Міністерство освіти і науки України
Рівненський державний гуманітарний університет
Кафедра інформаційних технологій та моделювання

Кваліфікаційна робота

за освітнім ступенем «бакалавр»

на тему:

Організація захисту персональних даних у системах з публічним доступом

Виконала:
студентка IV курсу
групи ІПЗ-41
спеціальності 121 «Інженерія програмного забезпечення»
Шеремет Олена Петрівна

Науковий керівник:
к.ю.н, доцент
Кіндрат П.В.

Рівне – 2024

ЗМІСТ

СПИСОК ВИКОРИСТАНИХ СКОРОЧЕНЬ

GDPR	General Data Protetcion Regulation
ISO	Intrnational Organization Regulation
СУІБ	Система управління інформаційної безпеки
ІКС	Інформаційно-комунікційні системи
БД	База даних
ORM	Object-Relational Mapping
2FA	Two-Factor Authentication

ВСТУП

Актуальність теми. Сучасне інформаційне суспільство ґрунтується на широкому залученні інформаційних технологій у повсякденне життя кожної людини. Розвиток Інтернету та відкритих мереж стимулював виникнення та розповсюдження різного роду сервісів орієнтованих на спрощення забезпечення рутинних інформаційних потреб і процесів та їх автоматизацію. Як наслідок, обсяг персоніфікованих даних які запитуються, обробляються та зберігаються онлайн сервісами продовжує зростати. Водночас, накопичення персональних даних окремими сервісами не лише дозволяє розробникам підвищити якість послуг що надаються, а й ставить перед ними серйозний виклик щодо захисту цих даних, оскільки зростають ризики порушення їх конфіденційності та безпеки. Тому захист персональних даних у системах загального доступу набуває вирішального значення.

Проблематика організації захисту інформації в інформаційних системах є предметом жвавого обговорення як в Україні (Бобало Ю. Я., Гребенюк А.М., Галінкіна В.С., Хорошко В.О. та ін.) так і в міжнародній спільноті (Lester Nichols, Ruben F. Pereira, Rohani Rohan та ін.). Особливої ваги вона набуває в контексті публічних систем, які мають доступ до персональних даних великої кількості користувачів (як то соціальні мережі, інтернет-магазини, електронні державні послуги тощо). Недостатній рівень захисту таких системах може мати серйозні наслідки обумовлені витоком чи крадіжкою персональних даних що призведуть до порушення права на приватність.

Більшість дослідників розглядає організацію захисту персональних даних у інформаційних системах як комплексну задачу, яка включає технічні, організаційні та правові заходи. Проте в Україні переважна більшість досліджень концентрується саме на правовій складовій, упускаючи при цьому методи імплементації результатів в практичну площину. Тому розробка ефективної стратегії захисту інформації для організацій, які працюють з персональними даними користувачів у публічному просторі Інтернету та її імплементація в практичній площині є безсумнівно актуальною задачею.

Мета дослідження полягає у вивченні методів та технологій організації інформаційної безпеки програмних комплексів, що можуть бути застосовані для забезпечення високого рівня конфіденційності інформації та безпеки персональних даних в системах з публічним доступом.

Дослідження спрямоване на розробку політики безпеки автоматизованої системи обробки інформації (АСОІ), що містить персональні дані, та практичне застосування її положень для організації належного захисту персональних даних у інформаційних системах з публічним доступом.

Для досягнення сформульованої мети було поставлено такі завдання:

– систематизувати та розкрити зміст нормативно-правових вимог до захисту персональних даних;

– дослідити методи технічного захисту інформації, зокрема в системах з публічним доступом;

– спроектувати інформаційну систему з публічним доступом яка здійснює збереження та обробку персональних;

– обґрунтувати потенційні вразливості розробленої системи та визначити загрози безпеці персональних даних;

– розробити політику інформаційної безпеки АСОІ, що містить персональні дані;

– імплементувати в інформаційну систему технічні заходи безпеки у відповідності до розробленої політики.

Об’єктом дослідження є захист інформації в системах з публічним доступом

Предметом дослідження є політика інформаційної безпеки щодо інформаційних систем що містять персональні дані та механізми її імплементації в системах з публічним доступом.

Методи дослідження. Для вирішення визначених завдань і досягнення поставленої мети дипломної роботи використовувався комплекс методів дослідження, таких як:

– аналіз літератури та нормативних актів: сюди входило вивчення наукових джерел, законів та стандартів щодо захисту персональних даних, необхідних для розуміння теоретичних аспектів;

– методи системного аналізу: для аналізу функціональних та архітектурних характеристик системи та для оцінки її поточного стану;

– методи ризик-менеджменту: для оцінки ризиків, пов’язаних з обробкою персональних даних, і для аналізу потенційних загроз і вразливостей в системі;

– емпіричні методи: для вивчення, аналізу існуючих методів із захисту даних, а також для оцінювання їх ефективності у реальних ситуаціях;

– методи проектування: для розробки та проектування політики безпеки та систем захисту персональних даних;

– технічні методи: для впровадження технічних заходів безпеки.

Практичне значення дослідження. Розробка політики безпеки інформаційних систем що здійснюють обробку персональних даних та знаходяться в публічному доступі дозволяє застосовувати її положення для будь-якої подібної системи що сприятиме мінімізації ризиків несанкціонованого доступу до персональних даних.

Запропонований підхід до імплементації запропонованої політики безпеки передбачає удосконалення системи захисту персональних даних включає шифрування, перевірку даних і безпечні з’єднання та забезпечує надійний захист конфіденційної інформації з точки зору публічного доступу. Запропоновані технічні інструменти і процедури контролю доступу можуть бути інтегровані як в існуючі так і нові інформаційні системи, які здійснюють обробку персональних даних, підвищуючи їх рівень захисту та забезпечуючи відповідність нормативним вимогам щодо захисту персональних даних.

Апробація і впровадження результатів дослідження.

Основні теоретичні та практичні результати дослідження було висвітлено та обговорено на XI Міжнародній науковій конференції «Студентські наукові дискусії поза форматом» (м. Івано-Франківськ, 11 квітня 2024 року). [12].

Структура роботи. Дипломна робота складається зі вступу, трьох розділів, висновків, переліку використаних джерел та додатків.

Вступ включає обґрунтування актуальності теми, визначення мети, завдань дослідження, а також методологічної основи роботи.

Розділ 1. Теоретичні аспекти захисту персональних даних охоплює три підрозділи. У першому розглядаються принципи обробки та захисту персональних даних, в другому – законодавчі вимоги до захисту персональних даних, а в третьому аналізуються технічні методи захисту персональних даних.

Розділ 2. Організація системи обробки персональних даних з публічним доступом складається з 4 підрозділів. У першому підрозділі описуються функціональні та архітектурні особливості системи. В другому визначаються можливі загрози та вразливості в системі. Третій розділ присвячений оцінці ризиків при обробці персональних даних в системі. Четвертий підрозділ присвячений аналізу існуючих методів захисту даних.

Розділ 3. Розробка та впровадження захисту персональних даних для систем з публічним доступом складається з 3 підрозділів. У першому описується проектування системи захисту, в другому описане детальне застосування технічних заходів безпеки в системі, а в третьому описана розробка процедур контролю доступу до персональних даних.

Висновки підсумовують дослідження, а також формулюють основні висновки та рекомендації.

Список використаних джерел включає список літератури та інших джерел, використаних при написанні роботи.

Додатки містять додаткову інформацію ,яка допомагає краще зрозуміти основний зміст роботи. Зокрема це :

1. Додаток А. Схема БД системи що здійснює обробку персональних даних.

2. Додаток Б. Політика інформаційної безпеки.

Загальний обсяг роботи становить 48 сторінок. Вона містить 1 рисунок. Список використаних джерел включає 20 найменувань. Обсяг додатків – 13 сторінок

РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ ЗАХИСТУ ПЕРСОНАЛЬНИХ ДАНИХ В ІНФОРМАЦІЙНИХ СИСТЕМАХ

1.1 Зміст обробки і захисту персональних даних

Функціонування сучасного суспільства передбачає генерування, обробку і накопичення значних об’ємів інформації різного ґатунку. На сьогодні людство генерує колосальні об’єми інформації які щороку стрімко збільшуються. Так по оцінкам компанії Seagate свідчать, що лише за один рік (з 2016 по 2017) обсяги згенерованої людством інформації зросли в 10 разів. [17] Обробка таких об’ємів даних потребує використання щоразу складніших та досконаліших інформаційних систем, що призводить до зростання ризиків пов’язаних із забезпеченням захищеності інформації.

Звісно, не вся інформація яка генерується людством потребує захисту. А отже і не всі системи обробки інформації мають володіти відповідним функціоналом. Тож вирішення питання необхідності інтеграції системи захисту у кожен окремий програмний продукт є важливим етапом для будь-якого розробника програмного забезпечення.

Для вирішення цього питання необхідно зрозуміти обробка якої інформації буде здійснюватись в системі, категоріювати її та співвіднести з певним переліком інформації що підлягає захисту. До таких категорій, зазвичай, відносять таємну інформацію, яка може містити державну чи комерційну таємницю. Визначення відповідного категріювання інформації врегульовується певним розпорядчим, або нормативним актом який може суттєво відрізнятися від системи до системи. Проте є один вид інформації що як на національному так і на міжнародному рівні визначений таким, що гарантовано потребує захисту – це персональні дані.

Користувачі не хочуть, щоб їхня персональна інформація без необхідності та безцільно передавалася третім особам. Це пов’язано з тим, що значна частина такої інформації може бути "чутливою". У таких випадках ті, хто передає, обробляє і зберігає персональні дані, повинні керуватися правовою

основою, яка визначає політику таких дій у будь-якій операції, де обробляються персональні дані.

Визначення персональних даних міститься в Законі України "Про захист персональних даних". *Персональні дані* – відомості чи сукупність відомостей про фізичну особу, яка ідентифікована або може бути конкретно ідентифікована. [2] Проте, як бачимо, це визначення не дає точного переліку персональних даних, та передбачає певну варіативність у своєму трактуванні. Це обумовлено як природою власне інформації, так і особливостями її використання. Так, до прикладу, 10 цифр номеру телефону не можуть вважатись персональними даними допоки вони не співвідносяться з конкретною фізичною особою.

Така варіативність трактування призводить до складнощів в організації належної обробки персональних даних – будь-якої дії або сукупності дій, таких як збирання, реєстрація, накопичення, зберігання, адаптування, зміна, поновлення, використання і поширення (розповсюдження, реалізація, передача), знеособлення, знищення персональних даних, у тому числі з використанням інформаційних (автоматизованих) систем. [2]

Інституція захисту персональних даних є відносно новою, тому відповідне їй законодавство та методологічні і технологічні рішення не повністю сформовані та динамічно розвиваються щоб відповідати як технологічним змінам та інноваціям що породжуються нашою цивілізацією, так і вимогам суспільства. На даний час науковці виділяють загальні принципи обробки та захисту персональних даних, якими є: прозорість, чесність, законність, обмеження мети, мінімізації даних, точності даних, обмеження періоду зберігання даних, безпеки даних та принцип підзвітності. [1]

Водночас, до основних проблем обробки персональних даних включають:

1. Безпеку та захист даних, що обумовлено ризиками витоку або крадіжки даних через кібератаки, зловмисне програмне забезпечення, фішинг та інші методи.

2. Приватність, яка може бути скомпрометована шляхом недотримання конфіденційності та некоректним повідомленням користувачів про те, як їхні дані використовуються.

3. Відповідність законодавству що обумовлена складність узгодження і дотримання різних нормативно-правових актів та стандартів на міжнародному та національному рівні.

4. Забезпечення доступу та контролю користувачів до своїх власних даних і можливість контролювати, як ці дані використовуються.

5. Вирішення етичних питань обробки персональних даних, як то використання персональних даних для маніпулювання поведінкою користувачів, або використання сучасних технологій, таких як штучний інтелект і великий обсяг даних (Big Data).

