупини свою діяльність. Ті, хто змогли дати відсіч атакам, екстрено впроваджували нові стратегії щодо захисту даних. Але, як показує практика, швидкі рішення в сфері кіберзахисту можуть привести до виникнення нових уразливостей в системі. А значить, кіберзлодії все так само можуть успішно атакувати мережі компаній, як показано на рис. 1.1. – частота зламів зростає. [7]

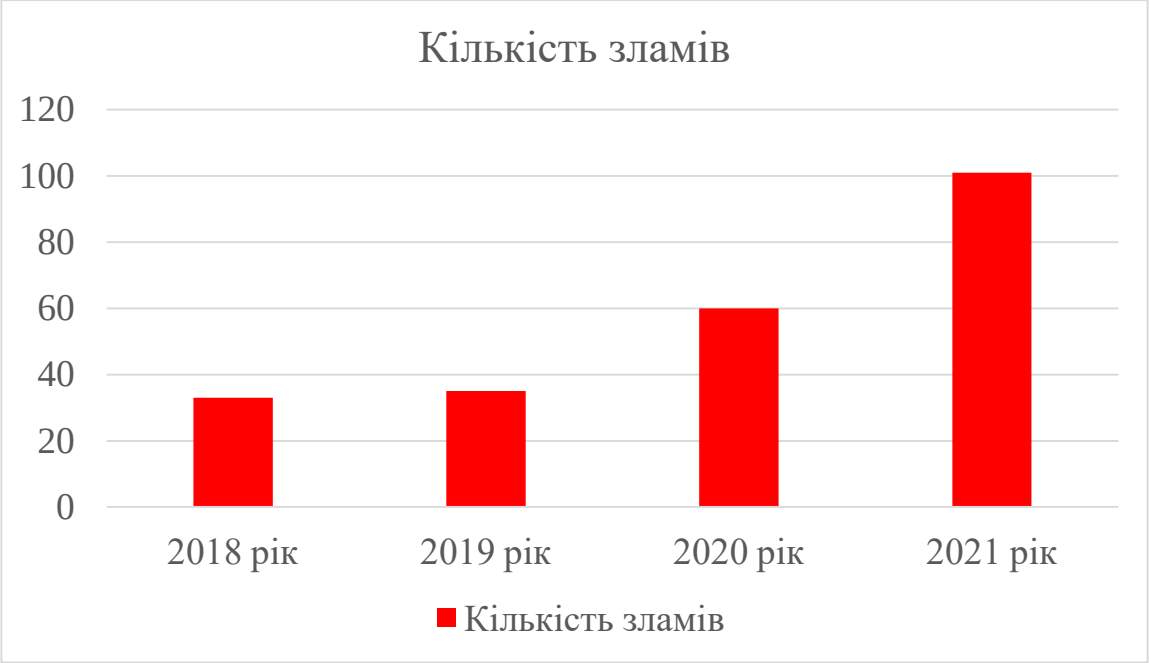

Рис. 1.1. Діаграма випадків зламу баз даних [7]

Провівши аналіз частоти випадків злому БД можна стверджувати, що захист даних від НСД є одною з пріоритетних задач при проектуванні будь-якої інформаційної системи. Для вирішення цих задач необхідно провести аналіз загроз БД та методів і засобів для їх запобігання. Руйнування комп'ютерних сховищ корпоративних даних або втрата, просто тимчасова недоступність цих даних може стати причиною справжньої катастрофи для підприємства. [3]

БД являє собою найважливіший корпоративний ресурс, який має бути належним чином захищений за допомогою всіх відомих засобів і методів захисту. Для швидкого виявлення об’єктів, що підвищують ступінь ризику, тобто потенційну можливість втрати або пошкодження даних необхідно заздалегідь знати та визначити відповідні загрози. [5]

Відомі наступні потенційні загрози БД:

– викрадення і фальсифікація даних;
– втрата конфіденційності (порушення таємниці);
– порушення недоторканності особистих даних;
– втрата цілісності;

– втрата доступності. [5]

Усі ці загрози мають певні зв’язки між собою, тобто, порушення захищеності системи в одному напрямку часто мають вплив на захищеність системи у інших напрямках.

Викрадення і фальсифікація даних можуть відбуватися не тільки в середовищі БД. Уся інформаційна система тієї чи іншої організації може бути під загрозою. Тому, будь-яка інформаційна система потребує постійного спостереження за всіма можливими каналами витоку конфіденційної інформації, якими потенційно володіє. [3]

1.2.1. Методи та засоби забезпечення захисту інформації в базах даних.

Не торкаючись загальних питань інформаційної безпеки, а також загальних методів і засобів захисту інформації, розглянемо методи і засоби захисту даних, реалізовані за допомогою можливостей, реалізованих в СКБД, з урахуванням особливостей схеми БД з універсальною моделлю даних.

Засоби і методи захисту БД відрізняються один від одного в залежності від систем керування базами даних, але в тій чи іншій мірі досить часто зустрічаються наступні:

– автентифікація користувачів;
– застосування представлень;
– резервне копіювання та відновлення;
– підтримка цілісності;
– шифрування;
– застосування стійких від поломок апаратних засобів. [3]

Для подолання проблем забезпечення ІБ СКБД необхідно перейти від методу закриття вразливих елементів БД до комплексного підходу забезпечення безпеки сховищ інформації. Основними етапами цього переходу повинні стати наступні положення:

1. Розробка комплексних методів забезпечення безпеки БД на поточному етапі. Створення комплексних методів дозволить застосовувати їх (або їх відповідні версії) при розробці БД і користувальницького ПЗ. [6]

2. Оцінка і класифікація загроз та вразливостей СКБД. Спеціалізована класифікація загроз і вразливостей СКБД дозволить упорядкувати їх для подальшого аналізу і захисту, надає можливість встановити залежність між вразливостями і причинами (джерелами) їх виникнення. В результаті при введенні конкретного механізму в СКБД з'явиться можливість встановити і спрогнозувати пов'язані з ним загрози і заздалегідь підготувати відповідні засоби забезпечення безпеки. [6]

3. Розробка стандартних (застосовуються до різних СКБД без внесення змін або з мінімальними змінами) механізмів забезпечення безпеки. Стандартизація підходів і мов роботи з даними дозволить створити багатоплатформені засоби забезпечення безпеки, які застосовуються до різних СКБД. З одного боку, це методичні та теоретичні підходи, що застосовуються в рамках моделі даних. На сьогоднішній день є напрацювання таких механізмів для реляційної моделі, однак вони не вирішують всіх нагальних питань безпеки. З іншого - це розробка теоретичного базису для нових СКБД, зокрема, конкретизація і формалізація агрегатних моделей даних. Поява готових програмних засобів багато в чому залежить від виробників і розробників СКБД і їх слідування стандартам, а також достатності визначених у стандарті засобів для побудови розвинених механізмів безпеки. [6]

4. Розробка теоретичної бази інформаційного захисту систем зберігання і маніпулювання даними. [6]

В Україні найбільше від кібератак страждають БД впливових медіа, фінансових інститутів та державних установ. При цьому на сьогодні зростає не тільки кількість атак на інформаційну інфраструктуру, але і їх складність. Для того щоб швидко та вчасно вживати заходи, що знижують ступінь ризику, тобто потенційну можливість втрати або пошкодження даних необхідно вивчати

відповідні загрози, аналізувати відповідність методів протидії та підходити комплексно до захисту БД. [6]

Дані в системах БД мають зберігатися з гарантуванням конфіденційності та безпеки. Інформація не може бути загубленою або викраденою. Під безпекою даних у базі розуміють захист даних від випадкового або спланованого доступу до них осіб, які не мають на це права, від несанкціонованого розкриття, зміни або видалення.

Безпека даних підтримується комплексом заходів і засобів:

– організаційно-методичні заходи передбачають розроблення інструкцій та правил, які регламентують доступ до даних та їхнє використання, а також створення відповідних служб і підрозділів, які стежать за дотриманням цих правил;

– правові та юридичні заходи передбачають юридичне закріплення прав і обов'язків щодо зберігання, використання й передавання в електронному вигляді даних, які підлягають захисту, на рівні державних законів та інших нормативних документів;

– технічні засоби захисту — це комплекс технічних засобів, які сприяють вирішенню проблеми захисту даних;

– програмні засоби захисту — це комплекс математичних, алгоритмічних і програмних засобів, що сприяють вирішенню проблеми захисту даних. [5]

Для будь-якої захищеної БД обов’язковими є процедури ідентифікації, автентифікації та авторизації. Сутність процедури ідентифікації полягає в призначенні користувачу БД – імені. Ім’я користувача – це деяка унікальна мітка, що відповідає прийнятим угодам і забезпечує однозначну ідентифікацію об’єкта реального світу в просторі об’єктів, що відображаються. Сутність процедури автентифікації полягає в підтвердженні автентичності користувача, що представив ідентифікатор. У ряді сучасних СКБД використовується такі способі автентифікації (рис .1.2.):

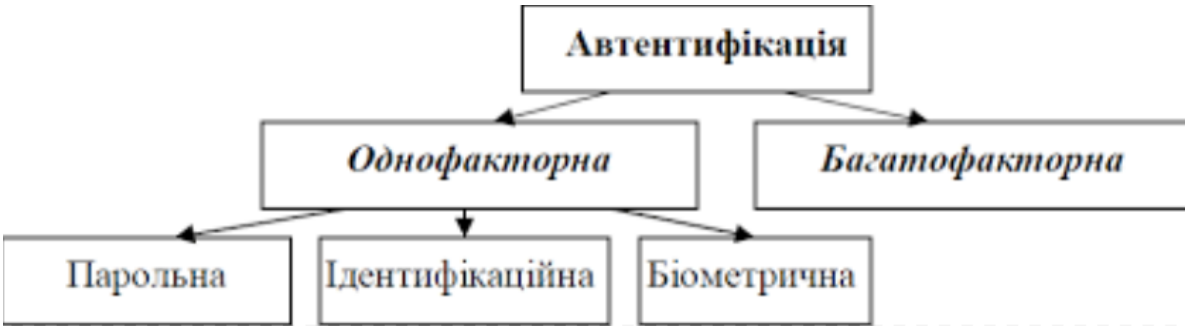

Рис. 1.2. Види існуючих автентифікацій користувача в БД. [5]

