ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут Інформаційних технологій
Кафедра комп'ютерної інженерії

Пояснювальна записка
до бакалаврської роботи
на ступінь вищої освіти бакалавр
на тему: «**БЕЗПЕЧНА АРХІТЕКТУРА ЦИФРОВОЇ ІДЕНТИФІКАЦІЇ ОСОБИ НА ОСНОВІ БЛОКЧЕЙН»**

Виконав: студент 6 курсу, групи КСДМ–61
спеціальності
123 Комп'ютерна інженерія
(шифр і назва спеціальності)
Чупахін М.С.
(прізвище та ініціали)
Керівник Антоненко А. В.
(прізвище та ініціали)
Рецензент _____
(прізвище та ініціали)

Київ – 2024

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Факультет Інформаційних технологій
Кафедра Комп'ютерної інженерії
Ступінь вищої освіти - «Магістр»
Спеціальність підготовки – 123 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ
Завідувач кафедри
Комп'ютерної інженерії

"____" _____2024 року

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Чупахіну Микиті Сергійовичу
(прізвище, ім'я, по батькові)

1.Тема роботи: «Безпечна архітектура цифрової ідентифікації особи на основі блокчейн»

Керівник роботи Антоненко Артем Васильович

кандидат технічних наук, доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу року №
2. Строк подання студентом роботи
3. Вхідні дані до роботи: основні методи реєстрації сигналів головного мозку, види периферійних пристроїв.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити).
5. Демонстраційні матеріали.
6. Дата видачі завдання

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Збір даних	14.08-24.09.2024	Виконано
2	Дослідження блокчейну, технологій захисту інформації в блокчейн системах	24.09-13.10.2024	Виконано
3	Дослідження хмарних систем та архітектур для діджиталізації особистості	13.10-19.10.2024	Виконано
4	Проектування системи та архітектури візової системи на базі блокчейн технологій	19.10-10.11.2024	Виконано
5	Обробка матеріалу	10.11-14.11.2024	Виконано
6	Висновки за результатами аналізу	14.11-17.11.2024	Виконано
7	Розробка теоретичної частини	17.11-18.11.2024	Виконано
8	Розробка презентації	18.11-19.11.2024	Виконано
9	Передзахист	9.12-11.12.2024	Виконано
10	Здача в деканат	20.12-29.12.2024	Виконано

Студент _____ Чупахін М.С.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Антоненко А. В.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина магістерської роботи с.66, Рис.1, Джерела 34.

БЛОКЧЕЙН, ДІДЖИТАЛІЗАЦІЯ ОСОБИСТОСТІ, ЦИФРОВА ІДЕНТИФІКАЦІЯ, БЕЗПЕКА ПЕРСОНАЛЬНИХ ДАНИХ, ДЕЦЕНТРАЛІЗАЦІЯ ТЕХНОЛОГІЇ.

Об'єкт дослідження. Використання технологій блокчейну для забезпечення безпеки та конфіденційності у процесі диджиталізації особистості.

Предмет дослідження. Методи інтеграції блокчейну в процеси створення цифрових ідентичностей, забезпечення їхньої безпеки, а також ефективне використання у сфері верифікації особи.

Мета роботи. Аналіз потенціалу використання блокчейн-технологій для створення децентралізованих і безпечних систем цифрової ідентифікації. Розробка концептуальної моделі використання блокчейну у процесі диджиталізації особистості, визначення основних викликів та переваг такого підходу.

Методи дослідження. Аналіз наукових праць і технологічної документації. Порівняльний аналіз існуючих систем цифрової ідентифікації. Узагальнення передових практик у галузі безпеки персональних даних.

Завдання дослідження. Вивчити сучасний стан диджиталізації особистості та проблеми, пов’язані з безпекою персональних даних. Проаналізувати переваги та недоліки використання блокчейну для цифрової ідентифікації. Розробити концептуальну модель інтеграції блокчейну у системи диджиталізації особистості. Визначити основні технічні та економічні виклики впровадження блокчейн-рішень.

Наукова новизна. Дослідження пропонує новий формат візової системи на основі блокчейн технологій, що забезпечує прозорість, високу безпеку та захист від підробки.

Практичне значення. Результати дослідження можуть бути використані для: розробки сучасних систем цифрової ідентифікації, що базуються на блокчейні; вдосконалення бізнес-процесів у сферах банківської справи, медицини, електронного урядування, де використовується цифрова ідентифікація; подальших досліджень у галузі кібербезпеки та децентралізованих технологій.

ЗМІСТ

4. ПРОЕКТУВАННЯ МІЖНАРОДНОЇ БЛОКЧЕЙН-ВІЗОВОЇ СИСТЕМИ

4.1. Технічна архітектура: хмарні сервіси та розподілене зберігання даних
4.2. Алгоритм роботи системи: створення та оновлення цифрових ідентифікаторів
4.3. Моделювання процесу взаємодії між ключовими органами (уряди, консульства, користувачі)
4.4. Технічна реалізація: приклади коду, інтеграція API та перевірка ідентифікації
4.5. Інтеграція з глобальними платформами: приклади взаємодії з ID2020, Hyperledger

5. ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ ВПРОВАДЖЕННЯ

5.1. Ресурси для розробки та впровадження системи
5.2. Моделювання навантаження та аналіз продуктивності
5.3. Вартість інтеграції у міжнародну систему
5.4. Порівняння економічної ефективності блокчейн-рішень з традиційними підходами
5.5. Оцінка ефективності з погляду безпеки та прозорості

ВИСНОВКИ
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

ВСТУП

Актуальність теми. У сучасному цифровому суспільстві диджиталізація особистості стає важливим напрямом розвитку інформаційних технологій. Зростання кількості онлайн-сервісів, впровадження електронного урядування та міжнародна взаємодія вимагають нових підходів до ідентифікації та захисту персональних даних. Традиційні методи централізованого зберігання інформації не забезпечують достатньої безпеки та часто піддаються кіберзагрозам. Технологія блокчейну, що забезпечує децентралізоване зберігання даних і прозорість процесів, пропонує перспективні рішення для створення безпечних і гнучких систем цифрової ідентифікації.