6. Транспарентність функціонування інформаційних систем, а саме відсутність прозорості у способах збирання, зберігання та обробки персональних даних. [16]

Для усунення визначених проблем пропонується використовувати комплексні системи захисту інформації, які мають на меті забезпечити захист персональних даних від незаконної обробки, а також від незаконного доступу до них, відповідно до плану захисту інформації в системі, що містить: завдання захисту персональної інформації, класифікації персональної інформації, опис особливостей технології обробки інформації, модель загроз для персональних даних в системі, вимоги щодо захисту персональних даних та правила доступу до них. [2]

1.2 Нормативно правові вимоги до захисту персональних даних

1.2.1 Загальне регулювання захисту персональних даних

Законодавчі акти, які регулюють обробку та захист персональних даних в Україні, відіграють ключову роль у створенні ефективної системи захисту приватності громадян та відповідності міжнародним стандартам у цій сфері. Основними з них є:

1. Закон України "Про захист персональних даних", який встановлює основні принципи та правила обробки персональних даних. Він передбачає, що персональні дані можуть оброблятися лише за згодою суб'єкта персональних даних або на інших законних підставах. Закон також визначає права суб'єктів персональних даних, включаючи право на доступ доданих та їх виправлення, а також встановлює обов'язок розпорядників даних забезпечувати безпеку та конфіденційність даних.[2]

2. Закон України "Про захист інформації в інформаційно-комунікаційних системах" визначає загальні вимоги до захисту інформації, в тому числі персональних даних, в інформаційно-комунікаційних системах. Закон встановлює обов'язок забезпечувати безпеку та конфіденційність інформації, а також механізми реагування на порушення безпеки даних.[3]

Ці два закони становлять основу правового регулювання механізмів обробки та захисту персональних даних в Україні. Вони встановлюють правові засади діяльності організацій, що здійснюють обробку даних, визначають права та обов'язки зацікавлених сторін, а також встановлюють механізми контролю та відповідальності за порушення стандартів захисту даних.

Крім того, українське законодавство містить інші нормативно-правові акти, які можуть мати відношення до захисту персональних даних. Це, наприклад, закони, що регулюють діяльність у певних секторах економіки, таких як фінанси, охорона здоров'я та освіта, а також закони, що встановлюють специфічні вимоги до захисту персональних даних у цих секторах.

Зазначені вище закони на нормативно-правовому рівні забезпечують належний рівень захисту персональних даних в Україні та гармонізацію з міжнародними стандартами у цій сфері.

1.2.2 Міжнародні стандарти щодо зберігання та обробки персональної інформації

Зважаючи на глобальний характер функціонування інформаційної сфери, потреба у нормативному врегулюванні обробки персональних даних не може обмежуватись виключно національним законодавством. З метою забезпечення

належної транскордонної співпраці Україна імплементувала низку міжнародних стандартів. Ці стандарти покликані забезпечити єдність та високий рівень захисту конфіденційності персональних даних у всьому світі та встановлюють вимоги, принципи та процедури, яких слід дотримуватися підчас зберігання та обробки персональних даних.

Одним з таких міжнародних стандартів, які є наріжним каменем інформаційної безпеки є *Загальний регламент про захист даних (GDPR),* прийнятий Європейським Союзом (ЄС).

Загальний регламент захисту персональних даних (GDPR) набрав чинності 25 травня 2018 року. GDPR автоматично набув законної сили у всіх державах-членах Європейського союзу та країнах Європейської економічної зони (ЄС/ЄЕЗ). [4]

Відповідно до вимог описаних в GDPR компанії повинні:

1. Забезпечити наявність безпечних систем для захисту персональних даних від кібератак та інших атак, пов’язаних з випадковою втратою даних, коли дані передаються між компаніями, шифрувати дані, щоб мінімізувати ризик їх втрати.

2. Мати можливість ідентифікувати та частково або повністю видалити всі дані, що стосуються особи, якщо особа просить видалити ці дані. Наприклад, коли особа звільняється з компанії, вона може вимагати, щоб усі дані про неї, включно з фотографіями, були назавжди видалені (за деякими винятками) з локальних дисків і мереж.

3. Довести, що збережені дані особи будуть використовуватися лише за призначенням. [15]

Крім усього, для компаній, що здійснюють обробку персональних даних, GDPR передбачає нові поняття, такі як DPO (data protection officer) і Representative.

DPO (офіцер з захисту даних) – за своїми функціями він відповідає інспектору з персональних даних, який в деяких випадках повинен обов’язково призначатися в штат компанії, що працює з персональними даними.

В той час як *Representative* (представник) – це обов'язковий представник (фізична або юридична особа), яка має можливість представляти інтереси компаній, які не зареєстровані на території Європейського Союзу і не мають своєї філії або представництва для можливості комунікації з місцевими органами управління з питань захисту персональних даних.

Також, в тексті GDPR зазначено багато інших вимог. Дослідження і застосування яких покликано допомогти компанії уникнути інформаційних ризиків у майбутньому.

У разі недотримання вимог GDPR або серйозне порушення цього закону штрафи можуть становити до 4% від обороту компанії або до 20 мільйонів євро (в залежності від того, що більше).

Положення GDPR стосуються усіх без винятку компаній, однак в першу чергу зосереджені на компаніях, які функціонують у сфері IT технологій і здійснюють свою діяльність через всесвітню мережу Internet, так як їх діяльність більшою мірою пов'язана з персональними даними ніж діяльність будь-яких інших компаній. [5]

Іншим важливим міжнародним стандартом є *ISO/IEC 27001*, який визначає стандарти для систем управління інформаційною безпекою.

ISO/IEC 27001 – це провідний міжнародний стандарт для впровадження інтегрованих систем управління інформаційною безпекою. Він орієнтований на виявлення, оцінку та управління ризиками в процесах обробки інформації. Безпека конфіденційної інформації виділяється як важливий стратегічний елемент.

Відповідно до ISO/IEC 27001 інформація що обробляється інформаційною системою має бути класифікована. На цій класифікації засновано заходи захисту системи управління інформаційною безпекою (СУІБ).

СУІБ створює основу для захисту персональних-даних та їхньої конфіденційності. Водночас цей стандарт гарантує доступність IT-систем. Ефективність СУІБ має оцінюватись і перевіряєтись незалежним зовнішнім органом.

Застосування загальних вимог стандарту ISO/IEC 27001 повинно відображати конкретну ситуацію в інформаційній системі та залежати від потреб, цілей і вимог безпеки, організаційних процесів, а також розміру та структури підприємства.

Гармонізація інформаційних процесів у відповідності до ISO 27001 надає ряд переваг, а саме:

1. Постійне підвищення рівня безпеки
2. Мінімізація існуючих ризиків
3. Виконання вимог відповідності
4. Підвищення обізнаності співробітників
5. Підвищення задоволеності клієнтів

Внутрішні аудити та управлінські огляди за участю вищого керівництва є внутрішніми засобами досягнення цілей стандарту.

Сертифікована система менеджменту дозволяє продемонструвати, що організація вирішує питання ризиків у структурований спосіб і прагне до постійного вдосконалення, що робить її більш стійкою до небажаних впливів.[6]

1.3 Технічні методи захисту персональних даних

Попри визначення необхідності захисту персональних даних нормативно-правові документи України не визначають методи та засоби здійснення такого захисту. Вони лише встановлюють права та обов’язки сторін в інформаційних відносинах та визначають відповідальних осіб з а дотримання вимог законів.

Натомість, міжнародні стандарти підходять до питання захисту персональних даних більш системно, визначаючи не лише напрямки їх захисту, а й обґрунтовуючи відповідні механізми їх реалізації. Згідно означених стандартів, до основних напрямків забезпечення захисту персональних даних в інформаційних системах відносять: гарантування конфіденційності даних, шифрування даних, моніторинг функціонування інформаційної системи.

1.3.1 Шифрування даних

Шифрування – це процес перетворення звичайного тексту (відомого як відкритий текст) у нечитабельну форму(відомий як зашифрований текст або шифротекст) за допомогою алгоритму (відомого як шифр) для забезпечення конфіденційності даних. [18. С. 125] Криптографія використовує ключ, який визначає спосіб перетворення повідомлення. Тільки ті, хто має правильний ключ, можуть розшифрувати дані і отримати доступ до оригінального вмісту.

Шифрування використовується в цифровому середовищі для захисту конфіденційної інформації, такої як паролі, банківська інформація та особисті повідомлення. Використовуваний метод шифрування може зашифрувати дані таким чином, що тільки ті, хто має правильний ключ, можуть їх розшифрувати, забезпечуючи високий ступінь безпеки інформації.

Існують різні типи шифрування, які використовуються по-різному залежно від мети та сценарію безпеки. [14] Найбільш розповсюдженими типами шифрування є:

– Симетричне шифрування: при симетричному шифруванні для шифрування і розшифрування даних використовується один і той же ключ. Це означає, що обидві сторони, які беруть участь в обміні даними, повинні мати доступ до цього ключа. До поширених алгоритмів симетричного шифрування належать DES (Data Encryption Standard), AES (Advanced Encryption Standard) та IDEA (International Data Encryption Algorithm). [18, с. 127]

– Асиметричне шифрування: використовує два ключі: відкритий і закритий ключі. Відкритий ключ використовується для шифрування даних, а закритий – для їх розшифрування. Така схема шифрування дозволяє безпечно обмінюватися ключами навіть через ненадійні канали зв’язку. Найвідоміший протокол асиметричного шифрування-RSA (Rivest-Shamir-Adleman). [18, с. 127]

– Хешування: процес перетворення даних у випадкові рядки фіксованої довжини (хеш-коди), які неможливо розшифрувати. [19, с. 178] Цей метод широко використовується для зберігання паролів і перевірки цілісності даних. Найпоширеніші хеш-функції включають MD5, SHA-1iSHA-256.

Кожне з цих шифрувань має свої переваги та недоліки і використовується в залежності від вимог безпеки конкретного сценарію.

Шифрування відіграє важливу роль у захисті персональних даних у цифровому середовищі. Гребенюк А.М. визначає серед ключових причин, чому шифрування важливе для безпеки даних наступні:

– Захист конфіденційності: шифрування гарантує, що дані зберігаються у форматі, доступ до якого мають лише особи з відповідним ключем. Це гарантує конфіденційність особистої інформації, такої як паролі, фінансові дані та особисті повідомлення.

– Захист від несанкціонованого доступу: шифрування робить конфіденційні дані недоступними для сторонніх осіб, які можуть їх вкрасти або перехопити. Це важливо для захисту від кібератак і витоку даних.

– Безпека під час передачі даних: шифрування забезпечує безпеку під час передачі даних мережею, запобігаючи їх перехопленню або зчитуванню третіми особами. Це особливо важливо для онлайн-транзакцій, електронної пошти та обміну повідомленнями.

– Дотримання законів про захист даних: у багатьох країнах існують закони, які вимагають захисту персональних даних користувачів. Використовуючи шифрування, організації можуть відповідати цим вимогам і уникнути штрафів за їх недотримання.

– Захист від втрати даних: шифрування запобігає втраті даних через крадіжку або випадкову втрату пристрою. Якщо пристрій загублено або викрадено, зашифровані дані залишаються в безпеці та недоступні для зловмисників. [13]

Використання шифрування в якості інструменту для захисту персональних даних є особливо важливим в умовах зростання кіберзлочинності. Використання надійних методів шифрування може допомогти захистити дані та забезпечити їхню конфіденційність у цифровому середовищі.[7]

1.3.2 Забезпечення конфіденційності персональних даних

Конфіденційність інформації – це один з наріжних каменів забезпечення інформаційної безпеки будь-якої системи, та знаходиться на одному рівні з доступністю та цілісністю інформації. Для забезпечення конфіденційності необхідно забезпечити надійну верифікацію користувача що здійснює доступ до інформації: його авторизацію та автентифікацію.

Автентифікація – це процес перевірки особи користувача або пристрою для надання доступу до конфіденційної інформації або систем. Це важливий аспект кібербезпеки, і для забезпечення безпечного доступу існують різні типи та методи автентифікації.[8]

Окремо в теорії та практиці кібербезпеки виділяють *фактор (тип) автентифікації* – це метод перевірки особи користувача перед наданням йому доступу до системи або програми. Існує три основні типи автентифікації:

– Те, що ви знаєте: цей тип автентифікації перевіряє особу користувача на основі секретної інформації, наприклад, PIN-коду або відповіді на секретне запитання.

– Що у вас є: цей тип автентифікації перевіряє особу користувача на основі володіння фізичним об’єктом, наприклад, смарт-карткою, токеном або мобільним пристроєм.

– Хто ви є: цей тип автентифікації перевіряє особу користувача на основі фізичних характеристик, таких як відбитки пальців, розпізнавання обличчя або сканування сітківки ока.

Фактори автентифікації відносяться до різної інформації або облікових даних, що використовуються для автентифікації користувача, таких як ідентифікатор користувача, пароль, ім’я користувача та комбінація імені користувача та пароля.

Сама процедура автенифікації може бути реалізована за допомогою таких технологічних рішень:

1. Автентифікація на основі пароля – це простий і широко використовуваний метод, коли користувач надає ім’я користувача або адресу електронної пошти та пароль для доступу до системи або програми. Пароль

порівнюється з хеш-значенням, що зберігається в системі, для перевірки того, що користувач має право доступу до системи.

2. Безпарольна автентифікація забезпечує доступ до систем і додатків без необхідності вводити паролі, що позбавляє користувачів необхідності створювати і запам’ятовувати складні паролі. Замість цього особа користувача перевіряється за допомогою інших засобів, таких як біометричні дані, токени або смарт-карти.

3. 2FA/MFA (двофакторна/багатофакторна автентифікація) – це метод, який вимагає від користувачів надання двох або більше форм автентифікації для доступу до системи або програми. Це включає введення пароля та сканування відбитків пальців.