– **біометрична автентифікація** – це процес доведення і перевірки автентичності заявленого користувачем імені через пред'явлення користувачем своїх біометричних характеристик (наприклад, відбитки пальців і долоні, звуки голосу, обличчя, відбиток сітківки ока, особливості роботи на клавіатурі, електронний цифровий підпис); [5]

– **парольна автентифікація** – це процес доведення і перевірки автентичності заявленого користувачем імені шляхом введення ним пароля або парольної фрази. Парольні фрази забезпечують більшу безпеку, ніж короткі паролі, але вимагають більшого часу для введення. Заходами, що дають змогу підвищити надійність парольного захисту є: накладання технічних обмежень, управління терміном дії паролів, обмеження доступу до файлу паролів, обмеження кількості невдалих спроб входу в систему, використання програмних генераторів паролі; [5]

– **автентифікація із застосуванням токенів**. Токен – це предмет або пристрій, володіння яким підтверджує автентичність користувача. [5]

Ефективність процедур ідентифікації та автентифікації істотним чином впливає на ефективність системи безпеки в цілому. В процесі автентифікації встановлюється набір можливих операцій з даними, які може здійснювати користувач.

1.2.2. Модель безпеки баз даних.

Управління доступом в БД включає такі питання, як доступ до таблиць і її полів. Для організації цього доступу використовуються моделі безпеки, які включають дискреційну, мандатну і рольову моделі. Способом формалізованого представлення дискреційного доступу є матриця доступу або списки управління доступом, що встановлюють перелік користувачів і перелік дозволених операцій відносно кожного об'єкта БД. Можливі декілька підходів до побудови дискреційного управління доступом: децентралізована, централізована та змішана моделі безпеки. [6] Саме змішаний варіант реалізований у більшості СКБД.

Мандатна модель поєднує захист і обмеження прав, що використовуються відносно комп'ютерних процесів, даних і системних пристроїв, та призначена для запобігання їх небажаному використовуванню. Для СКБД мандатна модель безпеки може розширювати або замінювати дискреційний контроль доступу і концепцію користувачів і груп. Права доступу кожного суб'єкта і характеристики конфіденційності кожного об'єкта відображаються у вигляді сукупності рівня конфіденційності і набору категорій конфіденційності. Для реалізації безпеки за допомогою мандатної моделі рядкам і стовпцям таблиці БД приписуються мітки, які потім надаються користувачам. Ефективно застосовуватися мандатна модель може тільки разом з дискреційною. [6]

Рольова модель – розвиток політики виборного управління доступом, при цьому права доступу суб'єктів системи на об'єкти групуються з урахуванням специфіки їх використання, утворюючи ролі. Управління правами доступу здійснюється як на основі матриці доступу, так і на основі правил, що регламентують поведінку (ролі) користувача та їх активацію під час сеансів. Рольове розмежування доступу дозволяє реалізувати гнучкі, динамічні правила розмежування доступу. Безпека в цій моделі забезпечується чіткими визначеннями ролей адміністратора БД і користувача БД на права доступу до об'єктів БД і прав на читання, модифікацію, запис і видалення об’єктів.

Технологія управління доступом на основі ролей є достатньо гнучкою і потужною, щоб змоделювати як виборне, так і мандатне управління доступом. [6]

Система безпеки є невід'ємною частиною правильно спроектованої інформаційної системи з базою даних. У загальному вигляді систему безпеки можна представити у вигляді моделі, яку складають шість відносно незалежних рівнів: фізична безпека, безпека мережевого доступу, доменна безпека, безпека локального комп'ютера, безпека сервера баз даних, безпека програмних додатків. Розглянемо детально кожен з цих рівнів. [8]

Фізична безпека. Перший рівень моделі безпеки інформаційної системи з базою даних представлений фізичною безпекою. Фізична безпека забезпечує захист доступу до інфраструктури, яку утворюють внутрішні мережеві компоненти та апаратне обладнання, що підтримує роботу серверних компонент системи. [8]

Фізична безпека має на меті захист життєво важливого серверного та мережевого обладнання шляхом розміщення їх у приміщеннях з обмеженим доступом. Для захисту від стихійного лиха необхідно ретельно планувати і регулярно створювати резервні копії БД, які слід зберігати окремо від місця експлуатації сервера. [8]

Безпека мережевого доступу. На другому рівні – безпека мережевого доступу. Вона включає такі компоненти, як шифрування пакетів та ізоляція транспортного протоколу. Шифрування пакетів реалізоване на проміжному рівні між клієнтом і сервером БД за допомогою протоколу SSL (Secure Socket Layer) або шифрування при RPC (виклик віддалених процедур). При обміні даними між комп'ютерами, під управлінням операційної системи Windows версій, випущених після 2000 року, шифрування пакетів реалізується за допомогою IP безпеки (IP Security, IPSec). IPSec підтримується також багатьма сучасними маршрутизаторами. Інша дієва методика забезпечення безпеки мережевого доступу полягає у застосуванні брандмауера або спеціалізованого обладнання для розмежування комп'ютерних мереж. Найпростіше рішення розмежування

мережевого доступу – використання різних протоколів у різних сегментах комп'ютерної мережі. Наприклад, у локальній мережі можна використовувати стек протоколів IPX/SPX, а в зовнішній (Інтернет) – TCP/IP. У сучасних реалізаціях локальних обчислювальних мереж потребу в розділенні доступу між сегментами мережі вирішують застосовуючи технологію віртуальних приватних мереж. [8]

Доменна безпека. Третій рівень забезпечує доменну безпеку. Вона реалізується за допомогою служб каталогу облікових записів області мережевих імен. Наприклад, у мережах Microsoft цю роль виконують служби Active Directory. Якщо комп'ютер, на якому працює сервер баз даних, є членом мережевого домена, можна наділити обліковий запис користувача або групи користувачів привілеями для доступу до сервера БД і виконання на ньому певних дій. Служби обліку доменних записів користувачів комп'ютерної мережі забезпечують надійну перевірку прав користувачів на рівні доступу до мережевих компонентів. Доменні паролі шифруються з метою уникнення їх перехоплення під час передачі по мережі. Зазвичай, вони чутливі до регістра символів, що ускладнює їх підбір. Крім того, широко застосовуються такі прийоми, як визначення мінімальної довжини пароля та терміну його дії. [8]

Безпека локального комп'ютера. На четвертому рівні забезпечується безпека локального комп'ютера. Вона зумовлює проведення аудиту засобами операційної системи, верифікацію прав доступу до файлів та реєстру, а також функціонування служб шифрування. Сервери сучасних баз даних, зазвичай, інсталюються на комп'ютерах, які працюють під управлінням операційних систем сімейств Windows або Unix. Ці операційні системи підтримують аудит системи безпеки, дозволяючи відстежувати такі події, як вхід користувачів у систему, спроби звернення до файлових об'єктів і, у тому числі, до файлів баз даних. [8]

Безпека сервера БД. П'ятий рівень забезпечує систему безпеки сервера баз даних, яка включає чотири категорії безпеки: автентифікацію, авторизацію, шифрування і служби аудиту. Процес надання доступу до БД складається з двох

фаз: спочатку виконується підключення до сервера баз даних, а потім відкривається доступ до БД з усіма її об’єктами (процес автентифікації). Дозволи на роботу з об’єктами дозволяють або забороняють користувачу виконувати дії над об’єктами БД, наприклад, таблицями та представленнями. Дозволи на виконання SQL-операторів дозволяють або забороняють користувачам створювати об’єкти БД, переглядати та маніпулювати даними, робити їх копії. [8]

Для зменшення кількості адміністративних операцій, які необхідно виконати для надання або позбавлення дозволів користувачам БД, сучасні СКБД підтримують групування користувачів за ролями. Ролі схожі на групи операційних систем, але створюються і супроводжуються в рамках сервера БД. Права доступу можна надати ролі, так само, як і окремому користувачу. Якщо права доступу призначаються ролі, то кожен користувач, включений у цю роль, набуває її права доступу. Спадкоємство – це здатність об’єкта (у даному випадку користувача) приймати всі властивості іншого об’єкта (у цьому випадку ролі). Ролі мають властивість вкладеності одна в одну, що дозволяє будувати ієрархію ролей. [8]

Існує два типи ролей: стандартні та прикладні. Стандартні ролі від початку присутні в сервері БД та наділені певними правами, які можуть успадковуватися користувачами, що одержали в них членство. Прикладні ролі створюються розробниками інформаційних систем з базами даних з метою вирішення пов’язаних з ними задач безпеки. Усі дії, що виконуються в базі даних, відстежуються за допомогою аудиту СКБД. Деякі об’єкти БД, наприклад, збережені процедури, дозволяється зашифровувати, щоб захистити їхній вміст від несанкціонованого читання. [8]

Безпека програмних додатків. Шостий рівень організовує безпеку програмних додатків. Додаток може розширювати можливості системи безпеки баз даних, доповнюючи її власними функціями безпеки. Програма, що звертається до БД, викликає спеціальну системну збережену процедуру з метою активізації ролі програмних додатків. Крім того, у додатку іноді реалізована

власна система безпеки, непідконтрольна СКБД. Для ізоляції додатків від деталей механізму доступу до даних застосовуються функції API доступу до даних. Згідно з моделлю безпеки БД, відповідальність за кожен її рівень розподіляється між визначеними посадовими особами. За реалізацію перших чотирьох рівнів системи безпеки БД відповідають мережеві та системні адміністратори, адміністратори та розробники БД відповідають за п’ятий рівень, а розробники додатків – за шостий. Фахівці з ІБ, як правило, спостерігають за проектуванням системи безпеки в загальному обсязі. [8]

Реалізація ефективної системи безпеки БД потребує її ретельного планування, при якому основна увага приділяється системі безпеки сервера БД, що займає п’ятий рівень моделі безпеки. Цей процес полягає у визначенні користувачів БД, доступних їм об’єктів та дій, які їм дозволено виконувати в базі даних. Потрібна інформація може бути сформована на основі вимог до інформаційної системи, основу якої утворює БД.