Ступінь опрацьованості проблеми. Тема диджиталізації особистості активно досліджується у наукових і практичних колах. Значна кількість праць присвячена питанням створення цифрових ідентичностей, розробці систем кібербезпеки та використанню блокчейну у фінансовій і державній сферах. Однак недостатньо уваги приділено питанням інтеграції блокчейн-технологій у процеси диджиталізації особистості, зокрема щодо їхнього застосування у міжнародних системах ідентифікації та мінімізації ризиків витоків даних.

Цілі та завдання роботи. Метою дослідження є розробка концептуальної моделі інтеграції блокчейн-технологій у процеси диджиталізації особистості для забезпечення високого рівня безпеки та прозорості управління персональними даними. Для досягнення мети необхідно виконати такі завдання: проаналізувати сучасні підходи до диджиталізації особистості, дослідити можливості блокчейну у цій сфері, розробити архітектурну модель децентралізованої системи, оцінити її ефективність і можливі економічні наслідки впровадження.

Об'єкт дослідження. Процеси диджиталізації особистості у контексті використання технологій блокчейну для створення децентралізованих систем управління персональними даними.

Предмет дослідження. Архітектурні й технічні аспекти створення децентралізованої системи диджиталізації особистості на основі блокчейну, включаючи протоколи безпеки, алгоритми консенсусу та методи управління доступом до персональних даних.

Методологія дослідження. Для досягнення поставлених завдань використовувалися методи аналізу наукових праць, моделювання архітектури блокчейн-систем, порівняння існуючих підходів до диджиталізації, а також експериментальні методи оцінки безпеки та ефективності запропонованого рішення.

Наукова новизна. Наукова новизна роботи полягає у розробці концептуальної моделі децентралізованої системи диджиталізації особистості на основі блокчейну. Запропонований підхід забезпечує прозорість, безпеку й анонімність у процесах управління персональними даними, а також враховує можливість глобальної інтеграції та використання у міжнародних платформах.

1. ОСНОВИ ДІДЖИТАЛІЗАЦІЇ ОСОБИСТОСТІ ТА БЛОКЧЕЙНУ

1.1. Сучасні підходи до диджиталізації особистих даних

У сучасному світі диджиталізація особистих даних є невід'ємною частиною розвитку цифрових технологій. Це дозволяє оптимізувати процеси в багатьох сферах: електронному урядуванні, банківській справі, охороні здоров'я, освіті та інших. Розглянемо основні підходи до диджиталізації особистих даних, які існують на сьогоднішній день.

1.1.1. Централізовані системи управління даними

Централізовані системи є традиційним підходом до зберігання і управління особистими даними. У таких системах всі дані зберігаються на одному або кількох серверах, які контролюються центральним органом (державним чи приватним).

Переваги:

- Висока швидкість обробки даних у межах однієї організації.
- Простота інтеграції з внутрішніми системами.

Недоліки:

- Високий ризик компрометації через одну точку уразливості.
- Обмежена прозорість для користувачів.
- Складність інтеграції з іншими системами, особливо на міжнародному рівні.

1.1.2. Децентралізовані системи управління даними

Цей підхід базується на розподілі даних між кількома вузлами мережі, які працюють незалежно. Децентралізація дозволяє зменшити залежність від одного адміністратора та підвищує стійкість до атак.

Переваги:

- Висока стійкість до збоїв і атак.
- Більша прозорість та контроль з боку користувачів.

Недоліки:

- Складність впровадження.
- Вища вартість підтримки у порівнянні з централізованими системами.

1.1.3. Хмарні технології для диджиталізації даних

Хмарні платформи дозволяють зберігати дані у централізованих або децентралізованих інфраструктурах із доступом через Інтернет. Вони забезпечують масштабованість та гнучкість для користувачів.

Переваги:

- Глобальний доступ до даних у реальному часі.
- Можливість інтеграції з багатьма сервісами.

Недоліки:

- Залежність від провайдерів хмарних послуг.
- Потенційні ризики втрати конфіденційності даних.

1.1.4. Технології блокчейну у диджиталізації даних

Блокчейн є інноваційним підходом до зберігання даних, який ґрунтується на розподілених реєстрах і криптографічному захисті. Ця технологія

дозволяє створювати децентралізовані системи управління даними, що забезпечують високий рівень безпеки, прозорості та надійності.

Особливості:

- Дані зберігаються у вигляді блоків, зв’язаних між собою криптографічно.
- Жодна сторона не має абсолютного контролю над даними.

Переваги:

- Підвищений рівень безпеки.
- Незмінність записів.
- Можливість використання смарт-контрактів для автоматизації.

Недоліки:

- Обмежена масштабованість у порівнянні з традиційними базами даних.
- Висока енерговитратність деяких протоколів.

1.2. Принципи роботи блокчейн-технології

Блокчейн є технологією розподіленого реєстру, що забезпечує зберігання даних у вигляді послідовності блоків, з’єднаних між собою криптографічними методами. Ця технологія гарантує незмінність, прозорість і безпеку даних, що робить її ідеальною для застосування у сферах, де потрібен високий рівень довіри між учасниками.