4. Єдиний вхід (SSO) дозволяє користувачам отримувати доступ до декількох додатків або систем за допомогою одного набору облікових даних для входу. Цей метод зменшує необхідність користувачам запам’ятовувати кілька паролів і спрощує процес входу в систему. SSO зазвичай використовується в корпоративних середовищах, де співробітникам потрібен доступ до різних систем і додатків.

5. Соціальна автентифікація дозволяє користувачам отримувати доступ до систем і додатків за допомогою облікових даних соціальних мереж, таких як акаунти Facebook або Google. Ця технологія спрощує процес автентифікації користувачів і може бути більш безпечною, ніж автентифікація на основі пароля, оскільки платформи соціальних мереж часто мають вдосконалені заходи безпеки.

Авторизація – це процес визначення того, чи має користувач або пристрій необхідні повноваження для доступу до певного ресурсу. Це допомагає захистити конфіденційну інформацію та системи від несанкціонованого доступу. [19, с. 42]

Визначення авторизації означає надання або відмову в доступі до певного ресурсу на основі статусу автентифікації та авторизації користувача або пристрою. Процес авторизації користувача перевіряє статус авторизації

користувача, перевіряє його облікові дані та підтверджує статус авторизації, щоб визначити, до яких ресурсів він має право доступу.

Існують різні типи авторизації, кожен з яких має свої переваги та недоліки:

1. Контроль доступу на основі ролей (RBAC) : цей тип авторизації надає доступ до ресурсів на основі ролі користувача в організації.

2. Контроль доступу на основі атрибутів (ABAC) : Цей тип авторизації надає доступ до ресурсів на основі атрибутів користувача, таких як посада, відділ або місце знаходження.

3. Обов'язковий контроль доступу (MAC) : цей тип авторизації ґрунтується на загальносистемних політиках, які визначають, які користувачі або процеси можуть отримати доступ до певних ресурсів.

4. Дискреційний контроль доступу (ДКД) : цей тип авторизації дозволяє окремим користувачам контролювати доступ до ресурсів, якими вони володіють або якими керують.

5. Контроль доступу на основі правил (RBAC) : цей тип авторизації надає доступ до ресурсів на основі набору заздалегідь визначених правил, які можуть створювати самі адміністратори або користувачі.

Методи авторизації :

1. Контроль доступу на основі ролей (RBAC) – це метод призначення користувачів на певні ролі в організації або системі та надання дозволів на основі цих ролей. Це спрощує управління дозволами користувачів і знижує ризик несанкціонованого доступу до конфіденційних даних.

2. Веб-маркери JSON (JWT) – це компактний і безпечний спосіб передачі даних між сторонами. Часто використовувані для автентифікації та авторизації користувачів у веб-додатках, JWT є самодостатніми і містять всю необхідну інформацію, усуваючи необхідність отримувати дані користувача з бази даних кожного разу, коли користувач запитує доступ.

3. SAML (Security Assertion Markup Language) – це протокол на основі XML для обміну даними автентифікації та авторизації між сторонами. SAML

часто використовується для автентифікації за допомогою єдиного входу (SSO) різних систем і додатків, дозволяє передавати інформацію про автентифікацію та авторизацію користувачів між різними доменами та додатками.

Щоб забезпечити інформаційну безпеку необхідно дотримуватися найкращих практик сертифікації та авторизації. Марія Цілина до таких практик відносить:

– Використання надійних механізмів автентифікації, такі як багатофакторна або двофакторна автентифікація для перевірки автентичності користувачів, систем і пристроїв.

– Впровадження механізмів контролю доступу, щоб гарантувати, що користувачі отримують доступ лише до необхідних їм ресурсів і виконують лише дозволені дії.

– Регулярне оцінювання та оновлення засобів контролю доступу, щоб переконатися, що вони ефективні та актуальні.

– Створення політики паролів, яка змусить користувачів створювати надійні паролі, наприклад, з використанням комбінації великих і малих літер, цифр і символів.

– Впровадження плану реагування на інциденти безпеки для своєчасного виявлення та реагування на інциденти безпеки, особливо при використанні сторонніх API та платформ хмарних обчислень.

– Використання шифрування для захисту даних як під час передачі, так і підчас зберігання, особливо коли маєте справу з конфіденційними даними.

– Регулярне навчання співробітників найкращим практикам автентифікації та авторизації, зокрема, як створювати надійні паролі та як розпізнавати фішинг-шахрайство.

– Проведення регулярного сканування для виявлення вразливостей і перевірки ефективності заходів безпеки, особливо на загальнодоступних платформах і в хмарних середовищах.[20]

1.3.3 Технічні методи захисту особистої інформації в інтернет мережі для систем з публічним доступом

Захист інформаційних систем що знаходяться в публічному доступі є комплексним явищем яке не обмежується виключно технічною реалізацією захисту у програмному продукті та його коректною експлуатацією. Важливим чинником є також налаштування та підтримка належного функціонування мережевої інфраструктури інформаційної системи. Таким чином до визначених вище методів захисту слід додати мережевий контроль.

Згідно розповсюдженої практики із забезпечення кібербезпеки [14] мережевий контроль має включати:

1. Використання і коректне налаштування мережевих брандмауерів та IDS/IPS: розгортання мережевих брандмауерів та систем виявлення та запобігання вторгненням (IDS/IPS) для моніторингу трафіку та виявлення аномалій і підозрілої активності.

Виявлення вторгнень – це процес моніторингу мережевого трафіку та його аналізу на предмет можливих ознак вторгнення, таких як спроби використання експлойтів або подій, які можуть становити пряму загрозу для мережі. Запобігання вторгненням – це процес виявлення вторгнень і запобігання виявленим подіям, зазвичай шляхом відкидання пакетів або завершення сеансів. Ці заходи безпеки пропонуються як системи виявлення вторгнень (IDS) і системи запобігання вторгненням (IPS) і включені у функціонал брандмауерів нового покоління (NGFW) як частина заходів мережевої безпеки для виявлення і зупинки потенційних інцидентів.

Системи виявлення вторгнень (IDS) та системи запобігання вторгненням (IPS) безперервно моніторять мережі, щоб виявляти та реєструвати потенційні інциденти, зупиняти інциденти та повідомляти адміністраторів безпеки. Крім того, деякі мережі використовують IDS/IPS для виявлення проблем з політикою безпеки і запобігання порушенням політики безпеки ; IDS/IPS можуть перешкодити зловмисникам збирати інформацію про мережу і повинні бути додані до політики безпеки більшості організацій. інфраструктури, оскільки вони можуть перешкодити зловмисникам збирати інформацію промережу.[10]

2. Здійснення системного моніторингу та журналювання: ведення журналів мережевої та системної активності для виявлення підозрілої активності та аналізу подій у разі інцидентів. Реєстрація та моніторинг подій безпеки – це дві частини єдиного процесу, який необхідний для підтримки безпечної інфраструктури. [11]

Будь-яка активність у середовищі, від електронних листів до входів у систему та оновлень брандмауера, вважається подією безпеки. Всі ці події реєструються (або повинні реєструватися), що дозволяє вам відстежувати все, що відбувається у вашому технологічному середовищі.

Журналювання і моніторинг подій безпеки можуть працювати тільки в тому випадку, якщо вони є частиною ефективного процесу збору та аналізу даних. Журнали безпеки часто містять величезні обсяги даних. Це робить майже неможливим ефективне виявлення загроз неозброєним оком. Це означає, що інциденти безпеки, хибні спрацьовування і дублюючу інформацію часто пропускають. Це означає, що ключ до ефективного процесу реєстрації та моніторингу безпеки полягає в здатності видаляти небажану інформацію. Зосередьтеся лише на критичних подіях, які можуть поставити під загрозу цілісність та/або доступність конфіденційної інформації. Ефективний процес збору та аналізу даних журналів повинен включати в себе інструменти для швидкого та легкого перегляду журналів аудиту на предмет виявлення критичних подій, таких як:

– Розвідка вашого оточення – супротивник вивчає ваше оточення. Це може зробити вас наступною ціллю.

– Озброєння – вторгнення у ваше середовище, коли зловмисник вирішує вжити заходів проти вашої мережі або IT-систем.

– Поширення – застосування експлойту до вразливостей у вашій мережі або IT-системі.

– Встановлення шкідливого програмного забезпечення – спостерігається, коли зловмисник змінює власну функціональність у вашому середовищі для підтримки стійкості.

– Командування та контроль – коли зловмисник отримує доступ до ваших серверів або систем, фактично беручи під контроль ваше середовище.

– Ініціювання дій – важливо визначити, що робить супротивник, і постійно підтримувати видимість супротивника.

Ведення журналів безпеки та моніторинг – це двостороння стратегія. Регулярний і безперервний аналіз даних моніторингу інцидентів необхідний для оцінки довгострокової ефективності розгорнутих систем і засобів контролю. Про всі підозрілі випадки повідомляють ключовому персоналу і негайно реагують на них, але зберігають їх централізовано для подальшого аналізу довгострокових тенденцій. [11]

РОЗДІЛ 2. ОРГАНІЗАЦІЯ СИСТЕМИ ОБРОБКИ ПЕРСОНАЛЬНИХ ДАНИХ З ПУБЛІЧНИМ ДОСТУПОМ

2.1 Функціональні та архітектурні особливості інформаційної системи

Визначення вимог є однією з важливих частин розробки інформаційної системи. Опис цієї частини дозволяє визначити конкретні функції або операції, які повинні виконуватись, включаючи операції додавання, зчитування оновлення та видалення даних у БД.

Опишемо функціональні вимоги, які є першочерговими для реалізації системи закладу медичної допомоги:

1. реєстрація та управління користувачами, що забезпечує можливість реєстрації клієнтів, співробітникв та адміністраторів за допомогою інтерфейсу, який реалізує форму реєстрації;

2. запис та оновлення даних: зберігання персональної інформації користувачів;

3. управління типами користувачів: зберігання інформації відповідно до типу користувачів.

Тепер опишемо нефункціональні вимоги, які є не менш важливими для бази даних та інтерфейсу :

1. відмовостійкість: забезпечення коректної обробки даних для забезпечення коректної роботи як у випадку коректних даних, так і у разі виникнення помилок;

2. швидкодія: забезпечення швидкого доступу до інформація, яка зберігається в базі даних під час виконання запитів;

3. масштабованість: забезпечення масштабування бази даних зі збільшенням обсягу інформації;

4. інтерфейси та взаємодія: забезпечення зручного для користувача інтерфейсу для взаємодії з базою даних за допомогою оптимізованих запитів та

зручної взаємодії через веб-інтерфейс, інтерфейси API та JDBC (Java DataBase Connectivity – з’єднання з базами даних на Java).

Враховуючи наведені вище функціональні та нефункціональні вимоги, можемо розробити базу даних, яка ефективно виконує завдання системи реєстрації клієнтів організації. В якості тематичної основи для розробки програмної реалізації ми обрали систему реєстрації пацієнтів лікарні. Це обумовлено як наочністю розподілу ролей в такій ІС, так і її зрозумілістю механізмів інтеграції в неї функціональних вимог які висуваються до системи з публічним доступом та здійснює обробку персональних даних.

Архітектурно система не має бути надто комплексною, прот має враховувати її окремі особливості:

Система базується на архітектурному підході Model-View-Controller (MVC). Цей підхід дозволяє структурувати додаток на три основні компоненти: моделі, представлення та контролери. Модель відповідає за управління даними та бізнес-логікою, представлення надає інформацію користувачеві, а контролер відповідає за обробку запитів користувачів та взаємодію між моделлю та представленням. Такий підхід ефективно відокремить логіку додатку від представлення та спростить розробку коду, тестування та супровід.

Система використовує фреймворк Spring для реалізації контролерів, ін’єкції залежностей, обробки запитів та управління бізнес-логікою. Використання Spring збільшить швидкість розробки за допомогою готових компонентів, а також спростить конфігурацію та забезпечить гнучкість і масштабованість системи.

Система взаємодіє з базою даних Microsoft SQL Server для зберігання та обробки медичної інформації. Для цього використовуються сервіси, які надаватимуть доступ до необхідної інформації та виконуватимуть необхідні операції над даними в базі даних MS SQL Server. Такий підхід забезпечить ефективне та безпечне зберігання медичних даних, а також підвищить продуктивність системи завдяки використанню різних можливостей MS SQL Server, таких як транзакції, цілісність даних, індексування та оптимізація

запитів. Для кращого розуміння системи в Додатку А розміщено схему БД розробленої системи.

2.2 Аналіз потенційних загроз та вразливостей

Оскільки у сучасному світі не існує повністю захищених систем, тому визначимо потенційні загрози та вразливості для розробленої нами системи лікарні:

1. SQL-ін’єкції: однією з основних загроз є SQL-ін’єкція, яка може виникнути через недостатню фільтрацію даних, введених користувачем, перед тим, як вони будуть використані в SQL-запиті. Зловмисник може використати цю вразливість для виконання шкідливого SQL-запиту з метою видалення, модифікації або витоку конфіденційної інформації з бази даних.