Після складання списку об’єктів БД, що потребують захисту та дій, що доступні не всім користувачам, слід створити список користувачів, які повинні мати доступ до БД. На основі цієї інформації готують список відповідностей «дія – користувач» у вигляді таблиці з перехресними посиланнями на елементи списків окремих дій і користувачів. Цей список визначає, яким користувачам доступні захищені об’єкти, а також, які захищені дії користувачі можуть виконувати в базі даних.

Існують загальні правила реалізації системи безпеки БД, які визначають, кому необхідно призначати дозволи: ролям або окремим користувачам. Вибір залежить від рівня персоніфікації цих дозволів. Наприклад, якщо одному користувачу потрібен унікальний набір дозволів на доступ та маніпулювання даними, слід призначити відповідні дозволи окремому обліковому імені. Проте таких дій, по можливості, слід уникати. Користувачів БД краще включати у визначені ролі, а не призначати їм окремі дозволи.

Для призначення дозволів на роботу з об’єктами і виконання SQL операторів краще використовувати стандартні ролі рівня сервера баз даних,

уникаючи безпосереднього призначення дозволів прикладним ролям і окремим користувачам. У системних ролей є певні особливості, які не можна змінити. Тому слід бути обережними, приписуючи користувачів до системних ролей.

У разі необхідності виділення прикладних ролей і окремих користувачів спочатку слід призначати дозволи ролям і тільки потім окремим користувачам. Нагорі ієрархії системи безпеки потрібно розташовувати мінімальні дозволи, які можна застосовувати до всіх користувачів, а нижче – розширені, для обмеженої кількості користувачів. [8]

Гарною практикою забезпечення контрольованого доступу до даних є створення представлень та збережених процедур, що використовують ланцюжки володіння. У випадку, коли право на використання об’єктів, що звертаються послідовно один до одного, належить одному користувачу і розташовується в одній базі даних, говорять про створення ланцюжка володіння. Тобто, якщо у користувача БД є дозвіл на роботу з першим об’єктом ланцюжка володіння (збереженою процедурою або представленням), то по ланцюжку він зможе діставати доступ до даних і виконувати потрібні дії.

1.2.3. Основні уразливості та типи атак на бази даних.

Безпека інформації у БД комп’ютерних систем досягається шляхом збереження конфіденційності, цілісності та надійності отриманих даних, а також доступності систем інформації та знань, яких вона містить. [9]

Розвиток інформаційних технологій сприяє вдосконаленню корисних інформаційних ресурсів та дає поштовх для розвитку різноманітних шкідливих програмних засобів для нанесення цілеспрямованих пошкоджень та атак, що спричиняють неабиякий збиток.

Згідно з [9], найпоширенішими типами атак та загроз для корпоративних інформаційних систем (рис. 1.3.) є:

1. аналізатори трафіку (sniffer) – перехват та прослуховування мережевого трафіку;
2. повний перебір (brute force) – метод грубої сили;
3. SQL-ін’єкції;

4. DoS-атаки (Distributed Denial of Service – “відмова від обслуговування”);
5. оновлення програмного забезпечення без перевірки підпису або з не авторитетних джерел. [9]

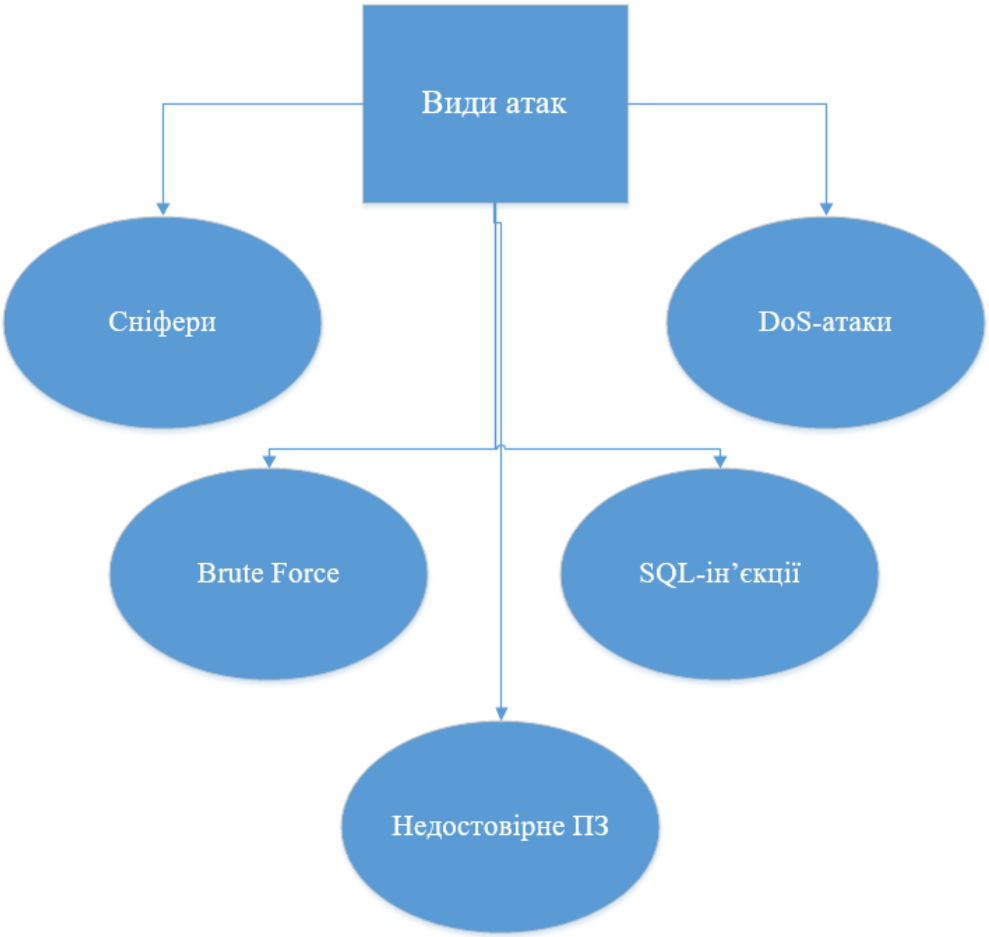

Рис. 1.3. Види атак на корпоративні ІС

Аналізатори трафіку – це програмні або апаратні засоби перехоплення та аналізу мережевого трафіку. Їх використовують шахраї для втручання до протоколу передавання даних між контрагентами задля викривлення або підміни вихідної інформації. Захистом корпоративних джерел інформації від такого виду атак є використання шифрованих SSL-з’єднань (Secure Sockets Layer). Цю опцію захисту, як правило, реалізовано в усіх найпопулярніших СКБД, наприклад, у MySQL. [10]

За її відсутності необхідно організовувати окреме шифроване підключення між серверами з використанням додаткових засобів, що реалізують технологію віртуальної приватної мережі OpenVPN (Virtual Private Network). Вона дає змогу встановлювати захищене з'єднання між комп'ютерами без необхідної зміни їх налаштувань. Для організації ефективного захисту корпоративних баз та сховищ даних доцільно приховувати їхні порти, а самі джерела інформації розміщувати у захищеному сегменті мережі. [11]

За потреби підключення до корпоративної інформації інших серверів вони повинні мати адресу із захищеного сегменту мережі. Підтвердженням доцільності саме такого підходу є здійснена у 2017 році масштабна атака за назвою WannaCry.

Хакери діяли через комп'ютери з відкритим TCP-портом 445 та використовували класичну схему вимагання грошових коштів за дешифрування інформації.

У разі перебування клієнта поза захищеним сегментом його необхідно підключити через захищений VPN, а аутентифікація має відбуватись з використанням сертифікатної аутентифікації (сертифікат при цьому може бути «зашитий» у захищеному сховищі – такому, як eToken). [11]

Повний перебір скерований на підбирання логінів та паролів перебором усіх варіантів та їх модифікацій. Особливістю такого виду атаки є можливість застосування проти шифру будь-якої складності, але вона може вимагати нереалістичних часових та ресурсних витрат.

Як і у випадку попереднього виду атаки, найефективнішим засобом є приховування портів баз даних та використання шифрованих з'єднань, як описано вище, а також використання складних паролів, при формуванні яких задіяно всі типи допустимих символів, букви верхніх та нижніх регістрів, спеціальні символи.