1.2.1. Структура блокчейну

Основними компонентами блокчейн-системи є:

1. Блоки:

Кожен блок містить три ключові елементи:

- Дані: інформація, яка зберігається в блоці (транзакції, записи тощо).
- Хеш блоку: унікальний криптографічний ідентифікатор блоку.
- Хеш попереднього блоку: зв’язок із попереднім блоком для формування ланцюга.

2. Ланцюг блоків:

Блоки з'єднані між собою за допомогою хешів, що забезпечує незмінність даних. Якщо хоча б один блок буде змінений, це порушить цілісність усього ланцюга.

3. Розподілений реєстр:

Копія блокчейну зберігається у кожного учасника мережі (вузла), що забезпечує децентралізований доступ до даних.

1.2.2. Алгоритми консенсусу

Щоб забезпечити погодження між вузлами щодо стану реєстру, використовуються алгоритми консенсусу:

1. Proof of Work (PoW):
    - Учасники (майнери) виконують обчислювальні завдання для додавання нового блоку.
    - Переваги: висока безпека.
    - Недоліки: велика енерговитратність.
2. Proof of Stake (PoS):
    - Вузли створюють блоки на основі кількості їхніх токенів (ставок).
    - Переваги: низьке енергоспоживання.

- Недоліки: можливість концентрації влади у власників великих ставок.

3. Delegated Proof of Stake (DPoS):
    - Учасники делегують право валідації блоків обраним представникам.
    - Переваги: швидкість обробки транзакцій.
    - Недоліки: залежність від представників.

1.2.3. Криптографія у блокчейні

1. Хешування:

Використовуються алгоритми (SHA-256, SHA-3) для створення унікального ідентифікатора даних.

2. Цифрові підписи:

Методи асиметричної криптографії (ECDSA, RSA) гарантують автентичність та захист транзакцій.

1.2.4. Смарт-контракти

Смарт-контракти — це програми, що автоматично виконують умови угод, записані у блокчейні. Вони забезпечують:

- Автоматизацію процесів.
- Незалежність від людського втручання.
- Прозорість виконання.

1.2.5. Мережеві рівні блокчейну

1. Інфраструктурний рівень: протоколи мережі та алгоритми консенсусу.
2. Дані: зберігання блоків і транзакцій.
3. Програмний рівень: реалізація смарт-контрактів і додатків.

1.3. Типи блокчейн-систем

Блокчейн-системи поділяються на три основні типи залежно від доступу до мережі, рівня децентралізації та моделі управління. Кожен із типів має свої особливості, переваги та недоліки, що визначають сферу їх застосування.

1.3.1. Публічні блокчейни

Це повністю децентралізовані системи, доступ до яких має кожен, хто підключається до мережі. Всі транзакції у публічному блокчейні є прозорими та доступними для перегляду.

Характеристики:

- Доступ: будь-яка людина може приєднатися до мережі як вузол, виконувати транзакції та брати участь у консенсусі.
- Прозорість: дані про всі транзакції зберігаються у відкритому доступі.
- Алгоритми консенсусу: зазвичай використовують Proof of Work (PoW) або Proof of Stake (PoS).

Переваги:

- Максимальна децентралізація.
- Висока довіра через прозорість.
- Стійкість до цензури.

Недоліки:

- Низька продуктивність у великих мережах (повільна обробка транзакцій).
- Висока енерговитратність у разі використання PoW.

Приклади:

- Bitcoin.
- Ethereum.

1.3.2. Приватні блокчейни

Приватні блокчейни є централізованими системами, де доступ до мережі контролюється певною організацією. Лише авторизовані користувачі можуть переглядати транзакції та брати участь у консенсусі.

Характеристики:

- Доступ: обмежений, дозволяється лише запрошеним вузлам.
- Управління: централізоване, контроль здійснюється одним органом.
- Консенсус: використовуються менш енерговитратні алгоритми (наприклад, Practical Byzantine Fault Tolerance, Proof of Authority).

Переваги:

- Висока продуктивність.
- Захищеність конфіденційної інформації.
- Простота управління.

Недоліки:

- Залежність від центрального органу управління.
- Обмежена прозорість для сторонніх користувачів.

Приклади:

- Hyperledger Fabric.
- Corda.

1.3.3. Консорціумні блокчейни

Це гібридна модель, яка поєднує в собі характеристики публічних та приватних блокчейнів. У консорціумних системах управління здійснюється групою організацій, які мають рівні права доступу.

Характеристики:

- Доступ: частково обмежений, учасниками мережі є обрані організації.
- Управління: децентралізоване серед членів консорціуму.
- Консенсус: забезпечується узгодженням між визначеним колом учасників.

Переваги:

- Баланс між децентралізацією та продуктивністю.
- Прозорість у межах консорціуму.
- Надійність і масштабованість.

Недоліки:

- Складність в управлінні у великих консорціумах.
- Вимоги до високого рівня координації між учасниками.

Приклади:

- Quorum.
- Ripple.

Таблиця 1.1. Порівняння типів блокчейн-систем.

Тип	Доступ	Управління	Прозорість	Застосування

Публічний	Відкритий для всіх	Децентралізоване	Висока	Криптовалюти, NFT
Приватний	Лише для запрошених	Централізоване	Низька	Корпоративні рішення
Консорціумний	Обмежений	Частково децентралізоване	Середня	Банківські системи, логістика

1.4. Технічні аспекти інтеграції блокчейну у процес диджиталізації

Інтеграція блокчейну у процеси диджиталізації особистих даних є складним завданням, яке вимагає врахування різноманітних технічних аспектів. Впровадження цієї технології забезпечує прозорість, безпеку та надійність управління даними, але потребує чітко спланованої архітектури, відповідних протоколів і інфраструктури.