2. Недостатня аутентифікація та авторизація може призвести до несанкціонованого доступу до системи. Наприклад, слабкі паролі, неадекватні засоби ідентифікації та недостатні права доступу можуть бути вразливими місцями, які дозволяють зловмиснику отримати доступ до конфіденційної інформації або виконати несанкціоновані дії в системі.

3. Неадекватна обробка помилок: неналежна обробка помилок або ж повна її відсутність може призвести до витоку інформації про систему та сприяти таким атакам, як переповнення буферів, витік інформації та збої в роботі системи. Наприклад, відображення повідомлень про помилки з детальною технічною інформацією може допомогти зловмисникам проаналізувати вразливості системи, що значно полегшить їм доступ до системи.

4. Недостатній захист від міжсайтового скриптингу (XSS) : недостатній захист від міжсайтових сценаріїв дозволяє зловмисникам впроваджувати та виконувати шкідливий JavaScript-код на сторінках веб-додатку, що може призвести до перехоплення сеансу, витоку конфіденційної інформації або ж перенаправлення користувачів на підроблені сторінки, .

Зазначені вище загрози та вразливості можуть серйозно підірвати безпеку системи лікарні та привести до негативних наслідків, тому необхідно приділити належну увагу їх виявлені, виправлені та запобігані.

2.3 Оцінка ризиків при обробці персональних даних в інформаційній системі

На основі описаних в пункті 2.2 загроз та вразливостей виділимо ризики обробки персональних даних , які варто вирішити. Розмістимо їх в порядку спадання важливості зменшення ризиків , де спочатку розміщенні ті , які потребують першочергового вирішення а в кінці ті, які не так критично впливатимуть на роботу системи:

1. Вразливості форм вводу, такі як SQL-ін’єкції або XSS атаки. Оскільки персональна інформація зберігається в БД ,варто вжити заходів для запобігання атак спрямованих на впровадження шкідливого коду на мові програмування JavaScript або коду мовою структурованих запитівSQL

2. Вразливості, пов’язані з некоректною обробкою помилок, можуть мати серйозні наслідки для безпеки системи і конфіденційності даних. Якщо підчас взаємодії з базою даних виникне помилка (наприклад, дублікація унікальних ключів або обмеження цілісності), важливо нерозголошувати конфіденційну інформацію або деталі структури бази даних. У скрипті варто звернути увагу на рядки коду, де обробляються помилки бази даних:

```
catch (DataIntegrityViolationException e) {
        // Обробка помилки збереження
        e.printStackTrace();
        // Виведення інформації про помилку (може бути передана на сторінку)
        return "pages/register";
    } catch (Exception e) {
        // Обробка інших помилок
        e.printStackTrace();
        // Виведення інформації про помилку (може бути передана на сторінку)
        return "pages/register";
    }
```

Замість того, щоб просто відображати стек викликів або повертати клієнтській стороні повідомлення про помилки в повному обсязі, краще

обмежити інформацію, що відображається користувачеві і зберігати додаткові деталі для реєстрації та аудиту.

Якщо виникне помилка, пов’язаний з конфіденційними даними (наприклад, іменами користувачів, паролями або іншою особистою інформацією), не варто виводити цю інформацію у стек викликів або повідомлення про помилки. Замість цього варто використовувати загальні повідомлення про помилки і переконатися, що конфіденційна інформація записується тільки на стороні сервера. Крім того, помилки можна обробляти на рівні проміжного програмного забезпечення за допомогою фільтрів і властивостей, щоб забезпечити узгоджену обробку винятків у всій програмі.

3. Несанкціонований доступ до персональних даних пацієнтів. У коді системи лікарні може виникнути ризик несанкціонованого доступу до персональних даних пацієнтів. Це може бути пов’язано з вразливостями в автентифікації, авторизації та шифруванні, які можуть дозволити зловмисникам отримати доступ до конфіденційної інформації про пацієнтів. Під час розробки системи використовувався механізм автентифікації, заснований на введенні імені користувача та пароля. Тому корисно використовувати надійні паролі задля уникнення цього ризику, а також варто обмежити спроби входу, щоб ускладнити несанкціонований доступ.

2.4 Інтегровані методи захисту даних

У розробленій інформаційній системі, яка використовує БД вже реалізовано деякі з методі захисту, а саме:

1) Аутентифікація користувачів: метод login() контролера AuthController (код наведено нижче) використовується для автентифікації користувачів шляхом введення імені користувача та пароля. Введені дані звіряються з обліковим записом користувача, що зберігається в системі. Якщо дані правильні, користувач авторизується і може отримати доступ до відповідного ресурсу. Цей метод захищає систему від несанкціонованого доступу,

автентифікуючи користувача перед тим, як надати йому доступ до додаткових функцій системи.

Метод login() контролера AuthController

```
@PostMapping("/login")
   public String login(@RequestParam String inputUsername, @RequestParam String password, Model model, HttpSession session) {
      try {
         User user = userService.authenticate(inputUsername, password);
         if (user != null) {
            // Встановлення об’єкту користувача у сесію лише, якщо його там ще немає
            if (session.getAttribute("user") = = null) {
               session.setAttribute("user", user);
            }
            // Перевірка типу користувача та перенаправлення відповідно
            switch (user.getUserType()) {
               case "patient":
                  return "redirect:/patients/main";
               case "administration":
                  return "redirect:/admin/main";
               case "doctor":
                  // Отримання doctorId під час аутентифікації лікаря
                  int doctorId = doctorService.getDoctorByUserId(user.getUserId()).getDoctorId();
                  // Встановлення атрибута doctorId у сесії
                  session.setAttribute("doctorId", doctorId);
                  return "redirect:/doctors/main";
               default:
                  model.addAttribute("error", "Невідомий тип користувача");
                  return "pages/login";
            }
         } else {
            model.addAttribute("error", "Невірне ім’я користувача або пароль");
            return "pages/login";
         }
      } catch (Exception e) {
         // Обробка помилок або відловлення винятків
         e.printStackTrace();
         model.addAttribute("error", "Виникла помилка при авторизації:" + e.getMessage());
         return "pages/login";
      }
   }
```

2) Збереження облікових записів користувачів: метод registerUser() контролера AuthController (код наведено нижче) відповідає за зберігання

облікових записів користувачів у системі. Він безпечно зберігає інформацію про користувача, таку як ім’я, пароль і тип користувача.

Метод registerUser() контролера AuthController:

```
@PostMapping("/register")
   public String registerUser(@ModelAttribute ("userModel") User user,
                              @ModelAttribute("patientModel") Patient patient,
                              @ModelAttribute("doctorModel") Doctor doctor,
                              @ModelAttribute("administratorModel") Administration
administrator,
                              @RequestParam("userType") String userType,
                              HttpSession session) {

      try {
         // Перевірка наявності імені користувача
         if (userService.userNameExists(user.getUserName())) {
            return "pages/register";
         }

         // Збереження користувача
         userService.saveUser(user);

         // Відповідно до вибору типу користувача зберігаємо відповідний об’єкт
(Пацієнт, Лікар, Адміністратор)
         switch (userType) {
            case "patient":
               // Прив’язка користувача до типу пацієнт після збереження
               patient.setUser(user);
               patientService.savePatient(patient);
               break;
            case "doctor":
               // Прив’язка користувача до типу лікар після збереження
               doctor.setUser(user);
               doctorService.saveDoctor(doctor);
               break;
            case "administrator":
               // Прив’язка користувача до типу адміністратор після збереження
               administrator.setUser(user);
               administrationService.saveAdministration(administrator);
               break;
            default:
               // Невідомий тип користувача
               return "pages/register";
         }


         // Встановлення об’єкта користувача у сесію
         session.setAttribute("user", user);
```

```
            return "redirect:/auth/login";
        }catch (DataIntegrityViolationException e) {
            // Обробка помилки збереження
            e.printStackTrace();
            // Виведення інформації про помилку (може бути передана на сторінку)
            return "pages/register";
        } catch (Exception e) {
            // Обробка інших помилок
            e.printStackTrace();
            // Виведення інформації про помилку (може бути передана на сторінку)
            return "pages/register";
        }
    }
```

3) Встановлення об’єкта користувача у сесію: методи login() та registerUser() контролера AuthController встановлюють об’єкт користувача в сесію. Це дозволяє системі зберігати статус автентифікації користувача та отримувати доступ до відповідних ресурсів під час сеансу.

4) Використання об’єктно-реляційного відображення (ORM): користуючись технологіями ORM, такими як Hibernate або Spring Data JPA, можна взаємодіяти з базою даних через Java-об’єкти, забезпечуючи автоматизоване управління SQL-запитами та захист від SQL-ін’єкцій.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ КОМПОНЕНТІВ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ СИСТЕМИ ЩО ЗДІЙСНЮЄ ОБРОБКУ ПЕРСОНАЛЬНИХ ДАНИХ

3.1 Проектування політики безпеки інформаційної системи

Проектування системи захисту інформаційної системи (ІС) заснована на методологічних та організаційних аспектах, що забезпечують її безпеку. Одним з ключових елементів є розробка політики безпеки, яка включає розуміння потенційних вразливостей, загроз та способів їх запобігання.

Політика безпеки інформаційної системи є основоположним документом і може відрізнятися своїм змістом і деталями в залежності від об’єкта, що захищається і умов експлуатації. Наприклад, у випадку систем публічного доступу, які обробляють особисті дані, політика безпеки включає як організаційні, так і адміністративні рішення в області інформаційної безпеки для програмних продуктів.

До основних компонентів політики безпеки відносяться:

1. Загальні положення: визначення термінів, цілей та сфер застосування.

2. Організаційна структура інформаційної безпеки: визначення та інтерпретація моделі для наслідування та повноважень користувачів інформаційної системи.

3. Процедури управління інформаційною безпекою: обов’язки і процедури співробітників при виконанні своїх обов’язків і при виявленні порушень інформаційної безпеки.

4. Принципи застосування інформаційної безпеки: методи захисту, включаючи технічні та програмні, для боротьби з конкретними загрозами.

5. Політика звітності та ведення журналу: реєстрація виявлених загроз та порушення політики безпеки.

Політика безпеки, представлена в Додатку Б, містить в собі компоненти, які безпосередньо пов’язані з обробкою персональних даних і публічним

доступом до них. Кожен з компонентів спрямований на забезпечення конфіденційності, цілісності та доступності даних. Детальніше розглянемо кожен з компонентів, які забезпечують захист персональних даних так контролюють доступ до інформації.

Найбільш критично важливим, особливо в галузі охорони здоров'я та сфері персональних даних, компонентом політики безпеки є захист конфіденційності даних. Ігнорування та недотримання вимог щодо конфіденційності може мати серйозні наслідки як для окремих користувачів, так і для організацій, які обробляють надані їм дані.

Шифрування, авторизація та автентифікація є найбільш широко використовуваними аспектами забезпечення конфіденційності даних.

Шифрування – це основний механізм захисту даних від несанкціонованого доступу. Використання наскрізного шифрування та алгоритмів шифрування під час зберігання даних має ключові переваги:

1. Унеможливлення несанкціонованого доступу: навіть якщо зловмисник має доступ до зашифрованих даних, розшифрувати і використати дані без відповідного ключа він не зможе. Це значно знижує ризик витоку конфіденційної інформації.

2. Підвищення довіри користувачів: користувачі віддають перевагу організаціям, які забезпечують високий рівень захисту персональних даних.

Авторизація та автентифікація є основними механізмами контролю доступу до даних, які забезпечують доступ до даних лише авторизованим користувачам за допомогою 2-факторної автентифікації та використанням складних паролів. Назвемо їх основні переваги:

1. Запобігання несанкціонованого доступу: використання складних паролів і їх регулярна заміна знижує ризик злому облікових даних. Двофакторна автентифікація забезпечує безпеку, просячи користувача підтвердити свою особу за допомогою другого елемента (наприклад, одноразового коду з SMS або електронної пошти).

2. Мінімізація людських помилок: механізм автоматичної перевірки складності пароля та вимоги до періодичної зміни пароля допомагають користувачам створювати та використовувати надійні паролі, що знижує ризик їх зламу або підбору.

3. Контроль доступу: авторизація дозволяє чітко вказати, хто має доступ до яких даних, що дозволяє лише певним користувачам отримувати доступ до конфіденційної інформації, мінімізуючи ризик внутрішніх загроз.

Наступний компонент, є не менш важливим для систем з публічним доступом є забезпечення цілісності даних . Забезпечення цілісності даних є надзвичайно важливим для будь-якої організації, оскільки неправильні або пошкоджені дані можуть мати серйозні наслідки для медичної організації, наприклад неправильні медичні рішення та порушення законодавчих вимог. Це досягається завдяки захисту від вразливостей та використання програмних заходів інформаційної безпеки.