SQL-ін'єкції є найпоширенішим видом атак на SQL БД. Ін'єкції шкідливого коду призводять до втрати маніпуляції операціями, що здійснюються в БДкінцевими користувачами додатка, тобто відбуваються

анулювання транзакцій, зміни та знищення вихідних даних, розкриття конфіденційної корпоративної інформації. [9]

DoS-атака призначена на цілenаправлене надсилання запитів до серверу у такій кількості, щоб він не зміг їх обробити. В результаті такої атаки сервер починає працювати повільно, обробляючи надмірний обсяг помилкових запитів, а потім може вийти з ладу. [9]

Оновлення програмних засобів без перевірки підпису або з неперевірених джерел є видом атаки, за якого здійснюється самостійне завантаження шкідливих оновлень. Основним захистом від такого виду атак є кваліфіковані адміністратори БД, але і вони не можуть забезпечити повної безпеки – підтвердженням є відома атака вірусу Petya.A. [9]

1.2.4. Проблема захисту баз даних.

Проаналізувавши засоби забезпечення безпеки даних, реалізовані у СКБД, архітектуру сховищ та БД, інтерфейси систем, відомі вразливості та інциденти безпеки, було виділено основні проблеми захисту сховищ та БД:

– на належному рівні проблемами захисту інформації займаються тільки провідні фірми виробники промислових, великих СКБД;

– при створенні програмних продуктів розробники намагаються використовувати лише стандартні засоби захисту, що надаються СКБД;

– різновид масштабів та виду інформації, що зберігається, потребує різних підходів до безпеки;

– майже кожна СКБД використовує різні лінгвістичні конструкції для доступу до даних, що організовані на основі однієї моделі. [9]

Актуальність захисту повністю пов’язана із розвитком IT-технологій, що зумовлює зростання можливостей комп’ютерної техніки. Розвиток засобів, методів і форм автоматизації процесів обробки інформації, а також масове застосування персональних комп’ютерів роблять інформацію набагато вразливішою.

Як відомо з [12], основними чинниками вразливості інформації є:

- збільшення обсягів та видів інформації, що обробляється за допомогою комп’ютерів, її збереження в електронному вигляді;
- накопичення інформації у сховищах та базах даних різнорідного призначення;
- розширення кола користувачів, що мають безпосередній доступ до ресурсів обчислювальної системи та масивів даних;
- ускладнення режимів роботи технічних засобів обчислювальних систем; обмін інформацією в локальних та глобальних мережах. [12]

Як подано в [12], основними загрозами корпоративного рівня є:

- використання прав доступу іншої особи;
- несанкціонована зміна або копіювання даних;
- зміна (підміна) програмного забезпечення;
- непродумані методики і процедури, що допускають змішування конфіденційних і звичайних даних в одному документі чи місці зберігання;
- підключення до кабельних мереж без вживання заходів захисту;
- введення кіберзлочинцями некоректних даних;
- шантаж;
- створення “лазівок” у системі;
- викрадення інформації, програмного забезпечення та обладнання;
- відмова систем захисту;
- недостатній рівень знань та недотримання правил безпеки персоналом;
- надання доступу до засекречених даних третім особам;
- електронні перешкоди і радіація;
- руйнування даних у результаті відключення або перенапруження в мережі електроживлення;
- пожежі, повені, диверсії;
- фізичне пошкодження обладнання та елементів інфраструктури; зараження комп’ютерними вірусами. [12]

Загрози бувають комбінованими та призводять до таких наслідків:

- викрадення і фальсифікація даних;
- втрата конфіденційності;
- порушення недоторканності особистих даних;
- втрата цілісності і доступності даних, що загрожує підприємству фінансовими збитками. [12]

Отже, враховуючи перелічені раніше вразливості актуальним є завдання убезпечення конфіденційності БД підприємства від зазначених у [12] загроз.

1.3. Правовий захист систем баз даних

У сучасному суспільстві інформація стає найбільшою цінністю, а індустрія отримання, обробки і трансляції інформації – провідною галуззю діяльності, капіталовкладення в яку щороку зростають.

Ст. 17 Конституції України визначає, що захист суверенітету і територіальної цілісності України, забезпечення її економічної та інформаційної безпеки є найважливішими функціями держави, справою всього українського народу. [13] Таким чином держава дає зрозуміти, що її інформаційне поле має бути захищене і це завдання є одним із найголовніших для держави, оскільки інформація – це зброя будь-якої спільноти.

Оскільки інформація може використовуватись як найефективніший засіб забезпечення національних інтересів і розв’язання суперечностей у різних сферах державної та суспільної діяльності, що породжує специфічну боротьбу засобами інформаційного впливу. Загроза впливу на інформаційну інфраструктуру іншої країни, на її інформаційні ресурси, суспільство, свідомість, підсвідомість особистості з метою нав’язати бажану для країни-ініціатора впливу систему цінностей, поглядів, інтересів і рішень у життєво важливих сферах суспільної й державної діяльності, керувати їхньою поведінкою і розвитком несе головну інформаційну загрозу національній безпеці.

Зростаючий попит на інформацію приводить до виокремлення інформаційної діяльності, у результаті якої виникають нематеріальні об’єкти правовідносин, в окремий вид. Зазначене актуалізує проблему наукових досліджень стосовно охорони та захисту інформаційних ресурсів у базах даних, зокрема в аспекті права власності на інформацію.

Ще задовго до появи інформаційно-комп’ютерних технологій та мереж були присутні дискусії відносно інтелектуальної власності, яка включає в себе право на об’єкти промислової власності та об’єкти авторського і суміжних прав.

Насамперед термін «інтелектуальна власність» був запроваджений для задоволення економічних потреб. Вживання цього поняття правомірне, якщо поставитися до нього як до умовної категорії, що має економічний сенс, предметом правового регулювання якої є дії щодо створення та використання нематеріальних об’єктів – нових знань, які одержують юридичну оболонку охороноздатності. [13]

Закон України «Про основні засади забезпечення кібербезпеки України» від 2017 року - поточна версія законопроекту вводить важливі базові поняття в галузі кіберзахисту та кібербезпеки і визначає права й обов'язки державних органів щодо кібербезпеки. Багато положень у тексті закону мають декларативний характер, він перевантажений положеннями, в яких йдеться про наміри, принципи, що є непритаманним для закону. [14]

Закон України «Про Національну програму інформатизації» дає визначення баз даних, як об’єкта інформаційно-правових відносин, а саме: БД – іменована сукупність даних, що відображає стан об’єктів та їх відношень у визначеній предметній галузі. Цей Закон визначає правові та організаційні основи забезпечення захисту життєво важливих інтересів людини і громадянина, суспільства та держави, національних інтересів України у кіберпросторі, основні цілі, напрями та принципи державної політики у сфері кібербезпеки, повноваження державних органів, підприємств, установ, організацій, осіб та громадян у цій сфері, основні засади координації їхньої діяльності із забезпечення кібербезпеки. [14]

Ця дефініція має доволі загальний характер, однак аналіз інформаційного законодавства дає змогу визначити характерні особливості БД:

– БД – це складні інформаційні об’єкти, які виникають у процесі цілеспрямованої інформаційної діяльності різноманітних суб’єктів;

– у БД, за певними правилами, систематизуються відомості про різні явища, які об’єднуються за принципом однорідності предметної сфери;

– БД – це сукупність інформації у цифровій формі, оскільки поняття «дані» в інформаційному законодавстві трактуються, як інформація у формі, придатній для автоматизованої обробки її засобами обчислювальної техніки;

– інформація в базах даних відзначається впорядкованістю, повнотою, достовірністю, релевантністю;

– цінність БД зберігається за умови постійної актуалізації інформації, яку вони містять;

– правовідносини, об’єктом яких є БД, відрізняються особливим суб’єктним складом;

– БД мають багато спільного з іншими складними інформаційними об’єктами – інформаційними системами, банками даних тощо. [15]

Використання інформації, що міститься в базах даних, дає змогу швидко вирішити ту чи іншу проблему у сфері управлінської, наукової, підприємницької чи іншої діяльності, що обумовлює потребу в їх надійному правовому захисті. Щодо проблеми правового захисту інформаційних ресурсів І. Бачіло зазначає [16], що право поки що не володіє іншими інститутами, крім інституту речової власності та інституту власності інтелектуальної. Одночасно у світовій практиці для захисту БД використовують інститути контрактного права, комерційної таємниці, права недобросовісної конкуренції і безпідставного збагачення.

1.4. Висновки до розділу 1

Проведений аналіз вразливостей та загроз БД показали, що в захисті сховищ та серверів, пов’язаних з БД, велику роль відіграють СКБД, але не всі вони задовольняють вимоги захисту відповідного рівня.

Використання додаткових засобів захисту посилять захист сховищ та БД будь-якої організації, навіть у разі їх рефакторингу чи модернізації елементів корпоративних інформаційних систем. Вибір додаткового засобу захисту повністю залежить від того, яка СКБД була обрана, а також від пріоритетів та можливостей керівництва кожного підприємства. Також необхідно зазначити, що використання більше ніж одного додаткового засобу захисту може призвести до виникнення конфліктів пріоритетності та перешкоджання один одному. Тому варто не менше уваги приділити до сумісності цих засобів.

Для підвищення ефективності безпеки баз та сховищ даних підприємствам достатньо дотримуватися зазначених раніше правил:

- використовувати підготовлені запити;
- захищати порти серверів, на яких встановлена СКБД;
- використовувати шифровані з’єднання; постійно на корпоративному рівні забезпечувати оновлення паролів та забезпечувати їх складність;
- використовувати розподілені права доступу з їх мінімізацією для кожного окремого користувача;
- забезпечувати мінімальний доступ до вузлів корпоративної мережі.

Дотримуючись цих правил підприємство знизить ризики витоку інформації та зможе зберегти першість у своїй сфері.

Для визначення юридичної площини дій фахівця з кібербезпеки було проаналізовано нормативно-правову базу України.