1.4.1. Архітектура блокчейн-системи для диджиталізації

Архітектура блокчейн-системи для роботи з особистими даними включає такі основні компоненти:

1. Розподілений реєстр:
    - Місце зберігання даних у вигляді блоків.
    - Дані записуються у вигляді транзакцій із посиланням на попередній блок.
2. Ноди (вузли):
    - Учасники мережі, які зберігають копії реєстру.

- Ноди можуть бути повними (зберігають увесь реєстр) або легкими (працюють з частковими даними).

3. Смарт-контракти:
    - Автоматизовані програми, які виконують наперед визначені умови.
    - Використовуються для перевірки прав доступу, оновлення даних тощо.
4. Інтеграційний API:
    - Інтерфейси для підключення зовнішніх систем (державних реєстрів, хмарних сервісів) до блокчейну.

1.4.2. Обробка та зберігання даних

1. Шифрування даних:
    - Особисті дані зберігаються у зашифрованому вигляді для забезпечення конфіденційності.
    - Використовуються симетричні та асиметричні алгоритми шифрування (AES, RSA, ECDSA).
2. Хешування даних:
    - Для верифікації автентичності даних застосовується хешування (SHA-256, SHA-3).
    - Хеші дозволяють уникнути зберігання великих обсягів інформації без втрати її цілісності.
3. Ончейн та офчейн рішення:
    - Ончейн: дані зберігаються безпосередньо у блокчейні, що гарантує їхню незмінність.
    - Офчейн: великі обсяги інформації зберігаються поза блокчейном (наприклад, у хмарних сховищах), а у блокчейні зберігаються лише посилання та хеші.

1.4.3. Мережеві протоколи та консенсус

1. Протоколи передачі даних:

Використовуються стандарти, такі як HTTPS, MQTT, WebSocket, для забезпечення захищеного обміну даними між вузлами.

2. Алгоритми консенсусу:

Вибір алгоритму залежить від типу блокчейну:

- PoW (Proof of Work): забезпечує захист від атак, але має високу енерговитратність.
- PoS (Proof of Stake): дозволяє знизити витрати на енергію.
- PBFT (Practical Byzantine Fault Tolerance): підходить для приватних і консорціумних блокчейнів завдяки високій швидкості.

1.4.4. Інтеграція з існуючими системами

1. Взаємодія з державними реєстрами:
    - Для автоматизації обміну даними з державними органами розробляються спеціальні шлюзи (gateway).
    - Блокчейн може використовуватися як інструмент для верифікації даних.
2. Хмарна інфраструктура:
    - Хмари використовуються для зберігання великих обсягів даних (офчейн) та обчислень.
    - Популярні рішення: AWS, Azure, Google Cloud.
3. Сумісність із мобільними та веб-додатками:

API інтегрується з додатками для роботи користувачів (реєстрація, верифікація, доступ до даних).

1.4.5. Проблеми масштабованості та продуктивності

1. Шардінг:

Розділення блокчейну на сегменти (шарди), які обробляються паралельно.

2. Другі рівні блокчейну:

Використання технологій, як-от Lightning Network, для зменшення навантаження на основний ланцюг.

3. Оптимізація смарт-контрактів:

Розробка ефективного коду для зменшення обчислювальної складності.

1.4.6. Безпека системи

1. Захист від атак:
    - Захист від атаки 51%, DNS-атак, підробки даних.
    - Регулярні оновлення протоколів та аудит смарт-контрактів.
2. Мультифакторна автентифікація:

Використання кількох рівнів авторизації для доступу до блокчейн-даних.

3. Резервне копіювання:

Створення копій даних для швидкого відновлення після збоїв.

2. АНАЛІЗ ПРОБЛЕМ БЕЗПЕКИ У ДІДЖИТАЛІЗАЦІЇ ОСОБИСТОСТІ

2.1. Основні загрози безпеці особистих даних

З розвитком технологій диджиталізації та впровадженням персональних даних у цифрове середовище виникає значна кількість ризиків, пов'язаних із безпекою. Особисті дані, будучи цінним активом, часто стають об'єктом кібератак і зловживань. Основні загрози безпеці можна класифікувати за кількома категоріями.

2.1.1. Несанкціонований доступ до даних

1. Хакерські атаки:
    - Злом баз даних із метою отримання конфіденційної інформації.
    - Використання викрадених облікових записів для доступу до систем.
2. Фішинг:
    - Обман користувачів через підроблені веб-сайти, електронні листи чи повідомлення.
    - Збір облікових даних для подальшого доступу до особистої інформації.
3. Брутфорс-атаки:
    - Підбір паролів для входу до акаунтів.
    - Актуально для слабких або стандартних паролів.

2.1.2. Порушення конфіденційності

1. Злив даних:
    - Незаконний продаж або розповсюдження особистих даних.
    - Часто трапляється через внутрішніх користувачів (інсайдерів) або недосконаліть безпеки.

2. Стороннє відстеження:

Збір інформації про користувачів без їхньої згоди (наприклад, через трекери у веб-додатках).

3. Проблеми з конфіденційністю у хмарних сховищах:

Витік даних через ненадійні або неправильно налаштовані хмарні сервіси.