Переваги захисту від вразливостей:

Кожна вразливість в програмному забезпеченні може бути використана зловмисниками для отримання несанкціонованого доступу або модифікації даних. Регулярні оновлення забезпечуватимуть запобігання експлуатації вже відомих вразливостей.

Оновлення програмного забезпечення не тільки виправляють помилки, але й допомагають запобігти збоям та втраті даних, покращуючи загальну стабільність системи та сумісність з іншими компонентами.

Програмні заходи інформаційної безпеки:

Інструменти виявлення та реагування на загрози в режимі реального часу допомагають своєчасно виявляти та усувати потенційні загрози, які можуть вплинути на цілісність даних.

Автоматизовані системи виявлення загроз скорочують час, необхідний для реагування на інциденти безпеки, і мінімізують негативний вплив на дані.

Використання передових технологій, таких як машинне навчання та нейронні мережі, допоможе ефективніше виявляти аномалії та потенційні

загрози, які можуть бути пропущені традиційними методами(сигнатурний аналіз, аналіз на основі правил та ін.).

Ще один важливий компонент політики безпеки, який гарантує безперебійну роботу системи та доступ авторизованих користувачів до даних без стороннього втручання і зтримок є забезпечення доступності даних.

Це особливо важливо для медичних закладів, де швидкий доступ до даних має вирішальне значення для життя та здоров'я пацієнтів. Давайте детальніше розглянемо, чому це важливо, і які заходи, включені в цей компонент.

1. Регулярна зміна паролів: Якщо облікові дані будь-якого користувача скомпрометовані (наприклад, за допомогою фішингу або інших методів атаки), періодичні зміни пароля може запобігти отримати довгостроковий доступу до системи зловмиснику. Якщо система використовує механізми регулярної зміни пароля, користувачам доведеться оновлювати свої облікові дані, зменшуючи ймовірність злому або повторного використання.

2. План відновлення після інцидентів: План відновлення може бути використаний організаціями у разі виникнення інцидентів (кібератаки, технічні збої тощо), що дозволяє швидко реагувати і мінімізувати системні збої. Медичні працівники не можуть надавати тривалі перерви, оскільки це може призвести до затримок у наданні медичних послуг. Планування відновлення забезпечує безперервність бізнес-процесів, особливо доступу до медичних даних.

План відновлення включає в себе заходи щодо резервного копіювання та відновлення даних, які мінімізують ризик втрати важливої інформації.

Політика звітності та ведення журналу інцидентів безпеки є важливою складовою всієї політики безпеки. Він забезпечує систематичний підхід до реєстрації, аналізу та звітування про події безпеки. Політика спрямована на виявлення та усунення недоліків безпеки, забезпечення дотримання нормативних вимог та мінімізацію ризиків для системи та даних

Кожна подія безпеки (спроби несанкціонованого доступу, зміни конфігурації системи та виявлення вразливостей та ін.), незалежно від її типу та ступеня тяжкості, повинна бути зареєстрована у спеціальному журналі подій безпеки.

Кожен інцидент повинен бути обов’язково проаналізований, що дозволяє:

- Визначити причину події, що дозволить розробити ефективні заходи їх усунення
- Виявити вразливості, протидія яким допоможе вдосконалити систему.
- Задокументувати результати аналізу для подальшого використання. Це дозволить створити базу знань, що допоможе вирішити подібні випадки в майбтньому.

3.2 Реалізація технічних заходів безпеки в системі (шифрування, верифікація даних, безпечні з’єднання).

Оскільки захист спроектованої інформаційної системи повинен відповідати розробленій політиці безпеки (Додаток Б) варто реалізувати технічні заходи безпеки захисту інформації .

Основна виявлена слабкість системи, яка значно впливає на безпеку в системі це відсутність перевірки пароля користувачів під час реєстрації. Варто зазначити , що дійсно можливість створення паролю зручного для користувача, який до того ж не потрібно придумувати відповідно до описаних параметрів, є позитивним аспектом для зручності користувачів, але разом з цим це може критично вплинути на безпеку облікових записів системи. Надійний пароль – це перша лінія захисту від несанкціонованого доступу до облікового запису. Часто для багатьох систем користувачі використовують простий і короткий пароль, який легко запам’ятати і можна швидко ввести у відповідне поле, такі паролі надто слабкі і можуть стати легкою мішенню для хакерів або зловмисників, які можуть зламати обліковий запис і отримати доступ до конфіденційної інформації. Варто не забувати , що такі слабкі паролі не є надійними і легко отримуються простим перебором паролів. Зловмисник може

атакувати слабкий пароль, використовуючи словник або список відомих паролів. Саме тому перевірка пароля під час реєстрації може допомогти запобігти таким атакам, вимагаючи від користувачів використання більш складних паролів.

Для того аби захистити облікові записи та дані користувачів для системи було введено систему перевірки пароля, яка повинна відповідати наступним вимогам:

1. Пароль повинен містити щонайменше 8 символів.
2. Пароль повинен містити принаймні одну цифру.
3. Пароль повинен містити принаймні одну велику літеру.
4. Пароль повинен містити принаймні одну маленьку літеру.
5. Пароль повинен містити принаймні один спеціальний символ.

Тепер під час реєстрації користувачів система в реальному часі аналізує введений пароль користувача і виводить під полем інформаційне повідомлення, якщо введений користувачем пароль не відповідає одному з описаних вище пунктів вимог до пароля. На Рис. 3.1 зображено інформаційне повідомлення яке з’являється якщо ввести лише 3 цифри в поле для пароля. Нижче наведено механізм перевірки пароля написаний на JavaScript:

```
<script>
// Перевірка складності пароля при введенні
    document.getElementById(‘userPassword’).addEventListener(‘input’, function () {
        // Отримуємо значення введеного пароля
        var password = this.value;
        // Отримуємо елемент для відображення вимог до пароля
        var requirements = document.getElementById(‘passwordRequirements’);
        // Перевіряємо наявність великої літери у паролі
        var hasUpperCase = /[A-Z]/.test(password);
        // Перевіряємо наявність маленької літери у паролі
        var hasLowerCase = /[a-z]/.test(password);
        // Перевіряємо наявність цифри у паролі
        var hasNumbers = /\d/.test(password);
        // Перевіряємо наявність спеціальних символів у паролі
        var hasSpecialChars = /[!@#$%^&*()_+\-=[\]{};’:"\\|,.<>/?]/.test(password);
        // Перевіряємо довжину пароля (мінімум 8 символів)
        var isLengthyEnough = password.length >= 8;
        // Ініціалізуємо повідомлення про вимоги
        var message = "";
```

```
            // Додаємо повідомлення, якщо пароль не містить щонайменше 8 символів
            if (!isLengthyEnough) {
                message += "Пароль повинен містити щонайменше 8 символів. ";
            }
            // Додаємо повідомлення, якщо пароль не містить принаймні одну велику літеру
            if (!hasUpperCase) {
                message += "Пароль повинен містити принаймні одну велику літеру. ";
            }
            // Додаємо повідомлення, якщо пароль не містить принаймні одну маленьку літеру
            if (!hasLowerCase) {
                message += "Пароль повинен містити принаймні одну маленьку літеру. ";
            }
            // Додаємо повідомлення, якщо пароль не містить принаймні одну цифру
            if (!hasNumbers) {
                message += "Пароль повинен містити принаймні одну цифру. ";
            }
            // Додаємо повідомлення, якщо пароль не містить принаймні один спеціальний символ
            if (!hasSpecialChars) {
                message += "Пароль повинен містити принаймні один спеціальний символ. ";
            }
            // Відображаємо повідомлення про вимоги до пароля
            requirements.textContent = message;
        });
    });

</script>
```

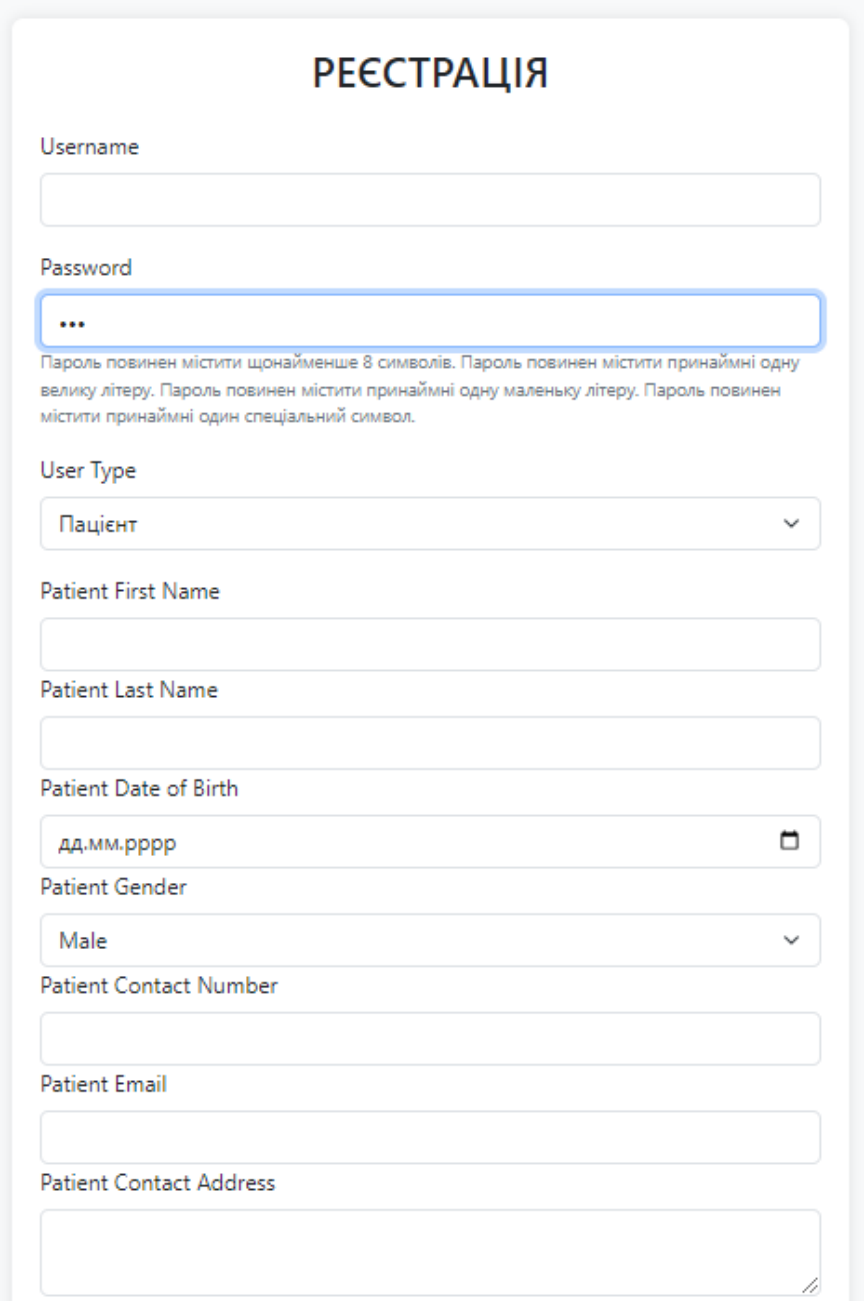

Рис. 3.1. Інформаційне повідомлення під час введення пароля

Ще одним важливим заходом безпеки є двохфакторна авторизація(2FA), яка може стати додатковим шаром захисту системи.

Реалізація двохфакторної авторизації допомагає зміцнити захист облікових записів. В цьому випадку 2FA виступатиме додатковою перешкодою для зловмисників, які отримають доступ до пароля одного з користувачів. Навіть, якщо пароль вже відомий зловмиснику, без другого фактора ніхто не зможе увійти в обліковий запис.

Зазвичай 2FA передбачає надсилання одноразового коду з обмеженням існування в часі або ж використання фізичного пристрою, що зменшує ризик проведення фішингових атак та соціальної інженерії , а також робить більш ефективний захист від них.

Також 2FA підвищує довіру користувачів, оскільки, знаючи, що для втручання зловмисникам потрібно не лише знати пароль, але й мати доступ до додаткового фактора, користувачі можуть бути впевненішими у захисті своїх облікових записів.

Хоча 2FA є важливим засобом безпеки, його не завжди можна легко реалізувати в маленькій системі.

Основні причини чом варто відмовитись від реалізації 2FA в маленькій системі:

– Вартість: деякі методи 2FA можуть вимагати додаткових витрат на інфраструктуру, таких як SMS-повідомлення та/або фізичні пристрої.

– Складність для користувачів: для деяких користувачів може бути складно та/або незручно додавати додаткові елементи аутентифікації.

– Технічні обмеження: деякі маленькі системи можуть мати обмеження технічної підтримки для певних методів 2FA.

– Масштабованість: для невеликих систем може бути важко адаптувати процес автентифікації до вимог 2FA без значного збільшення складності системи.