Виявлено існуючу проблему недостатнього убезпечення БД наявними методами та способами, для вирішення якої пропонується створення додаткового фактору безпеки БД підприємства

РОЗДІЛ 2. РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ ПЕРЕДАЧІ ТОКЕНА АВТЕНТИФІКАЦІЇ

Перед початком розробки програмного модуля варто визначити: на що саме буде розрахований розроблений модуль. Порушення конфіденційності інформації починається на кроці отримання користувачем доступу до системи збереження та обробки даних.

2.1. База даних як об’єкт захисту

Оскільки для будь-якого підприємства важливим об’єктом для захисту є БД, яка зберігає важливу інформацію, варто розглянути:

1. Як саме користувач отримує доступ до роботи з даними.
2. Як система ідентифікує користувача в БД.
3. Як система реагує на НСД до своїх ресурсів.

2.1.1. Існуючі методи автентифікації у системі баз даних та їх порівняння.

Як зазначено в [17], було розглянуто існуючі методи автентифікації при порівнянні за своїми властивостями, а саме:

1. Слабка автентифікація.
2. Двофакторна автентифікація.
3. Біометрична автентифікація. [17]

Для доступу до будь-якої інформації в захищених системах необхідно пройти ідентифікацію та автентифікацію. У теоретичному розділі було розглянуто визначення цих термінів. Представлення користувача системі зазвичай виглядає як вказання свого імені (логіну) та паролю, який є підтвердженням, що користувач являється тим за кого себе видає. [17]

Найпростіший спосіб автентифікації полягає у тому, що система звіряє передані користувачем дані у вигляді логіна та пароля з даними, які містяться в базі. При повному співпадінні користувачу надаються права відповідні до його ролі в системі. Така автентифікація є слабкою і уразливою перед перехопленням та методом підбору паролю. [17]

Більш складнішим методом автентифікації є спосіб за допомогою додаткового значення, окрім логіна та паролю. Прикладом такої автентифікації є перевірка сервером логіна, пароля і додатково перевірка IP адреси користувача, який намагається автентифікуватись. У такій ситуації IP адреса виступає додатковим значенням перевірки користувача, оскільки кожний комп'ютер в мережі має унікальний мережевий ідентифікатор. Але якщо користувач перепідключиться до мережі, пристрій отримає новий мережевий ідентифікатор, а це означає, що сервер вже буде мати розбіжність у даних, отриманих від користувача. Навіть при співпадінні логіна і пароля, мережевий ідентифікатор буде відрізнятись від значення, яке отримав сервер при автентифікації користувача. Тому система зареєструє таку спробу входу в систему потенційно небезпечною. Як наслідок така не значна з першого погляду ситуація потрібна буде розглянута спеціалістом з безпеки підприємства, що в свою чергу може відволікти від можливо справжньої спроби НСД. [17]

Також варто розглянути метод двофакторної автентифікації. Суть якого полягає у додатковому підтвердженні особистості користувача, який намагається отримати доступ до системи. Наприклад, після введення логіна та паролю, користувачу буде надісланий код на мобільний, який складається с 4-6 символів. Зазвичай це є 4 або 6 символів у вигляді цифр, після введення яких користувач повністю підтверджує свою особистість, отримує токени доступу та заходить до системи. Безперечно, такий метод на перший погляд здається безпечним, який обійти зловмисником зводиться до нуля. Але якщо кіберзлодій викраде токени доступу, то зловмиснику вже не потрібно буде проходити автентифікацію. Система визначить таку автентифікацію як безпечну, оскільки користувач повністю пройшов автентифікацію і являється безпечним для

системи. Через деякий час, коли система знову починає перевіряти токени доступу та при умові, що вони являються валідними, надає користувачу новий токен доступу для продовження роботи в системі. [17]

Розглянемо найзатратніший для підприємства метод автентифікації: біометрична система ідентифікації користувача у системі. Безперечно, як показує статистика такий метод є безпечнішим на 86-93% ніж розглянуті раніше, оскільки характеристики такої системи вражають. Ймовірність відмови у доступі такої системи дорівнює 2-6%. А ймовірність помилкової ідентифікації користувача дорівнює 0,0001%. [18]

Постає задача: як уникнути такої ситуації, коли спеціаліст та система безпеки «шукають не там». Звичайно, першим на думку приходить зробити так, щоб під час перевірки системою токенів доступу, до уваги був піднесений додатковий параметр пристрою користувача. Якщо користуватись лише методом двофакторної автентифікації, то це означає, що користувачу потрібно буде кожний певний період отримувати код на мобільний для проходження автентифікації. Такий спосіб є безпечним, але не є зручним для самого користувача. Варто приділити цьому питанню увагу.

Для простішого розуміння було складено таблицю 2.1, де представлено порівняння відомих методів автентифікації. [20]

Таблиця 2.1

Порівняння методів автентифікації

Метод автентифікації	**Основні характеристики методу, що визначають його стійкість**	**Галузь застосування**	**Приклади застосування**
Багаторазові паролі	Безліч символів (алфавіт); Довжина.	WEB-сайти, мережеві служби, автентифікація, смартфони, планшети - Автентифікація на web-сайтах.	- Автентифікація користувачів в операційній системі. - Автентифікація при віддаленому доступі до мережевих служб.
Одноразові паролі	Безліч символів (алфавіт); Довжина.	Мобільний банк, Web-сайти, OTP-токени.	- Автентифікація при оплаті послуг через інтернет. - Автентифікація при відновленні пароля на Web-сайтах;
Двофакторна автентифікація	Додатковий елемент підтвердження.	Комп’ютери.	- Автентифікація при доступі до інформаційної системи
Біометрична автентифікація	Унікальність ознаки.	Комп’ютери, смартфони, планшети, системи збереження інформації	- Автентифікація при доступі до пристрою

Проаналізувавши методи простої автентифікації, потрібно відзначити, що вони мають ряд суттєвих недоліків, які негативно впливають на забезпечення безпеки інформації, зокрема вразливість системи з підробкою даних автентифікації. Щоб уникнути надання зловмисникові потенційної можливості для порушення конфіденційності, доступності та цілісності в системі, слід виконувати рекомендації по створенню і зберіганню пароля і використовувати схеми більш складної автентифікації. [20]

2.1.2. Проблематика методів автентифікації у системі БД.

Вся проблема методів автентифікації в системі БД полягає у тому, що кожний із способів має недоліки у своєму функціонуванні. Ідеального методу не існує, але перед спеціалістами з безпеки на сьогоднішній день стоїть важлива задача, суть якої полягає у створенні такого методу, який має менше уразливості перед кіберзлочинцями. [18]

Ідеальним способом ідентифікації користувача може бути біометрична автентифікація. Але як було зазначено раніше: такий спосіб є дуже затратним для підприємства.

Постає завдання розробки такого способу автентифікації, який буде максимально безпечним та не буде занадто затратним для підприємства.

2.2. Використання JSON Web Token при автентифікації в системі БД

Використання JWT пояснюється рядом переваг. Така технологія ще недостатньо вивчена зловмисниками. Для використання JWT достатньо базових знань такої мови програмування як Java Script. Тому варто поширювати використання такої технології.

2.2.1. Визначення JSON Web Token, його можливості та переваги

JWT містить в собі три блоки, а саме:

1. заголовок,
2. набір полів

3. сигнатуру.

Усі JWT-токени є зашифровані та представляють собою засіб автентифікації для кожного запиту від клієнта до серверу. Вони генеруються на сервері базуючись на секретному ключі, який в свою чергу зберігається на сервері. Токен доступу зберігається у клієнта та використовується при необхідності автентифікації будь-якого запиту. [21]

При спробі підробити дані, які містить в собі зашифрований токен, кіберзлочинець зіткнеться з тим, що такий токен стане не валідним, оскільки його сигнатура не буде відповідати початковим значенням. Можливість сгенерувати нову сигнатуру у зловмисника відсутня, тому що секретний ключ шифрування знаходиться на сервері. Також варто зазначити, що токен зберігається не в локальній пам’яті пристрою клієнта, а в пам’яті клієнтського додатку.

Як відомо з [19], токени поділяються на два види та є обмежені у часі існування:

1. Токени доступу, які використовуються для автентифікації запитів та зберігання додаткової інформації про користувача.

2. Токени повтору, які видаються сервером у результаті успішної автентифікації і використовуються у подальшому для створення нових токенів.

Такі токени слід зберігати в БД, тому що вони використовуються для обліку доступу та інвалідації викрадених токенів. Таким чином серверу буде відомо яким користувачам дозволено автентифікуватись.

Як зазначено в [19], працює це наступним чином:

1. Користувач ідентифікує себе, передаючи свій логін, пароль та унікальний ідентифікатор пристрою, з якого здійснює вхід у систему.

2. Сервер перевіряє коректність введеного логіна та паролю.

3. Якщо введені дані вірні, то сервер записує сесію в БД та створює токен доступу.

4. Сервер відправляє клієнту відповідні токени, створені на основі даних користувача.

5. Клієнт зберігає отримані від сервера токени. [19]

Варто зауважити, що процес запису сесії до таблиці БД має мати свої заходи безпеки. Сервер має перевірити скільки повторних сесій є у користувача. Наприклад, якщо у користувача більше ніж 1 сесія на це варто звернути увагу. При спробі встановити ще одну нову сесію, сервер повинен видаляти попередню. Таким чином можна уникнути підозрілих дій зі сторони користувача, оскільки система обнулить всі сесії окрім останньої.