2.1.3. Модифікація або знищення даних

1. Атаки на цілісність даних:
    - Несанкціоноване змінення інформації, що може призвести до неправдивих записів.
    - Наприклад, зміна особистих даних у державному реєстрі.
2. Ransomware-атаки:

Шифрування даних із вимогою викупу за їх розшифрування.

3. Ненавмисна втрата даних:

Помилки користувачів або програмного забезпечення.

2.1.4. Атаки на ідентифікацію та автентифікацію

1. Викрадення ідентифікаційних даних:

Крадіжка ідентифікаційних номерів, таких як паспортні дані або унікальні ідентифікатори.

2. Фальсифікація автентифікації:
    - Використання підроблених документів або інформації для доступу до системи.
3. Атаки «людина посередині» (Man-in-the-Middle):

Перехоплення даних під час передачі між користувачем і сервером.

2.1.5. Соціальна інженерія

1. Шантаж або обман:

Вимагання інформації шляхом тиску на користувача.

2. Соціальні атаки:

Використання психологічних маніпуляцій для отримання доступу до конфіденційних даних.

2.1.6. Правові та етичні ризики

1. Невідповідність нормативним вимогам:

Недотримання міжнародних стандартів і законодавства у сфері захисту даних (наприклад, GDPR).

2. Проблеми прозорості:

Відсутність чіткої інформації про те, як використовуються та захищаються дані.

3. Неналежна згода:

Використання даних без чіткої згоди користувачів.

2.2. Уразливості блокчейн-мереж: атаки 51%, Sybil-атаки, DoS

Попри високий рівень безпеки блокчейн-технології, її архітектура не є повністю захищеною від кібератак. Деякі уразливості виникають через особливості роботи децентралізованих мереж та алгоритмів консенсусу. Нижче розглядаються три основні типи атак: 51%, Sybil-атаки та атаки типу DoS.

2.2.1. Атака 51%

Атака 51% виникає, коли зловмисник або група зловмисників отримує контроль над більш ніж 50% обчислювальної потужності (у разі Proof of Work) або частки стейків (у разі Proof of Stake) у блокчейн-мережі.

Механізм атаки:

1. Зловмисник створює власний альтернативний ланцюг блоків (форк).

2. Завдяки контрольованій більшості ресурсів він може перевизначити ланцюг блоків, змусивши мережу прийняти фальшиві транзакції.
3. Наслідки:
    - Подвійне витрачання (double-spending): транзакція фальсифікується таким чином, що гроші можна використати двічі.
    - Відкат транзакцій: скасування вже підтверджених транзакцій.
    - Порушення довіри до мережі: зниження надійності та вартості криптовалюти чи даних у блокчейні.

Методи захисту:

- Децентралізація: збільшення кількості нод у мережі.
- Рандомізація вибору вузлів: у випадку PoS використовуються алгоритми, які ускладнюють маніпуляцію.
- Захист менших мереж: впровадження чекпоінтів (контрольних точок), які фіксують ланцюг у певний момент часу.

2.2.2. Sybil-атака

Sybil-атака полягає у створенні великої кількості підроблених вузлів (ідентичностей) у мережі, щоб отримати контроль над процесами голосування або консенсусу.

Механізм атаки:

1. Зловмисник створює безліч фальшивих нод із різними ID.
2. Ці ноди можуть впливати на процес консенсусу, фальсифікувати результати голосування або блокувати певні транзакції.

Наслідки:

- Маніпуляція консенсусом.
- Забезпечення домінування зловмисника у децентралізованій мережі.

- Дестабілізація роботи мережі та її учасників.

Методи захисту:

- CAPTCHA: перевірка на створення реальних ідентичностей.
- Proof of Identity (PoI): використання унікальних перевірених ідентифікаторів для нод.
- Економічні бар'єри: механізми, які ускладнюють створення великої кількості ідентичностей (наприклад, необхідність внесення застави).

2.2.3. Атака типу DoS (Denial of Service)

DoS-атака спрямована на перевантаження вузлів або мережі загалом, що унеможливлює її нормальну роботу.

Механізм атаки:

1. Зловмисник відправляє велику кількість транзакцій, які блокують обробку легітимних запитів.
2. Уразливі вузли мережі стають недоступними, а загальна продуктивність падає.

Різновиди атак:

- Спам-атаки: заповнення блокчейн-мережі великою кількістю транзакцій із незначними даними.
- Атака на пули майнерів: перевантаження вузлів, які генерують блоки.

Наслідки:

- Затримки підтвердження транзакцій.
- Підвищення плати за транзакції через збільшення попиту на обробку.

- Потенційна втрата довіри користувачів до мережі.

Методи захисту:

- Обмеження швидкості транзакцій: встановлення лімітів на обробку запитів від одного вузла.
- Фільтрація спаму: відхилення транзакцій із підозрілими параметрами.
- Мережеві протоколи захисту: використання firewall і алгоритмів для розпізнавання аномалій.

2.3. Проблеми масштабованості та консенсусу в блокчейні

Розвиток блокчейн-технологій стикається з низкою технічних викликів, серед яких найбільш актуальними є проблеми масштабованості та ефективності консенсусу. Ці проблеми впливають на швидкість обробки транзакцій, стабільність роботи мережі та її здатність обслуговувати велику кількість користувачів.

2.3.1. Масштабованість блокчейну

Масштабованість — це здатність блокчейну підтримувати стабільну продуктивність із ростом кількості учасників і транзакцій.