Спираючись на описані вище складнощі було прийнято рішення відмовитись від реалізації двохфакторної авторизації в розробленій системі.

Для подальшого забезпечення безпеки, мінімізації виникнення вразливостей та безперебійної роботи сайту варто забезпечити для системи регулярні оновлення програмного забезпечення. Для цього розробникам, які будуть проводити подальшу підтримку системи варто:

– Визначити часовий графік випуску оновлень.

– Автоматизувати процес оновлення ПЗ

– проводити регулярні аудити безпеки для виявлення вразливостей та проблем безпеки

3.3 Процедури контролю доступу до персональних даних.

Ефективні процедури контролю доступу в є ключовим фактором забезпечення безпеки та конфіденційності інформації. Вони створюють основу для захисту цінних даних, запобігаючи несанкціонованому доступу та мінімізуючи ризик порушення безпеки.

Для розробленої системи було введено наступні прицидури контролю:

1. Аутентифікація і авторизація:

Система підтримє парольну автентифікацію. Для її проходження користувач повиннен ввести свій ідентифікатор користувача, який є іменем користувача, яке ідентифікує конкретного користувача, та ввести пароль,який при реєстрації повинен відповідати визначеним вимогам в пункті 3.2. Після чого система повинна перевірити відповідність ідентифікатора та пароля перед наданням доступу.

2. Жорстке обмеження доступу до системи:

Користувачі не можуть отримати доступ до системи або її функцій, якщо вони не пройшли процедуру авторизації. Усі користувачі повинні пройти аутентифікацію перед тим, як отримати доступ до будь-яких функцій або даних в системі.

Для ситеми було визначено чотири основні типи ролей для користувачів: адміністратори першого рівня, адміністратори другого рівня, привілейовані користувачі, які є співробітниками організації, що використовує систему, а також звичайні користувачі, клієнти організації. Для кожної з ролей визначено свої особливості, такі як певні права та обмеження, які регулюватимуть взаємодію користувачів з даними та в цілому з системою. Опишемо детальніше які саме ролі та права було визначено для кожного з користувачів:

Адміністратори першого рівня (адміністратори інформаційної системи) мають найвищий рівень доступу в системі, тому вони можуть виконувати всі критичні операції, необхідні для обслуговування та управління системою, а також забезпечення безпеки.

Ролі та права:

– Доступ до даних клієнтів: мають повний доступ до всіх даних клієнтів, який влючає в собі можливість перегляду інформації, редагування даних клєнтів та видалення інформації клієнтів з БД (лише за умови відповідного запиту від користувача).

Цей рівень доступу дозволяє адміністраторам першого рівня ефективно управляти даними клієнтів і забезпечувати їх актуальність і цілісність.

– Робота з системою: мають повний доступ до всіх системних налаштувань, що включає в себе конфігуацію сервера, БД та мережевих налаштувань. Адміністратори першого рівня можуть змінювати всі технічні параметри системи, що забезпечує її стабільну роботу та відповідність вимогам безпеки.

– Робота з даними співробітників: мають повний доступ до всіх даних співробітників, який включає в собі можливіть перегляду інформації про співробітників, редагування даних та видалення інформації співробітників з БД (за умови відповідного запиту від співробітника або ж у випадку звільнення конкретного працівника). Адміністратори першого рівня можуть керувати персональними даними співробітників організації, що дозволить підтримувати інформацію в актуальному стані і вирішувати проблеми управління персоналом.

– Створення нових користувачів з відповідними ролями: мають можливість створення облікових записів нових користувачів в системі з відповідними ролями та правами доступу в залежності від їх функціональних обов’язків та потреб роботі з ситемою. Адміністратори першого рівня стежать за тим, щоб кожен створений користувач мав відповідні права доступу,

відповідні ролям і функціональним обов’язкам в організації, що дозволить гнучко управляти доступом до ресурсів системи.

Обмеження: адміністратори 1-го рівня не мають обмежень на доступ до ситеми і даних.

Адміністратори другого рівня (адміністратори персональних даних) покликані забезпечити належний контроль за дотриманням національного законодавства та міжнародних стандартів з обробки персональних даних які зберігаються в базі даних.

Ролі та права:

– Доступ до даних клієнтів: мають доступ до всіх даних клієнтів, що включає в себе перегляд інформації про клієнтів їх персональних даних та редагування даних чи додання нових записів. Здатність перегляду та редагування даних клієнтів дозволяє адміністраторам швидко отримувати доступ до інформації про клієнтів та їхні потреби, що допомагає в розумінні індивідуальних вимог клієнтів та наданні персоналізованої підтримки.

– Робота з системою: мають обмежений доступ до налаштвань системи, що обмежується управлінням БД, що являє собою створення нових записів в таблицях БД, оновленя полів та налаштванням системи.

– Робота з даними співробітників: мають доступ до всіх даних співробітників,який включає в собі можливість перегляду інформації про співробітників, їх персональних даних, інформацію про посаду, та редагування даних. Маніплювання даними співробітників дозволяє ефективно керувати даними про співробітників, а також дозволяє уникнути можливих проблем пов’язаних з втратою даних або порушенням конфіденційності.

Обмеження:

– Адміністратори другого рівня не мають доступу до налаштувань фізичного сервера та мережевих компонентів системи.

– Адміністратори другого рівня мають повний доступ до перегляду та редагування даних клієнтів і співробітників, але немають прав на видалення

цих даних. Це гарантує, що важлива інформація не буде видалена випадково або навмисно.

Привілейовані користувачі (співробітники):

Ролі та права:

– Доступ до даних клієнтів: мають обмежений доступ до даних клієнтів, що включає в собі можливість перегляду та редагування лише тих даних,які стосуються їхньої роботи. Обмеження доступу до даних клієнтів, що забезпечує приватність користувачів та знижє ризик витоку даних.

– Робота з системою: мають доступ до внутрішніх ресурсів та інструментів,які потрібні для виконання посадових обов'язків. Обмеження доступу до системи допомогає уникнти несанціонованого доступу до конфіденційних даних та запобігає можливісті несанкціонованої зміни параметрів, що може привести до порушення безпеки даних.

– Доступ до власних даних: мають повний доступ до своїх персональних даних, що включає в собі можливість їх перегляду та редагування. Доступ до песональних даних дозволяє підтримувати актуальність інформації в системі та забезпечити обробку і зберігання коректної інформації .

– Взаємодія з клієнтами: мають можливість взаємодіяти з клієнтами через систему. Можливість взаємодії з клієнтами через систему обмежена функціональними можливостями допомагає забезпечити безпеку персональних даних клієнтів,що дозволяє отримати доступ до конфіденційних даних клієнтів задля виконання певних завдань визначених посадовими обов'язками лише за умови мінімальної кількісті необхідних прав.

Обмеження:

– Привілейовані користувачі не мають доступ до налаштувань системи. Це запобігає можливості зміни конфігурації системи, що може вплинути на безпеку даних та правильність роботи системи.

– Привілейовані користувачі не мають прав на перегляд, редагування та видалення даних інших користувачів. Це забезпечує конфіденційність та

цілісність персональниххх даних інших користувачів системи та запобігає можливості витоку інформації.

– Доступ привілейованих користувачів до даниз клієнтів обмежений їхніми посадовими обов’язками. Це мінімізує ризик несанкціонованого доступу до персональних даних .

Звичайні користувачі (клієнти) мають найменших рівень доступу в ситемі.

Ролі та права:

– Доступ до своїх даних: мають повних доступ до своїх даних, що включає в собі перегляд та редагування інформації. Повних доступ до власних даних дозволяє користувачам контролювати інформацію, яка використовуєть та розголошється, що допомагає підтримувати конфіденційність і актуальність даних,в том числі і персональних.

– Взаємодія з системою: мають можливість взаємодіяти з ситемою через інтерфейс окристувача, що включає в собі створеня запитів до співробітників та отримання підтримки. Можливість взаємодіяти з системою через користувальницький інтерфейс дозволяє користувачам отримувати допомогу і підтримку, що допомагає швидко вирішувати будь-які питання або неполадки, що виникають у користувачів щодо їх персональних даних або функціонування системи, а також інтерфейс дозволяє відправляти запити на послуги в міру необхідності.

Обмеження:

– Звичайні користувачі не маютьдоступу до персональних даних інших користувачів. Обмеження доступу до даних інших користувачів і співробітників гарантує, що кожен користувач отримує доступ тільки до своїхданих, що забезпечує збереження конфіденційності персональної інформації та запобігає несанкціонованому доступу до даних інших людей.

– Звичайні користувачі не мають доступу до налаштувань системи та до адміністративних функцій. Заборона доступу до налаштвань системи запобігає

внесені змін в конфігурацію системи, що забезпечить конфіденційність, цілісність та достпність даних.

– Звичайні користувачі мають обмежений доступ до функцій системи відповідно до ролі клієнта. Обмежений доступ до системних функцій відповідно до ролей клієнтів дозволяє користувачам отримувати доступ тільки до тих ресурсів, які необхідні їм для виконання своїх обов’язків, що зменшує ризик несанкціонованого доступу та порушення конфіденційності даних.

Описані вище ролі та права користувачів забезпечують структурований і безпечний підхід до управління доступом користувачів до системних ресурсів. Різні рівні доступу ефективно розподіляють повноваження та відповідальності, що зменшує ризик несанкціонованого доступу та забезпечує захист конфіденційної інформації.

Адміністратори першого рівня мають найширший спектр дозволів, включаючи повний доступ до даних клієнтів, співробітників та конфігурації системи. Адміністратори другого рівня мають обмежений доступ до системних налаштувань і не можуть видаляти дані. Привілейовані користувачі (співробітники) можуть отримувати доступ до даних, що стосуються їхніх обов’язків,та взаємодіяти з клієнтами через систему. Звичайні користувачі (клієнти) мають доступ лише до власних даних.

3. Регулярні навчання та інструктажі: забезпечують зниження ризику втрати даних через необізнаність працівників організації, що використовує ситему. Навчання повинні проводитися з визначеною регулярністю як у формі онлайн-курсів, так і у формі лекцій або тренінгів з використанням відеоматеріалів та презентацій. Надання чітких інструкцій щодо використання системи може допомогти вамуникнути помилок та неправильного використання, які можуть призвести до потенційних загроз безпеці.

Додаткові тренінги можуть бути організовані для спеціалізованих груп співробітників, таких як адміністратори системи, розробники програмного забезпечення або відділ інформаційної безпеки.

Ці тренінги можуть включати більш глибокий аналіз тем, пов’язаних з безпекою, та специфічні методи захисту від ризиків.

ВИСНОВКИ

Забезпечення належного рівня захищеності персональних даних є комплексним процесом який має гармонійно суміщати різні аспекти інформаційної безпеки. У відповідності до вимог міжнародних стандартів та вітчизняних нормативно-правових актів, воно має ґрунтуватись на розробці та імплементації політики інформаційної безпеки інформаційної системи яка покликана забезпечити конфіденційність, цілісність та доступність персональних даних.

В процесі дослідження було встановлено, що для забезпечення високого рівня довіри користувачів та своєчасного і ефективного реагування на сучасні виклики та загрози необхідно належним чином реалізувати заходи організаційного (навчання та інструктаж персоналу, розуміння ними своїх обов’язків та відповідальності за порушення правил поводження з персональними даними), технічного (використання відповідного апаратного забезпечення для дотримання заходів безпеки) і програмного (належне проектування програмного забезпечення, методів маніпулювання даними та реагування на виявлені загрози) характеру.

Розроблена та реалізована в практичній площині модель інформаційної системи обробки персональних даних з публічним доступом дозволила, на основі проведеного аналізу її функціональних та архітектурних особливостей, виокремити потенційні вразливості та загрози безпеці персональних даних. Виконана оцінка ризиків надала можливість обґрунтувати пріоритетні напрямки протидії загрозам які мають бути враховані при розробці стратегій захисту персональних даних (забезпечення захисту даних від викрадення, надійність системи авторизації користувачів).

Для протидії виявленим загрозам було розроблено політику інформаційної безпеки в якій були систематизовані заходи із забезпечення безпеки, визначені ролі та обов’язки користувачів інформаційної системи,

встановлені процедури обробки даних, висвітлені механізми моніторингу інцидентів і реагування на них.

Відповідно до розробленої політики інформаційної безпеки у функціонування розробленої інформаційної системи було імплементовано технічні заходи безпеки і процедури контролю доступу, проведено їх тестування та валідацію. Реалізація запропонованих технічних заходів продемонструвала складність забезпечення коректної роботи різних компонент системи спрямованих на захист персональних даних як цілісної системи.