Запис унікального ідентифікатора пристрою також несе в собі не менше значення. При ситуації коли кіберзлодій отримав токен доступу, сервер перевіряє час існування такого токена. Якщо час існування вичерпаний, сервер звертається до токена повтору, який надсилає зловмисник разом зі своїм ідентифікатором пристрою. Звісно, цей ідентифікатор буде відрізнятись з ідентифікатором, який був переданий користувачем при створенні токенів, тому сервер видалить зі своєї БД таку сесію та зафіксує спробу несанкціонованого оновлення токенів. Коли справжній користувач зробить спробу отримати доступ до БД через відсутність токену, система відхилить таку спробу та направить такого користувача на повторну автентифікацію. [19]

Такий спосіб автентифікації має переваги над іншими, тому що унікальний ідентифікатор кожного пристрою є особливим та не може повторюватись. У порівнянні з автентифікацією, де унікальним значенням виступає ідентифікатор мережевого рівня, тобто IP адреса, використання ідентифікатору пристрою має переваги в тому, що при необхідності використання VPN сервісів, користувачу БД не потрібно буде знову автентифікуватись в системі та проходити відповідний етап. Повторна автентифікація відбудеться тільки при спробі зловмисником перехватити токен доступу або при використанні нового пристрою, що призведе до зміни його ідентифікатору. Така ситуація трапляється на багато рідше ніж зміна IP адреси пристрою, яким орудує користувач. Простим прикладом є ситуація, коли підприємство змінює інтернет провайдера. Тоді всім користувачам системи

необхідно буде знову автентифікуватись, що може призвести до значних затримок у функціонуванні підприємства.

Перевагою такого методу є можливість використання такого способу підприємствами навіть низького рівня, оскільки він не є затратним і не потребує у своєму використанні високого рівня спеціаліста.

2.3 Розробка модуля забезпечення захисту

Розробка модуля забезпечення була основана на перевагах технології JSON Web Token, що роблять процес створення модулю не складною та корисною.

Спочатку необхідно визначити алгоритм модуля, що розробляється, блок-схема якого наведена рис. 2.1.

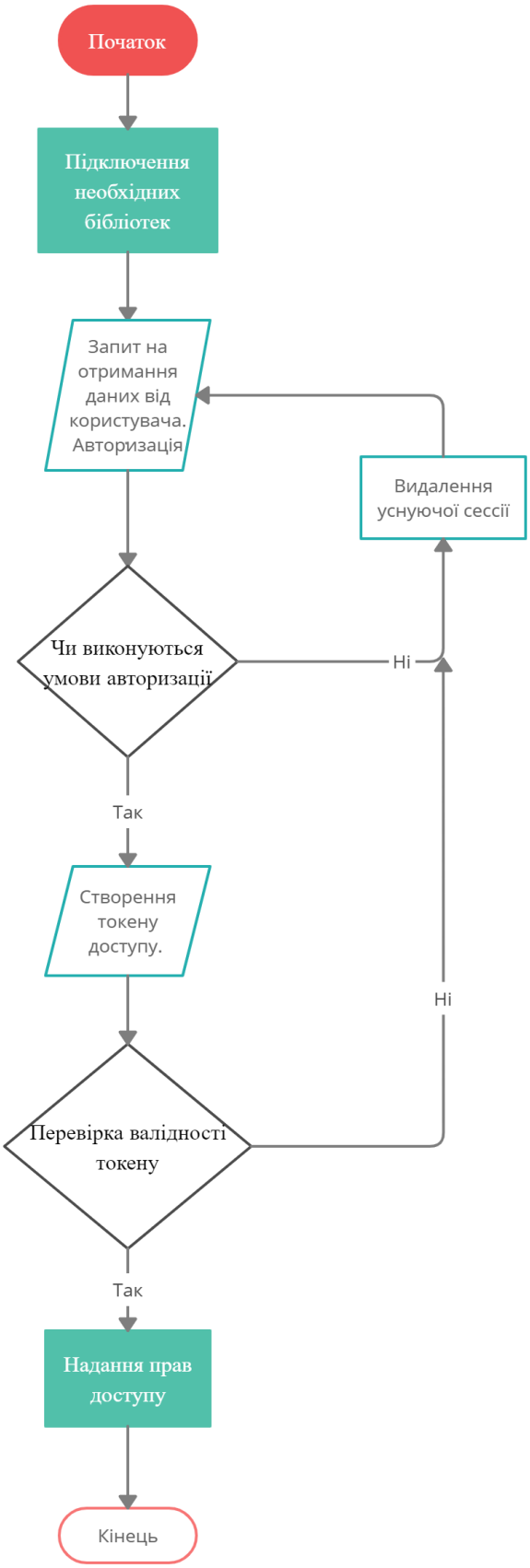

Рис. 2.1. Блок-схема програмного модуля, що розробляється

Програма працює за наступним алгоритмом:

1. Система отримує від користувача логін, пароль і ідентифікатор пристрою.
2. Сервер перевіряє коректність отриманих даних.
    a. Якщо дані коректні, буде створено токен доступу.
    b. Якщо дані не коректні, система поверне користувача до етапу автентифікації.
3. Сервер перевіряє валідність токену доступу через деякий час.
    a. Якщо токен валідний, система продовжує надавати доступ користувачу.
    b. Якщо токен не валідний, система поверне користувача до етапу автентифікації.

2.3.1. Технології використані для розробки модуля.

У вирішенні поставленого завдання була обрана мова JavaScript, оскільки це є достатньо популярна мова, яка містить в собі усі необхідні функції.

Оскільки JavaScript є мовою, що інтерпретується, дуже часто вона позиціонується як мова сценаріїв, а не як мова програмування, при цьому мається на увазі, що мови сценаріїв простіші і більшою мірою орієнтовані не на програмістів, а на звичайних користувачів. Справді, за відсутності контролю типів JavaScript можна допускати не критичні помилки, які допускають недосвідчені програмісти. [19]

Спочатку JavaScript розроблявся з метою на вбудовування в будь-які додатки і надання можливості виконувати сценарії. З найперших днів веб-сервери компанії Netscape включали в себе інтерпретатор JavaScript, що дозволяло виконувати JavaScript сценарії на стороні сервера. Аналогічним чином на додаток до Internet Explorer корпорація Microsoft використовує інтерпретатор JScript в своєму веб-сервері IIS і в продукті Windows Scripting Host. Компанія Adobe задіяла похідну від Java Script мову для управління своїм програвачем Flash-файлів. Компанія Sun також вбудувала інтерпретатор JavaScript в

дистрибутив Java 6.0, що істотно полегшує можливість вбудовування сценаріїв в будь-який Java-додаток [19].

2.3.2. Нововведення в автентифікації користувача.

Отже, ми розглянули основні аспекти автентифікації користувача:

1. Представлення себе в системі та отримання токену доступу.

   Користувач представляє себе в системі за допомогою логіна (login), пароля (password) та унікального ідентифікатора пристрою (uid).

   Основні елементи зображені на (рис. 2.2).

```
POST /api/auth/login
{
    "email": "",
    "password": "",
    "fingerprint": ""
}
```

Рис. 2.2. Основні елементи автентифікації користувача

де «login» - ім’я користувача в системі;

«password» - пароль користувача;

«uid» – унікальний ідентифікатор пристрою, який записується як додаткове значення.

Після визначення основних елементів автентифікації користувача потрібно задати формат даних для входу, який зображено на (рис. 2.3).

```
static get validationRules () {
  return {
    body: {
      password: new RequestRule(AuthModel.schema.password, { required: true }),
      email: new RequestRule(AuthModel.schema.email, { required: true }),
      uid: new RequestRule(AuthModel.schema.uid, { required: true })
    }
  }
}
```

Рис. 2.3. Фрагмент коду - функція задання формату даних входу

Коли користувач вводить логін та пароль, необхідно перевірити коректність введених даних (рис. 2.4.)

```
function checkPassword (password, hash) {
  assert.string(password, { notEmpty: true })
  assert.string(hash, { notEmpty: true })

  return new Promise((resolve, reject) => {
    bcrypt.compare(password, hash, (error, result) => {
      if (error) return reject(new AppError(error))
      if (!result) return reject(new AppError({ ...errorCodes.INVALID_PASSWORD }))
      return resolve(result)
    })
  })
```

Рис. 2.4. Фрагмент коду - перевірка введених даних

Після чого буде сформовано дані, які будуть використані для формування токену доступу (рис 2.5.)

```
const newRefreshSession = new RefreshSessionEntity({
  userId: user.id,
  ip: ctx.ip,
  ua: ctx.headers['User-Agent'],
  uid: ctx.body.uid,
  expiresIn: refTokenExpiresInMilliseconds
})
```

Рис. 2.5. Фрагмент коду - формування даних для створення токену

Далі на основі отриманих даних формується сам токен доступу (рис. 2.6.)

```
function makeAccessToken (userEntity) {
  assert.object(userEntity, { required: true })

  let config = {
    payload: {
      tokenType: type,
      username: userEntity.name,
      userRole: userEntity.role,
      email: userEntity.email,
      iss
    },

    options: {
      algorithm: 'HS512',
      subject: userEntity.id,
      expiresIn
    }
  }
```

Рис. 2.6. Фрагмент коду - формування токену доступу

Під час цього процесу сервер перевіряє максимальну кількість сесій. Як було зазначено раніше, якщо таких сесій більше ніж одна, то сервер має видалити попередній токен доступу(якщо такий є) та створити новий рис. 2.7.

```
function verifyRefreshSession (oldRefreshSession, newuid) {
  assert.instanceOf(oldRefreshSession, RefreshSessionEntity)
  assert.string(newuid, { notEmpty: true })

  return new Promise((resolve, reject) => {
    const nowTime = new Date().getTime()

    if (nowTime > oldRefreshSession.expiresIn) return reject(new AppError({ ...errorCodes.SESSION_EXPIRED }))

    if (oldRefreshSession.uid !== newuid) return reject(new AppError({ ...errorCodes.INVALID_REFRESH_SESSION }))
    return resolve()
  })
}
```

Рис. 2.7. Фрагмент коду - перевірка на наявність існуючих сесій та унікального цифрого ідентифікатора пристрою

Для досягнення безпеки максимальна кількість існуючих сесій користувача має бути не більше ніж одна. Інакше попередні сесії будуть видалені.