Основні обмеження:

1. Пропускна здатність (TPS):
    - У традиційних блокчейнах, таких як Bitcoin або Ethereum, кількість транзакцій, які можуть оброблятися за секунду (TPS), обмежена (до 7 TPS для Bitcoin і ~30 TPS для Ethereum).
    - Це значно менше, ніж у централізованих системах, наприклад, платіжних мережах Visa (до 24 000 TPS).

2. Розмір блоку:

Невеликий розмір блоку (наприклад, 1 МБ у Bitcoin) обмежує кількість транзакцій, які можуть бути записані в один блок.

3. Час генерації блоку:

У більшості мереж він становить від 10 секунд до 10 хвилин, що спричиняє затримки в підтвердженні транзакцій.

4. Мережеве навантаження:

Усі вузли повинні зберігати повну копію блокчейну, що збільшує потребу у пам’яті та пропускній здатності мережі.

Рішення проблем масштабованості:

1. Segregated Witness (SegWit):
    - Винесення частини даних транзакцій за межі основного блоку.
    - Збільшення пропускної здатності без змінення розміру блоку.
2. Рішення другого рівня (Layer 2):
    - Lightning Network: обробка транзакцій поза основним блокчейном із подальшим записом їхніх результатів.
    - Sidechains: паралельні ланцюги, які працюють поряд із основною мережею.
3. Шардінг (Sharding):
    - Розподіл блокчейну на сегменти (шарди), які обробляються окремо.
    - Це знижує навантаження на окремі вузли.
4. Proof of Stake (PoS):

Використання менш енергозатратних алгоритмів консенсусу для підвищення ефективності.

2.3.2. Проблеми консенсусу

Алгоритми консенсусу є основою роботи блокчейн-мережі, забезпечуючи узгодженість транзакцій між вузлами. Проте кожен з них має свої недоліки.

Основні виклики:

1. Proof of Work (PoW):
    - Високі енергетичні витрати через необхідність виконання складних обчислень.
    - Централізація потужностей через майнінгові пули.
    - Низька швидкість підтвердження транзакцій.
2. Proof of Stake (PoS):
    - Ризик централізації через концентрацію великої кількості токенів у деяких учасників.
    - Проблеми із захистом від атаки «нічого на кону» (Nothing-at-Stake).
3. Delegated Proof of Stake (DPoS):

Вузьке коло делегатів, що може спричинити олігархічну централізацію.

4. Byzantine Fault Tolerance (BFT):

Високі вимоги до обчислювальних потужностей при збільшенні кількості вузлів.

5. Гібридні алгоритми (наприклад, PoW+PoS):
    - Складність реалізації та підтримки.
    - Поєднання недоліків базових алгоритмів.

Рішення проблем консенсусу:

1. Оптимізація алгоритмів консенсусу:

Розробка нових алгоритмів, таких як Proof of History (PoH) у Solana або Avalanche Consensus.

2. Розподіл повноважень:

Розподіл ролей у мережі між різними типами вузлів для підвищення ефективності.

3. Модифікація існуючих алгоритмів:

Покращення PoS через динамічні механізми вибору валідаторів.

2.4. Вимоги до безпеки цифрових ідентифікаторів у міжнародних системах

Цифрові ідентифікатори є ключовим елементом у процесі диджиталізації особистості, особливо в міжнародних системах, таких як візові або паспортні системи. Їх використання вимагає високого рівня безпеки для запобігання шахрайству, несанкціонованому доступу та маніпуляціям даними.

Основні вимоги до безпеки цифрових ідентифікаторів

1. Конфіденційність
    - Цифрові ідентифікатори повинні бути захищеними від несанкціонованого доступу.
    - Використання криптографічних методів шифрування даних (AES, RSA).
2. Ідентичність та автентифікація
    - Кожен ідентифікатор має бути унікальним та прив’язаним до конкретної особи.
    - Використання мультифакторної автентифікації (MFA), що включає паролі, біометричні дані та апаратні токени.
3. Цілісність даних
    - Дані цифрових ідентифікаторів не повинні змінюватися без відповідного дозволу.

- Впровадження механізмів перевірки хешів (SHA-256, SHA-3) для контролю змін у даних.

4. Доступність
    - Цифрові ідентифікатори повинні бути доступними для авторизованих користувачів у будь-який час.
    - Захист системи від DoS-атак через балансування навантаження та резервування ресурсів.
5. Анонімність та мінімізація даних
    - Використання підходу "знання нуля" (Zero-Knowledge Proofs) для автентифікації без розкриття повної інформації.
    - Зберігання тільки необхідних даних для ідентифікації особи.
6. Міжнародна сумісність
    - Система цифрових ідентифікаторів повинна відповідати міжнародним стандартам (ISO/IEC 24760, eIDAS).
    - Забезпечення сумісності з різними блокчейн-мережами та існуючими базами даних.

Особливі вимоги для міжнародних візових систем

1. Централізована та децентралізована структура
    - Поєднання централізованого управління базами даних для урядових органів і децентралізованої архітектури для зберігання цифрових ідентифікаторів.
    - Розподіл функцій між різними вузлами мережі для зниження ризиків централізації.
2. Захист від підробки
    - Використання цифрових підписів для верифікації автентичності ідентифікаторів.
    - Впровадження блокчейну для збереження записів про кожну зміну або запит до ідентифікатора.
3. Прозорість і контроль доступу

- Чіткий розподіл прав доступу до даних між державними органами, організаціями та користувачами.
- Використання механізмів керування доступом на основі ролей (RBAC).

4. Можливість відстеження та оновлення
    - Відстеження всіх дій із цифровими ідентифікаторами за допомогою смарт-контрактів.
    - Забезпечення можливості динамічного оновлення даних без порушення їхньої цілісності.