Запропоновані та розглянуті в роботі підходи до організації захисту персональних даних в системах з публічним доступом є лише першим кроком у розробці комплексного підходу до формулювання та втілення в життя політики інформаційної безпеки. Вони потребують подальшого доопрацювання та удосконалення. До перспективних напрямків подальших досліджень слід віднести: удосконалення технічних та програмних засобів забезпечення конфіденційності персональних даних (розширення засобів багатофакторної авторизації), покращення підходів до забезпечення цілісності даних (розробка та впровадження надійних систем потокового шифрування даних, деталізація механізму розподілу ролей користувачів інформаційної системи тощо).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Галінкіна, В. С. (2024). Основні принципи обробки та захисту персональних даних. *Науковий вісник Ужгородського національного університету. Серія: Право*, *1*(81), 111-115. URL: https://visnyk-juris-uzhnu.com/wp-content/uploads/2024/03/19.pdf.

2. Закон України "Про захист персональних даних". № 2297-VI від 01.06.2010. Відомості Верховної Ради України. 2010

3. Закон України "Про захист інформації в інформаційно-комунікаційних системах". № 80/94-ВР від 05.10.1994. Відомості Верховної Ради України. 1994

4. Загальний регламент захисту даних (GDPR). GlobalLogic. URL: https://www.globallogic.com/ua/gdpr/.

5. GDPR: загальний регламент щодо захисту даних. UHY Prostir LLC. URL: https://www.uhy-prostir.com/blog/zagalnij-reglament-shhodo-zahistu-danih/.

6. ISO 27001 – сертифікація інформаційної безпеки. URL: https://www.dqsglobal.com/uk-ua/sertifikujte/sertifikaciya-iso-27001

7. Роль шифрування у захисті особистої інформації. mindscope.biz.ua. URL: https://mindscope.biz.ua/rol-shyfruvannya-u-zahysti-osobystoyi-informacziyi/.

8. У чому різниця між аутентифікацією та авторизацією?. INTROSERV. URL: https://introserv.com/ua/blog/u-chomu-rizniczya-mizh-autentifikaczieyu-ta-avtorizaczieyu/.

9. Що таке https? І навіщо він потрібен кожному сайту. hosting UKRAINE. URL: https://www.ukraine.com.ua/uk/blog/seo-optimization/chto-takoe-https-i-zachem-on-nuzhen-kazhdomu-sajtu.html#%20Навіщо%20потрібно%20використовувати%20HTTPS?.

10. What is IDS and IPS?. juniper.net. URL: https://www.juniper.net/us/en/research-topics/what-is-ids-ips.html.

11. WHAT IS SECURITY LOGGING AND MONITORING?. BitLyft. URL: https://www.bitlyft.com/resources/what-is-security-logging-and-monitoring#what-is-security-logging-monitoring.

12. Шеремет О.П. Забезпечення захисту персональних даних у системах з публічним доступом / Студентські наукові дискусії поза форматом: матеріали XI Міжнародної наукової конференції (м. Івано-Франківськ, 11 квітня 2024 року). Івано-Франківськ : Редакційно-видавничий відділ ЗВО «Університет Короля Данила». 2024. С. 486-489.

13. Гребенюк А.М. Основи управління інформаційною безпекою: навч. посібник / А.М. Гребенюк, Л.В. Рибальченко. Дніпро: Дніпроп. держ. унт внутріш. справ, 2020. – 144 с.

14. Nichols, L. (2024). Cybersecurity Architect's Handbook: An end-to-end guide to implementing and maintaining robust security architecture.

15. Almeida Teixeira, G., Mira da Silva, M. and Pereira, R. (2019), "The critical success factors of GDPR implementation: a systematic literature review", *Digital Policy, Regulation and Governance*, Vol. 21 No. 4, pp. 402-418. https://doi.org/10.1108/DPRG-01-2019-0007

16. Rohan R, Pal D, Hautamäki J, Funilkul S, Chutimaskul W, Thapliyal H. (2023), "A systematic literature review of cybersecurity scales assessing information security awareness." Heliyon. Vol. 9 No 3 DOI: https://doi.org/10.1016/j.heliyon.2023.e14234

17. David Reinsel, John Gantz, John Rydning (April 2017) Data Age 2025: The Evolution of Data to Life-Critical URL: https://www.seagate.com/files/www-content/our-story/trends/files/Seagate-WP-DataAge2025-March-2017.pdf

18. Інформаційна безпека: Навчальний посібник / Ю. Я. Бобало, І. В. Горбатий, М. Д. Кіселичник, та ін. Львів : Видавництво Львівської політехніки, 2019. 580 с.

19. Козюра, В. Д., Хорошко, В. О., Шелест, М. Є., Ткач, Ю. М., Балюнов, О. О. (2020). Захист інформації в комп'ютерних системах: підручник. *Ніжин: ФОП Лук'яненко ВВ, ТПК «Орхідея»*.

20. Цілина, М. (2022). Зарубіжний досвід забезпечення захисту конфіденційної інформації. *Український журнал з бібліотекознавства та інформаційних наук*, (9), 22-32.

ДОДАТКИ

Додаток А. Схема бази даних системи що здійснює обробку персональних даних

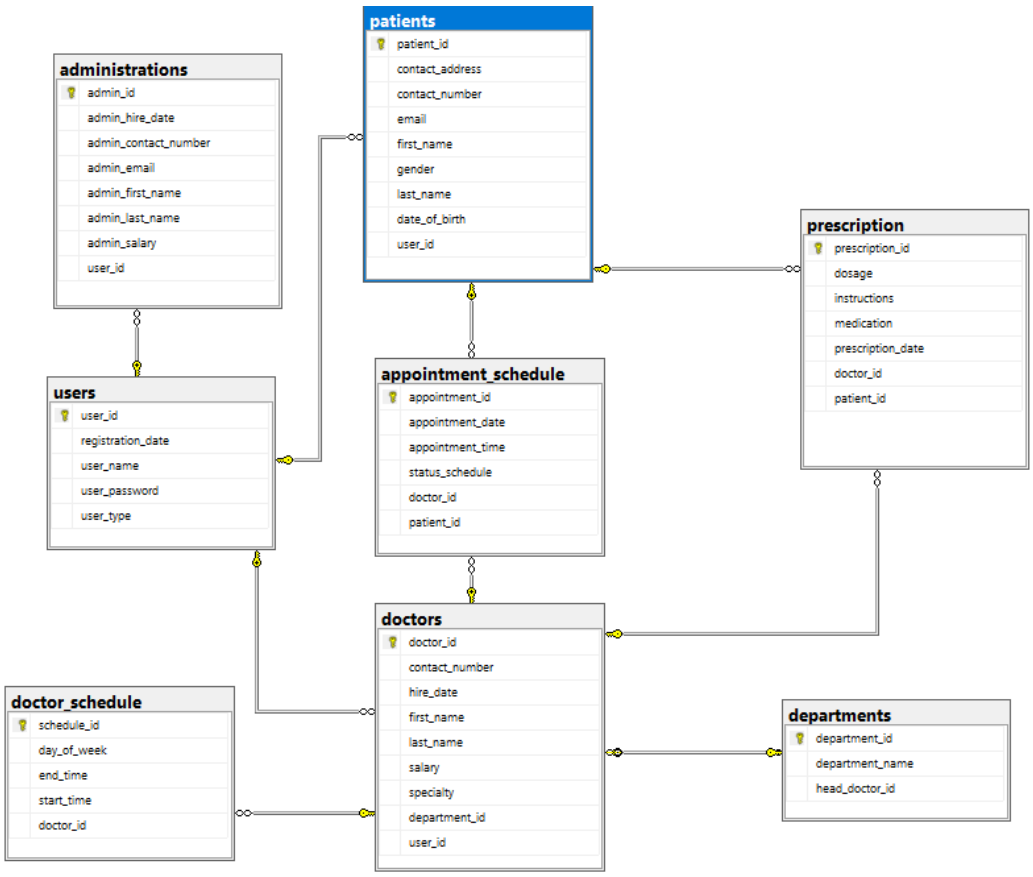

Рис. А.1. Схема бази даних для системи медичного закладу

Для опису структури опишемо таблиці та їх поля, в яких будуть зберігатися дані:

1. Таблиця users для даних користувачів:

• user_id – поле для збереження унікального ідентифікатора користувача, повинно набувати цілочисельних значень, які генеруються автоматично системою для гарантування унікальних значень користувачів. Це поле повинно бути первинним ключем який буде зв’язувати таблицю з іншими;

• user_name – поле для імені користувача, яке буде використовуватись для авторизації в системі. Його значення не повинні набувати значень NULL, які в базі даних позначають відсутність даних;

• user_password – пароль користувача, значення якого як і поле userName не повинно набувати значення NULL;

• user_type – поле для типу користувача, його значення повинні набувати лише ‘doctor’, ‘patient’ та ‘administration’, що позначають лікаря, пацієнта та адміністратора відповідно;

• registration_date – поле з типом даних Date, для позначення дати реєстрації.

2. Таблиця doctors для персональної інформації, яка стосується лікарів:

• doctor_id – унікальне поле для ідентифіккації лікаря, яке задається автоматично системою і є первинним ключем;

• user_id – поле для ідентифікації користувача, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці users(пов’язує дві таблиці(users та doctors));

• first_name – поле для реального імені лікаря, значення не повинні набувати значення NULL;

• last_name – поле для прізвища лікаря, аналогічно first_name значення NULL не допустиме;

• specialty – поле для збереження про спеціальність лікаря, обов’язково повинно бути заповнене, тобто не набувати значень NULL;

• contact_number – поле для контактного номеру телефону лікаря;

• hire_date – дата прийому на роботу лікаря, значення повинні відповідати типу Date для позначення дати;

• salary – поле для заробітньої плати, не є обов’язковим для заповнення під час реєтрації, значення можуть бути як цілочисельними так і дійсними;

• department_id – поле для ідентифікації відділення, до якого відноситься лікар, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці department (відділення лікарні) (пов’язує дві таблиці(department та doctors));

3. Таблиця patients для персональної інформації , яка стосується пацієнтів:

• patient_id – унікальне поле для ідентифіккації пацієнта, яке задається автоматично системою і є первинним ключем;

• user_id – поле для ідентифікації користувача, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці users(пов’язує дві таблиці(users та patients));

• first_name – поле для реального імені користувача з типом patient, значення не повинні набувати значення NULL;

• last_name – поле для прізвища пацієнта, аналогічно first_name значення NULL не допустиме;

• gender – поле для визначення статі пацієнта , містить лише два значення male(чоловік) та female(жінка);

• date_of_birth – поле для дати народження користувача, значення повинні відповідати типу Date для позначення дати;

• contact_number – поле для контактного номеру телефону пацієнта;

• email – поле для контактної електронної пошти;

• contact_address – поле для фактичної адреси проживання користувача з типом ‘patient’.

4. Таблиця administrations для персональної інформації , яка стосується адміністраторіа:

• admin_id – унікальне поле для ідентифіккації адміністратора, яке задається автоматично системою і є первинним ключем;

• user_id – поле для ідентифікації користувача, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці users(пов’язує дві таблиці(users та administrations));

• first_name – поле для реального імені адміністратора, значення не повинні набувати значення NULL;

• last_name – поле для прізвища адміністратора, аналогічно first_name значення NULL не допустиме;

• contact_number – поле для контактного номеру телефону адміністратора;

• email – поле для контактної електронної пошти;

• hire_date – дата прийому на роботу адміністратора, значення повинні відповідати типу Date для позначення дати;

• salary – поле для заробітньої плати, не є обов’язковим для заповнення під час реєтрації, значення можуть бути як цілочисельними так і дійсними;

5. Таблиця departments для даних про відділення у лікарні:

• department_id – унікальне поле для ідентифіккації відділення, яке задається автоматично системою і є первинним ключем;

• department_name – поле для назви відділення, не повинне набувати значення NULL, яке позначає відсутність даних;

• head_doctor_id – поле для ідентифікації лікаря, який є головний у відділенні, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці doctors(пов’язує дві таблиці(doctors та adepartments));

6. Таблиця appointment_schedule для збереження розкладу прийому пацієнтів:

• appointment_id – унікальне поле для ідентифіккації запису на прийом, яке задається автоматично системою і є первинним ключем;

• doctor_id – поле для ідентифікації лікаря, який проводитиме прийом, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці doctors (пов’язує дві таблиці (doctors та appointment_schedule));

• patient_id – поле для ідентифікації пацієнта, який записався на прийом, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці patients (пов’язує дві таблиці (patients та appointment_schedule));

• appointment_date – дата проведення прийому, значення повинні відповідати типу Date для позначення дати;

• appointment_time – час проведення прийому, значення повинні відповідати типу Time для позначення часу;

• status_schedule – поле для позначення статусу прийому, яке може набивати лише значень ‘scheduled’ (заплановано), ‘completed’ (завершено), ‘canceled’(скасовано).