У разі успіху користувач зберігає на своєму пристрої токен доступу, який виглядає в наступному форматі (рис. 2.8.).

```
Set-Cookie: refreshToken='c84f18a2-c6c7-4850-be15-93f9cbaef3b3';
{
  body: {
    accessToken: 'eyJhbGciOiJIUzUxMiIsI...',
    refreshToken: 'c84f18a2-c6c7-4850-be15-93f9cbaef3b3'
  }
}
```

Рис. 2.8. Загальний вигляд токенів доступу

На основі отриманих даних від користувача, було створено токен доступу. Це робиться для того, щоб користувач системи не проходив етап автентифікації повторно, коли система через деякий час дасть запит на перевірку валідності цього токена.

2. Використання токена повтору. Саме цей пункт є важливим. Якщо користувач автентифікувався в системі і вже має токен доступу, то такий токен необхідно перевірити на валідність.

Це відбувається наступним чином. Сервер запитує токен повтору, який зберігає користувач. Потім видаляє старий сеанс та перевіряє початковий унікальний ідентифікатор пристрою користувача.

Для початку перевірка існуючих сесій, кількість яких не повинна перевищувати одиниці (рис. 2.9.)

```
async function _isValidSessionsCount (userId) {
  assert.validate(userId, UserModel.schema.id, { required: true })

  const existingSessionsCount = await RefreshSessionDAO.baseGetCount({ userId })
  return existingSessionsCount < MAX_REFRESH_SESSIONS_COUNT
}
```

Рис. 2.9. Фрагмент коду - перевірка кількості активних сесій користувача

Це є важливим моментом для вирішення поставленого завдання, оскільки саме тут система вирішує продовжити надання доступу користувачу або ж при ситуації, коли першопочатковий ідентифікатор не співпадає з ідентифікатором, отриманим повторно через деякий час, обмежити доступ до системи та створити запит на повторну автентифікації користувача.

У випадку успішного співпадіння ідентифікатора пристрою створюється, за раніше описаним способом, новий токен повтору, який є таким самим, як при створенні першого разу. Важливо щоб створений токен співпадав з першостворeним токеном повтору. Інакше знову ж таки система зафіксує таку спробу автентифікації як потенційно небезпечною.

Суть захисту, який представляє нововведення, полягає в наступному:

1. Зловмисник отримав токени доступу.
2. Закінчився час існування такого токену.
3. Зловмисник відправляє системі свій ідентифікатор пристрою (uid) для отримання повторного токену доступу.
4. Сервер перевіряє наданий користувачем ідентифікатор. Звісно, співпадіння не виявляється.
5. Сервер видаляє всі сесії користувача та створює нову сесії, в якій потрібно автентифікуватись.

2.3.3. Тестування програмного модуля в імітованому середовищі.

Розглянемо роботу розробленого модуля на прикладі. Існує БД в якій необхідно провести процес автентифікації у відповідному вікні. Виглядає воно наступним чином. Існує два поля. Одне з них – це логін. Друге – пароль.

Після введення коректного логіна і пароля система надає користувачу доступ до інформації, яку зберігає БД. Токен доступу має обмежений час існування. Через 30 хвилин користування БД потрібно буде отримати оновлений токен для подальшого користування системою. Якщо користувач не буде змінювати пристрій з якого працює, то це буде виглядати як швидке оновлення сторінки з можливістю продовжити роботу в БД.

Тепер необхідно розглянути ситуацію, коли зловмисник перехопив токен доступу та намагається отримати доступ. Складність системи може зіграти на користь, тому що за час поки кіберзлодій буде шукати інформацію в БД, сплине час дії токену та сервер буде потребувати його оновлення. Сервер перевірить, що сесії користувача існує вже дві, а не одна. Та зі спробою отримання унікального ідентифікатора буде виявлено, що друга сесія має не коректне значення. Після чого відбудеться оновлення сторінки на пристрої зловмисника і відповідно буде перехід на сторінку з автентифікацією. А зі сторони справжнього користувача буде таке ж швидке оновлення сторінки з можливістю подальшої роботи в БД.

Для тестування було використане ПЗ Fiddler. За її допомоги було перехоплено трафік та викрадено токен доступу. Та розширення GetThisCookies для спроби підмінити викрадений токен вже на іншому пристрої.

При спробі підміни було отримано відмову у вигляді помилки зображеної на рис. 2.10.

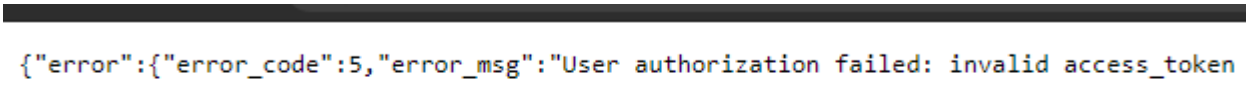

Рис. 2.10. Зображення помилки невалідної автентифікації

Це означає, що токен, який був представлений користувачем системі, не є валідним, що підтвердило дієвість розробленого модуля.

2.3.4. Переваги розробленого модуля забезпечення захисту.

Основна перевага такої розробки є простота в своєму використанні та в незначних затратах підприємства в підтримці розробленого модуля. Також варто відмітити випадок, коли зловмисник розуміє, що в системі існує перевірка ідентифікатору пристрою. Та навіть знаючи це, шанс на проникнення в систему не збільшується. Вгадати або підібрати такий ідентифікатор буде практично неможливим або займе у зловмисника значного обсягу часу.

При розробці модуля була використана динамічна, об’єктно-орієнтована мова, що набирає популярність в усіх сферах інноваційних технологій, Java Script. Тому спеціалісти, які можуть у подальшому використовувати цей програмний модуль, зможуть з легкістю модифікувати його в своїх цілях. Оскільки обрана мова програмування є однією з найпопулярніших мов, яка в наш час неухильно розвивається. Її використання у вирішені завдання є цілком коректним.

У порівнянні з безпечним біометричним методом автентифікації представлений в даній роботі метод є набагато простіше в обслуговуванні та менш затратний для підприємства низького або середнього рівня. Звісно, такий модуль є менш безпечним за біометричний метод, опираючись на характеристики біометричного методу з таблиці 1.

Порівнюючи наведений модуль захисту БД з методом двофакторної автентифікації такий спосіб менш затратний за часом, тому що користувач пропускає момент з відправленням коду на мобільний та час на введення коду у відповідне вікно програми. Також варто відмітити, що при спробі обходу двофакторної автентифікації, зловмисник може отримати номер мобільного користувача. У свою чергу це може загрожувати підприємству тим, що кіберзлодій може виявитись досить вмілим соціальним інженером. У подальшому за допомогою номера мобільного користувача зловмисник шляхом соціальних навичок може отримати конфіденціальну інформацію про

підприємство. При умові, що зловмисник дізнається про унікальний ідентифікатор пристрою, яким користується лише один робітник підприємства, це йому вцілому нічого не дає. Доступ до такого пристрою є і має бути лише у визначеного користувача, ідентифікатор якого ніде не записується та не оголошується.

2.3.5. Рекомендації щодо використання модуля.

Раніше було зазначено, що представлений модуль забезпечення безпеки не потребує від користувача високих навичок у сфері інноваційних технологій. Оскільки була обрана мова Java Script, розроблений програмний модуль встановлюється на сервер як додатковий сценарій.

За необхідністю можливо зменшити час існування токену доступу користувача, що дасть змогу здійснити більший контроль над користувачами в системі. Зменшивши час існування токену, спеціаліст збільшить кількість звертань від сервера до клієнта, що в свою чергу призведе до збільшення навантаження на систему. Тому варто обрати такий час, при якому навантаження на сервер буде мінімальним, але за котрий зловмисник не зможе знайти важливу для підприємства інформацію.

Поетапне використання модуля:

1. Встановити сценарій на сервер.
2. Встановити необхідні обмеження автентифікації користувача.

2.4. Висновки до розділу 2

Розглянувши відомі вразливості автентифікації, було поставлено завдання: забезпечити безпеку автентифікації БД підприємства та запобігти несанкціонованому доступу зловмисників до конфіденціальної інформації. У вирішенні цього завдання було використано мову програмування JavaScript для написання сценарію сервера та також технології для тестування, а саме програма

Fiddler для перехоплення та аналізу трафіку і розширення EditThisCookie для підміни токену доступу, який був перехоплений.

Використавши унікальний ідентифікатор пристрою, як додаткове значення при верифікації та ідентифікації користувача, було зменшено час автентифікації клієнта в системі, при чому було збережено достатній рівень захищеності такої автентифікації.

Переваги розробленого модуля полягають у наступному:

1. При розробці було використано популярну динамічну мову програмування, що у майбутньому надає можливість модифікувати розроблений модуль;
2. У порівнянні з простою автентифікацією розроблений модуль надає більший рівень захищеності шляхом створення додаткового рівня захисту під час автентифікації користувача в БД.
3. У порівнянні з біометричним способом автентифікації, розроблений модуль є простішим у своєму використанні, обслуговуванні та є набагато менш затратним для підприємства.
4. У порівнянні з двофакторною автентифікацією, розроблений модуль є менш затратний за часом, що надає перевагу у швидкості доступу користувача до інформації, яка зберігається в БД.