Технічні рішення для підвищення безпеки

1. Блокчейн-платформи
    - Використання публічних або гібридних блокчейнів для запису та перевірки цифрових ідентифікаторів.
    - Впровадження консенсусних механізмів (Proof of Stake, Delegated Proof of Stake) для зниження ризику атак.
2. Смарт-контракти
    - Автоматизація процесів верифікації та оновлення ідентифікаторів.
    - Контроль доступу до даних через інтегровані механізми перевірки прав.
3. Біометрична ідентифікація
    - Інтеграція біометричних даних (відбитки пальців, сканування обличчя, райдужки ока) для автентифікації користувачів.
    - Захист біометричних даних за допомогою гомоморфного шифрування.
4. Протоколи захисту
    - Використання протоколів SSL/TLS для захисту передачі даних.

- Впровадження сучасних методів шифрування, таких як Elliptic Curve Cryptography (ECC).

3. РОЗРОБКА ІННОВАЦІЙНИХ РІШЕНЬ НА ОСНОВІ БЛОКЧЕЙНУ

3.1. Архітектура унікальної системи цифрового ідентифікатора

Архітектура системи цифрового ідентифікатора на основі блокчейну повинна забезпечувати децентралізацію, безпеку, масштабованість і прозорість. Запропонована система орієнтована на використання в міжнародних візових системах і включає ключові компоненти, які взаємодіють через захищені протоколи.

Основні компоненти архітектури

1. Блокчейн-мережа
    - Тип мережі: глобальний приватно-публічний блокчейн (hybrid blockchain), що поєднує публічну доступність для перевірки і приватний доступ для управління даними.
    - Алгоритм консенсусу: Proof of Stake (PoS) або Delegated Proof of Stake (DPoS) для підвищення енергоефективності та масштабованості.
2. Децентралізовані ідентифікаційні вузли (DID Nodes)
    - Розташування: урядові установи, консульства, міжнародні організації.
    - Функції: обробка запитів на створення, оновлення та верифікацію ідентифікаторів.
3. Смарт-контракти
    - Призначення: автоматизація процесів реєстрації, перевірки та оновлення цифрових ідентифікаторів.
    - Приклади: смарт-контракти для створення нових ідентифікаторів, перевірки дійсності даних, управління ролями доступу.
4. Об'єднане сховище даних (Off-chain storage)

- Дані: конфіденційна інформація, наприклад, біометрія та скановані документи.
- Технологія: захищені хмарні сховища або децентралізовані системи, такі як IPFS (InterPlanetary File System).
- Посилання на дані зберігаються у блокчейні через хешування.

5. Біометрична ідентифікація
    - Використовувані методи: сканування обличчя, райдужки ока, відбитки пальців.
    - Верифікація: біометричні дані захищені через гомоморфне шифрування, а автентифікація здійснюється через смарт-контракти.
6. Протоколи передачі даних
    - Використання SSL/TLS для шифрування переданих даних.
    - Інтеграція Zero-Knowledge Proofs (ZKP) для перевірки без розкриття зайвої інформації.

Етапи взаємодії системи

1. Реєстрація цифрового ідентифікатора
    - Користувач подає заявку через інтегровану платформу.
    - Заявка передається до вузла (DID Node) для перевірки.
    - Смарт-контракт створює унікальний цифровий ідентифікатор та записує його в блокчейн.
2. Верифікація ідентифікатора
    - Уповноважені організації (консульства, урядові установи) запитують перевірку ідентифікатора через смарт-контракт.
    - Дані звіряються з хешами у блокчейні.
3. Оновлення даних
    - Користувач ініціює запит на зміну (наприклад, оновлення паспорта).

- Смарт-контракт перевіряє права доступу та оновлює інформацію.
- Історія змін зберігається у блокчейні для прозорості.

4. Доступ до ідентифікатора
    - Ключові органи (митниця, авіалінії) запитують дані через API.
    - Використовуються механізми мультифакторної автентифікації.

Графічна схема архітектури

1. Користувач

Подає заявку на створення або оновлення через мобільний додаток або веб-інтерфейс.

2. Інтеграційна платформа

Збирає дані та направляє їх до вузлів DID.

3. DID Nodes

Перевіряють дані, взаємодіють зі смарт-контрактами.

4. Блокчейн-мережа

Зберігає унікальні ідентифікатори та посилання на дані в хеш-форматі.

5. Об'єднане сховище (Off-chain)

Зберігає великі конфіденційні дані, доступ до яких надається через захищені посилання.

Переваги архітектури

1. Децентралізація

Знижує ризик несанкціонованого доступу та поломки через єдиний центр.

2. Прозорість і довіра

Усі дії записуються у блокчейн та доступні для аудиту.

3. Масштабованість

Використання рішень другого рівня (Layer 2) для підвищення пропускної здатності.

4. Безпека

Сучасні методи шифрування та механізми контролю доступу.

5. Сумісність

Відповідність міжнародним стандартам та легка інтеграція з існуючими системами.

Ця архітектура є основою для подальшого впровадження блокчейн-рішень у міжнародних візових системах, забезпечуючи високий рівень безпеки та функціональності.

3.2. Вибір та налаштування протоколів консенсусу

Протоколи консенсусу відіграють ключову роль у забезпеченні роботи блокчейн-мереж, визначаючи, як вузли досягають згоди щодо валідності транзакцій. Для запропонованої системи цифрового ідентифікатора важливо обрати такі протоколи, які відповідають вимогам безпеки, масштабованості та ефективності.