7. Таблиця doctor_schedule для збереження робочого графіку лікаря:

• schedule_id – унікальне поле для ідентифіккації запису про графік роботи лікаря , яке задається автоматично системою і є первинним ключем;

• doctor_id – поле для ідентифікації лікаря, який працює за цим робочим графіком, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці doctors (пов’язує дві таблиці (doctors та doctors_schedule));

• date_of_week – поля для позначення дня тижня, значення повинні набувати значень ‘Monday’ (Понеділок), ‘Tuesday’ (Вівторок). ‘Wednesday’ (Середа), ‘Thursday’ (Четвер), ‘Friday’ (П’ятниця)

• start_time – час початку роботи лікаря, значення повинні відповідати типу Time для позначення часу;

• end_time – час закінчення роботи лікаря, значення повинні відповідати типу Time для позначення часу;

8. Таблиця prescription для рецепту лікування написаного лікарем для пацієнта:

• prescription_id – унікальне поле для ідентифікації рецепту, яке задається автоматично системою і є первинним ключем;

• doctor_id – поле для ідентифікації лікаря, який виписав рецепт, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці doctors (пов’язує дві таблиці (doctors та prescription));

• patient_id – поле для ідентифікації пацієнта, для якого було виписано рецепт, це зовнішній ключ, який містить посилання на поле первинного ключа в таблиці patients (пов’язує дві таблиці (patients та prescription));

• prescription_date – дата створення рецепту, значення повинні відповідати типу Date для позначення дати;

• medication – поле для назви призначеного лікарем препарату;

• dosage – поле для визначення дозування лікарського препарату;

• instructions – поле для детальних інструкцій прийому препарату.

На основі структури окремо опишемо взаємозв’язки:

• таблиця users пов’язана з таблицями patients, doctors, administrations за допогою свого первинного ключа та зовнішніх ключів таблиць;

• таблиця appointment_schedule пов’язана з patients та doctors через зовнішні ключі patient_id та doctor_id;

- таблиця department використовує head_doctor_id як зовнішній ключ для визначення лікаря, який керує відділенням;
- таблиця doctor_schedule пов’язана з doctors через зовнішній ключ doctor_id;
- таблиця prescription пов’язана з doctors та patients через зовнішні ключі doctor_id та patient_id.

Додаток Б . Політика інформаційної безпеки

Загальні положення

Визначення термінів

Інформаційна система (ІС) – сукупність програмного та апаратного забезпечення яке використовується для обробки даних.

Політика безпеки – правила та процедури що використовуються для захисту ІС від несанкціонованого доступу, використання, розкриття, порушень конфіденційності, цілісності та доступності.

Користувач – особа, яка має доступ до ІС (*наприклад,* для системи лікарні це пацієнти, лікарі та адміністративний персонал).

Персональні дані – відомості чи сукупність відомостей про фізичну особу, яка ідентифікована або може бути конкретно ідентифікована.

Особиста інформація – персональні дані та не персоніфікована інформація які вноситься в ІС клієнтами та можуть ними редагуватись.

Конфіденційність – властивість інформації, яка полягає в тому, що інформація не може бути отримана неавторизованим користувачем та/або процесом.

Цілісність – властивість інформації, яка полягає в тому, що інформація не може бути модифікована неавторизованим користувачем та/або процесом.

Доступність – властивість досяжності й можливості використання інформації на вимогу авторизованого об’єкта.

Адміністратор інформаційної системи – особа, яка у відповідності до своїх посадових обов’язків організовує та підтримує в належному стані ІС як цілісний апаратно-програмний комплекс здійснюючи її налаштування, обслуговування та захист.

Адміністратор персональних даних – особа, на яку покладено відповідальність за організацію належного контролю за дотриманням національного законодавства та міжнародних стандартів з обробки персональних даних в процесі обробки та зберігання інформації в ІС.

Співробітники – особи, які отримують обмежений доступ до інформації що зберігається в ІС, в обсязі необхідному для виконання свої безпосередніх завдань.

Клієнти – особи які можуть здійснювати доступ до інформаційної системи віддалено, виключно в межах попередньо визначних функціональних можливостей і лише в частині особистої інформації.

Цілі документа

Ціллю даної політики є впровадження та ефективне функціонування системи управління інформаційною безпекою організації, яка полягає в:

1. Захисті конфіденційності даних, що включає в себе забезпечення захисту персональних даних користувачів від несанкціонованого доступу.

2. Забезпеченні цілісності даних, що включає в себе забезпечення їх точності, надійності та повноти.

3. Забезпеченні доступності даних, що включає в себе безперебійну роботу системи та доступ авторизованих користувачів до даних без стороннього втручання і затримок.

4. Дотриманні нормативно-правових вимог про захист персональних даних та забезпечення безпеки ІС, що включає в себе дотримання місцевих та міжнародних стандартів, які стосуються персональних даних.

5. Управлінні ризиками, а саме: в регулярній оцінці ризиків та постійному моніторингу ІС для завчасного виявлення вразливостей та загроз, їх усунення або ж мінімізацію ризиків їх реалізації.

6. Освіті та навчанні персоналу задля забезпечення захищеності ІС та підвищення обізнаності користувачів про важливість дотримання процедур політики безпеки.

7. Відповіді на інциденти, що включає в себе процедури реагування (оцінку, розслідування та усунення інцидентів) та план відновлення у разі повного виведення системи з робочого стану.

Сфера застосування

Розроблена політика безпеки застосовується до всіх аспектів інформаційних систем що здійснюють обробку персональних даних і охоплює апаратне та програмне забезпечення, бази даних, мережі та персонал. Для забезпечення всебічного захисту всі компоненти ІС повинні відповідати визначеним стандартам безпеки та принципам інформаційної безпеки.

Організаційна структура інформаційної безпеки

В організаціях, що та використовують систему з публічним доступом та здійснюють обробку персональних даних організаційна структура інформаційної безпеки передбачає 4 типи користувачів на різних рівнях: клієнти, співробітники, адміністратори персональних даних та адміністратори ІС.

Клієнти є основним джерелом персональних даних для ІС. Вони мають мінімальні права доступу та можуть запитувати в ІС лише особисту інформацію. До повноважень і обов’язків клієнтів входять наступні аспекти:

– забезпечення цілісності даних: клієнти мають право переглядати та редагувати особисті персональні дані, які зберігаються в ІС .

– конфіденційність даних: клієнти несуть персональну відповідальність за зберігання параметрів авторизації в ІС вони повинні зберігати ці дані в таємниці та не розкривати їх третім особам;

– доступність даних: клієнти, у разі необхідності, можуть звертатись з клопотанням до адміністратора персональних даних з запитом на видалення частини особистих персональних даних з ІС;

– повідомлення про інциденти: клієнти повинні повідомити адміністратора ІС про підозрілі дії або інциденти, які були помічені під час використання ІС.

Співробітники використовують ІС для виконання своїх посадових обов’язків та можуть в їх межах отримувати доступ до персональних даних які зберігаються в ІС. Для забезпечення дотримання інформаційної безпеки на співробітників покладено ряд повноважень та обов’язків. А саме:

– забезпечення цілісності даних: в ІС співробітники можуть переглядати, редагувати та створювати пов’язані записи клієнтів в частині що не містить персональних даних;

– конфіденційність даних: співробітники мають усвідомлювати та забезпечувати конфіденційність персональної інформації клієнтів і не розкривати її без відповідного дозволу;

– доступність даних: у разі необхідності отримання розширеного переліку персональних даних чи їх редагування, співробітники мають звертатись з відповідним клопотанням до адміністратора персональних даних;

– звітування про інциденти: співробітники повинні негайно повідомити адміністратора ІС, якщо під час роботи системи відбулися підозрілі дії або інциденти.

Адміністратори персональних даних здійснюють загальний нагляд за дотриманням національного законодавства та міжнародних стандартів з обробки персональних даних в ІС та мають найширші права доступу до персональних даних які обробляються в ІС, проте не мають прав редагування структури даних. До повноважень і обов’язків адміністраторів персональних даних відносять:

– забезпечення цілісності та доступності даних: додавання, редагування чи видалення персональних даних має здійснюватись виключно як відповідь на відповідне клопотанні і у відповідності до логіки функціонування ІС;

– конфіденційність даних: адміністратори персональних даних мають забезпечувати конфіденційність персональних даних що надаються за запитом співробітників і не розкривати її без відповідного дозволу (або за наявності прямої заборони). За можливості інформація має бути де персоніфікована;

Адміністратори ІС відіграють ключову роль у координації та управлінні діяльністю організації, забезпечуючи ефективність та безпеку інформаційної системи. Водночас, адміністратори ІС мають бути обмежені правах доступу та обробки персональних даних, здійснюючи лише визначення структури даних та їх взаємозв’язків. Їхні повноваження та обов’язки включають:

– моніторинг системи: здійснення постійного контролю ІС (виявлення потенційних загроз та ненормальної поведінки системи);

– забезпечення безпеки та конфіденційності даних: підтримка заходів захисту даних від несанкціонованого доступу сторонніми особами, їх викрадення, втрату чи зловмисного використання.

– забезпечення цілісності даних: у разі якщо виявлено порушення цілісності бази даних ІС в частині що містить персональні дані, виправлення помилок та відновлення даних має відбуватись в узгодженні з адміністратором персональних даних;

– контроль доступу: адміністратори мають право додавати до системи нових користувачів, надавати чи обмежувати їх права, а також видаляти вже існуючих користувачів за запитом.

Технічні заходи інформаційної безпеки

Засоби авторизації та автентифікації користувачів:

– необхідно здійснювати перевірку згенерованого (запропонованого користувачем) пароля на надійність як при створенні нового користувача, так і при заміні пароля в майбутньому. Система повинна мати механізми перевірки складності пароля, при чому користувачеві повинна надаватися інформація щодо описаних в системі вимог для створення надійного пароля. Система повинна в реальному часі надавати відповідні рекомендації під час введення пароля на сторінці реєстрації методом повідомлення про ступінь надійності веденого пароля.

– задля підвищення рівня конфіденційності даних користувачів ІС повинна мати функцію автоматичної періодичної зміни пароля, яка повинна бути реалізована за допомогою генератора складних паролів , який з заданою періодичністю створює для користувача новий пароль та відправляє його через один із можливих каналів зв’язку(електронна пошта, SMS і т.п.) або ж функцію примусової зміни пароля користувачем, яка буде вимагати від користувачів змінювати паролі через визначений період часу надсилаючи нагадування на один з можливих каналів зв’язку.

– задля підтвердження особи користувача система повинна підтримувати кілька методів двохфакторної авторизації , наприклад, після введення імені користувача та пароля, користувач повинен ввести одноразовий код з SMS/електронної пошти .

Шифрування даних є надійним способом забезпечити як конфіденційність так і цілісність даних:

– система повинна використовувати наскрізне шифрування даних БД. Задля забезпечення шифрування чутливих даних система повинна застосовувати алгоритми шифрування при їх зберіганні, при чому повинні використовуватись надійні методи управління ключами шифрування.

– задля безпечного обміну даних між клієнтом та сервером повинно використовуватись шифрування даних, які передаються по каналам зв’язку, наприклад використання HTTPS та використання TLS, при чому під час роботи системи повинно відбуватись регулярне оновлення та підтримка усіх протоколів безпечного з’єднання, які використанні в системі.

Захист від вразливостей:

– задля мінімізації ризиків або/і запобігання виникнення вразливостей система повинна регулярно оновлювати усі компоненти програмного забезпечення.

Політика звітності та ведення журналу

Реєстрація загроз та порушень:

Кожен інцидент безпеки, незалежно від його типу та ступеня тяжкості реєструється у відповідному журналі подій безпеки. Цей журнал містить детальну інформацію про дату та час події, її функції (такі як спроби несанкціонованого доступу, зміни конфігурації системи та виявлення вразливостей), а також усі дії, вжиті для вирішення ситуації.

З метою виявлення причин виникнення інцидентів безпеки, виявлення недоліків безпеки і розробки заходів щодо їх усунення, кожен інцидент повинен бути обов’язково проаналізований. Результати аналізу повинні бути задокументовані для подальшого використання.

Журнал інцидентів безпеки повинен зберігатися протягом періоду визначеного внутрішніми правилами та нормативними вимогами організації що використовує інформаційну систему.

Системний адміністратор повинен регулярно повідомляти про стан безпеки інформаційної системи керівнику організації, яка її використовує. Звіти мають складатися у відповідності з фіксованим графіком або вразі виникнення серйозного інциденту. Плановий звіт щодо виконання політики безпеки ІС має включати в себе аналіз історії інцидентів безпеки. Зокрема в ньому слід підкреслити виявлені повторювані атаки та інциденти, а також відмітити тенденції розвитку інформаційних загроз. Звіт повинен містити інформацію про заходи щодо усунення виявлених загроз і порушень, поточний стан захисту системи та рекомендації щодо подальших дій задля підвищення рівня інформаційної безпеки.

У разі серйозного інциденту з безпекою адміністратор повинен негайно повідомити про це керівнику організації та вжити термінових заходів, що дозволить швидко реагувати на можливі загрози та мінімізувати шкоду системі та даним.