Вироблені рекомендації щодо користування розробленим модулем, який розрахований на використання спеціалістом з інформаційної безпеки для підвищення контролю над доступом до бази даних:

1. У значенні змінної MAX_REFRESH_SESSIONS_COUNT необхідно вказати максимально допустиму кількість сесій користувача.
2. Внести до бази даних користувачів додаткове значення у вигляді унікального ідентифікатора пристрою, окрім логіна та пароля.
3. Запускати такий модуль бажано у приватній мережі підприємства

Проведення тестування розробленого модуля дозволило переконатись в його дієвості, оскільки дозволило запобігти зловмисним діям щодо НСД до конфіденційної інформації БД підприємства.

Отже, представлений програмний модуль забезпечення конфіденційності інформації підприємства надає не тільки переваги в захищеності системи над слабкою автентифікацією, а також скорочує час доступу користувача до даних, яка зберігає БД. Поставлене завдання, метою якого було забезпечити конфіденційність інформації БД підприємства, було досягнуто.

ВИСНОВКИ

У представленій дипломній роботі було розглянуто основні визначення стосовно такого терміну як БД, методи автентифікації та ідентифікації користувача при роботі з БД, вимоги та умови до використання таких методів. Також було переглянуто та враховано статистику вразливостей БД. Такі системи зі значними об’ємами інформації потребують захищеності, тому що кожного дня піддаються атакам зловмисників. Будь-який виток інформації може призвести до необоротних наслідків. Для запобігання цьому, необхідно отримувати актуальну інформацію про вразливості систем та прикладати зусиль у розробці ПЗ, щоб зменшити шанс несанкціонованого проникнення до ІС.

Роблячи висновки зі статистики, представлених у роботі джерел, на даний момент ще не існує такого засобу забезпечення захисту, який би міг повністю забезпечити непроникність до системи та збереження конфіденційності інформації підприємства. З кожним днем зловмисники вигадують нові способи обійти існуючий захист ІС, а спеціалісти з питань безпеки намагаються перешкоджати у цьому. Тому як завдання було поставлено мету розробити такий модуль ПЗ, який міг би забезпечити збереження конфіденційності інформації БД підприємства на основі обмеження НСД до ІС.

Розглянувши відомі вразливості автентифікації, було поставлено завдання: забезпечити безпеку автентифікації БД підприємства та запобігти несанкціонованому доступу зловмисників до конфіденціальної інформації. У вирішенні цього завдання було використано мову програмування JavaScript для написання сценарію сервера та також технології для тестування, а саме програма Fiddler для перехоплення та аналізу трафіку і розширення EditThisCookie для підміни токену доступу, який був перехоплений.

Використавши унікальний ідентифікатор пристрою, як додаткове значення при верифікації та ідентифікації користувача, було зменшено час автентифікації

клієнта в системі, при чому було збережено достатній рівень захищеності такої автентифікації.

Отже, представлений програмний модуль забезпечення конфіденційності інформації підприємства надає не тільки переваги в захищеності системи над слабкою автентифікацією, а також скорочує час доступу користувача до даних, яка зберігає БД. Поставлене завдання, метою якого було забезпечити конфіденційність інформації БД підприємства, було досягнуто.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Інформація та документація. Бібліотечно-інформаційна діяльність. Терміни та визначення понять : ДСТУ 7448:2013. — Київ : Мінекономрозвитку України, 2014. [Електроний ресурс] – режим доступа : URL http://www.library.kherson.ua/2016/dstu7448-2013.pdf
2. Організація баз даних: Методичні вказівки до самостійної роботи студентів за спеціальностями 6.050102/123 «Комп'ютерна інженерія»,125 «Кібербезпека» / уклад. В.В. Сидоренко, Л.В. Константинова—Кропивницький: ЦНТУ, 2017. [Електроний ресурс] – режим доступа : URL http://dspace.kntu.kr.ua/jspui/bitstream/123456789/8428/3/Org_baz_dan_sam_rab_metod.pdf
3. Глушков С.В., Ломотько Д.В. Базы данных. — Х., 2000; Джексон П. Введение в экспертные системы. — М., 2001; Шафрин Ю.А. Основы компьютерной технологии. — М., 1998. [Електроний ресурс] – режим доступа : URL https://www.twirpx.com/file/116530/
4. Домарев В.В. Безопасность информационных технологий. Системный подход / Домарев В.В. – К.: ООО «ТИД «ДС», 2004. [Електроний ресурс] – режим доступа : URL https://security.ukrnet.net/wp-content/uploads/2020/01/chap_02-1.pdf
5. Есин В. И. Безопасность информационных систем и технологий / Есин В. И., Кузнецов А. А., Сорока Л. С. – Х. : ЭДЭНА, 2010. [Електроний ресурс] – режим доступа : URL https://core.ac.uk/download/pdf/84825441.pdf
6. Полтавцева М. А., Хабаров А. Р. Безопасность баз данных: проблемы и перспективы Международный журнал Программные продукты и системы №3 2016. [Електроний ресурс] – режим доступа : URL https://cyberleninka.ru/article/n/bezopasnost-baz-dannyh-problemy-i-perspektivy/viewer

7. Дослідження витоків інформації з підприємств. [Електроний ресурс] – режим доступа URL: https://www.canalys.com/newsroom/cybersecurity-investment-2020
8. Традиционные методы и средства защиты данных, реализованные в базе данных с универсальной моделью данных / Л. С. Сорока, В. И. Есин, М. В. Есина // Академія митної служби України. Вісник Академії митної служби України. Серія "Технічні науки". 2010 р. № 2 (44) / Академія митної служби України. - Д., 2010. [Електроний ресурс]. – режим доступа : URL http://irbis-nbuv.gov.ua/cgi-bin/irbis_nbuv/cgiirbis_64.exe?C21COM=2&I21DBN=UJRN&P21DBN=UJRN&IMAGE_FILE_DOWNLOAD=1&Image_file_name=PDF/vamsutn_2010_2(44)__3.pdf
9. Кавун С.В. Інформаційна безпека. Навчальний посібник. Ч.1/С.В. Кавун, В.В. Носов, О.В. Мажай. – Харків: Вид. ХНЕУ, 2008. [Електроний ресурс] – режим доступа : URL http://www.repository.hneu.edu.ua/jspui/bitstream/123456789/3068/1/%D0%9D%D0%B0%D0%B2%D1%87%D0%B0%D0%BB%D1%8C%D0%BD%D0%B8%D0%B9%20%D0%BF%D0%BE%D1%81%D1%96%D0%B1%D0%BD%D0%B8%D0%BA.%20%D0%86%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D1%96%D0%B9%D0%BD%D0%B0%20%D0%B1%D0%B5%D0%B7%D0%BF%D0%B5%D0%BA%D0%B0.%20%D0%9A%D0%B0%D0%B2%D1%83%D0%BD%20%D0%A1.%D0%92..pdf
10. Using Secure Connections [Електроний ресурс] – режим доступа : URL https://dev.mysql.com/doc/refman/5.6/en/secureconnections. html.
11. Douglas Crawford OpenVPN over TCP vs. UDP: what is the difference, and which should I choose? [Електроний ресурс] – режим доступа : URL https://www.bestvpn.com/openvpn-tcp-vsudp-difference-choose/
12. Козаченко І. П., Голубєв В. О. Загальні принципи захисту інформації в банківських автоматизованих системах, 2005. – 172 с.

13. Конституція України: Закон України від 28 черв. 1996 р. № 254 к/96-ВР // Відом. Верховної Ради України. – 1996. – № 30. – [Електронний ресурс]. – Режим доступу: URL https://zakon.rada.gov.ua/laws/show/254%D0%BA/96%D0%B2%D1%80#Text.
14. Про Національну програму інформатизації : Закон України від 4 лют.1998 р. № 74/98-ВР/ [Електронний ресурс]. – Режим доступу: http://zakon5.rada.gov.ua/laws/show/74/98-%D0 %B2 %D1 %80.
15. Про телекомунікації : Закон України від 18 листоп. 2003 р. № 1280–IV [Електронний ресурс]. – Режим доступу: http://zakon5.rada.gov.ua/laws/show/1280–15/page5.
16. Бачило И. Л. Информационное право : основы практической информатики : учеб. пособие / И. Л. Бачило. – Москва, 2001. [Електронний ресурс]. – Режим доступу: https://ifap.ru/library/book281.pdf
17. Види біометричної автентифікації та методи їх оцінки / К.В. Колесніков, Б.П. Ободовський // Штучний інтелект. — 2017. — № 3-4. [Електронний ресурс]. – Режим доступу: http://dspace.nbuv.gov.ua/handle/123456789/162340
18. Дворянкин С. В. Речевая подпись / Под ред. заслуженного деятеля науки, д.т.н. проф. А. В. Петракова. – М.: РИО МТУСИ, 2003 [Електронний ресурс]. – Режим доступу: https://ela.kpi.ua/bitstream/123456789/9839/1/26.pdf
19. JavaScript. Подробное руководство. – Пер. с англ. – СПб: СимволПлюс, 2008. [Електронний ресурс]. – Режим доступу: http://kharchuk.ru/JavaScript.pdf
20. Мартынова, Л. Е. Исследование и сравнительный анализ методов аутентификации / Л. Е. Мартынова, М. Ю. Умницын, К. Е. Назарова, И. П. Пересыпкин. — Текст : непосредственный // Молодой ученый. — 2016. — № 19 (123). [Електронний ресурс]. – Режим доступу: http://rrinformation.ru/media/information/2019/3/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D1%8B%D0%B5_%D1%82%D0%B5%D1%85%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D0%B8_4_3.pdf