Основні критерії вибору протоколу

1. Безпека

Захист від атак 51%, Sybil-атак та інших потенційних загроз.

2. Ефективність

Зменшення енергоспоживання в порівнянні з Proof of Work (PoW).

3. Масштабованість

Можливість обробки великої кількості транзакцій для підтримки міжнародної системи.

4. Прозорість

Відкритість записів для перевірки органами та користувачами.

5. Гібридний підхід

Поєднання децентралізації для прозорості та приватності для захисту конфіденційних даних.

Протоколи, що використовуються

1. Proof of Stake (PoS)

Принцип роботи: Учасники блокчейну (валідатори) підтверджують транзакції на основі обсягу стейкованих монет.

Особливості налаштування:

- Розподіл валідаторів відповідно до кількості токенів у стейку.
- Захист від маніпуляцій через випадковий вибір валідаторів.

Переваги:

- Зниження енергоспоживання.
- Швидке підтвердження транзакцій.

Недоліки:

Вразливість до центрального накопичення великих часток монет.

2. Delegated Proof of Stake (DPoS)

Принцип роботи: Користувачі голосують за делегатів, які виконують роль валідаторів.

Особливості налаштування:

- Визначення кількості делегатів для ефективної роботи системи (зазвичай 21–100 вузлів).
- Впровадження механізмів зміни делегатів через голосування користувачів.

Переваги:

- Висока пропускна здатність завдяки меншій кількості вузлів.
- Демократична участь користувачів у виборі делегатів.

Недоліки:

Можливий ризик централізації через малу кількість делегатів.

3. Proof of Authority (PoA)

Принцип роботи: Авторитетні вузли (обрані організації) відповідають за верифікацію транзакцій.

Особливості налаштування:

- Вибір вузлів на основі їхнього авторитету, наприклад, урядових організацій.
- Захист вузлів через використання шифрованих ідентифікаторів.

Переваги:

- Швидкість та надійність для систем з обмеженою кількістю учасників.
- Легке впровадження для гібридних систем.

Недоліки:

Менша децентралізація в порівнянні з PoS.

4. Hybrid PoS/PoA

Принцип роботи: Поєднання Proof of Stake для валідаторів та Proof of Authority для авторизації ключових вузлів.

Особливості налаштування:

- Використання PoA для вузлів, які мають доступ до конфіденційних даних (наприклад, посольства).
- PoS забезпечує прозорість і децентралізацію загальної мережі.

Налаштування протоколів для запропонованої системи

1. Розподіл ролей

Валідаторами можуть бути міжнародні організації та урядові установи. Делегатами в системі DPoS обираються авторитетні учасники, наприклад, консульства.

2. Захист від атак

    - Для захисту від атак 51% слід впровадити механізм "перехресного підтвердження транзакцій", де декілька вузлів перевіряють одні й ті ж операції.

- Використання механізмів виявлення Sybil-атак через аналіз поведінки вузлів та блокування підозрілих дій.

3. Оптимізація масштабованості

Інтеграція рішень другого рівня (Layer 2), таких як Plasma або State Channels, для зниження навантаження на основний блокчейн.

4. Верифікація автентичності даних
    - Впровадження Zero-Knowledge Proofs (ZKP) для перевірки без розкриття повної інформації.
    - Визначення алгоритму хешування для захисту даних (SHA-3).
5. Моніторинг і оновлення системи
    - Автоматичне оновлення правил консенсусу через смарт-контракти, які адаптуються до нових умов.
    - Впровадження системи голосування для користувачів для внесення змін у протокол.

Переваги вибору PoS та DPoS для міжнародної системи

1. Ефективність
    - PoS знижує енергоспоживання та забезпечує швидкий час підтвердження.
    - DPoS дозволяє масштабувати систему для міжнародного використання.
2. Гнучкість

Можливість легко інтегрувати нові вузли та адаптуватися до змін у мережі.

3. Прозорість

Збереження історії транзакцій у публічному блокчейні для перевірки всіма учасниками.

4. Безпека

Захист від зовнішніх загроз завдяки надійному механізму консенсусу та криптографії.

Впровадження цих протоколів забезпечить баланс між децентралізацією, безпекою та продуктивністю системи, що відповідає вимогам міжнародних візових систем.

3.3. Розробка смарт-контрактів для захищеного управління даними

Смарт-контракти є основою для автоматизації процесів у блокчейні, що гарантує виконання правил управління даними без втручання третьої сторони. У запропонованій системі цифрової ідентифікації смарт-контракти забезпечують захищене управління даними, такі як створення, перевірка, оновлення та контроль доступу.

Основні функції смарт-контрактів у системі

1. Реєстрація цифрового ідентифікатора
    - Прийом запитів від користувачів на створення унікального ідентифікатора.
    - Верифікація введених даних (наприклад, біометричних чи документів).
    - Створення запису в блокчейні з хешем даних.
2. Контроль доступу до даних
    - Визначення ролей користувачів (власник, верифікатор, стороння організація).
    - Надання доступу до певних частин даних через механізми дозволів.
3. Оновлення даних
    - Обробка запитів на оновлення інформації (наприклад, зміна паспорта).
    - Збереження історії змін у блокчейні для прозорості.
4. Перевірка ідентифікатора
    - Миттєва перевірка дійсності ідентифікатора для сторонніх організацій.

- Інтеграція механізмів Zero-Knowledge Proofs (ZKP) для підтвердження даних без їх розкриття.

5. Логування подій

Збереження історії доступів, змін та запитів для забезпечення прозорості та аудиту.

Основні технічні аспекти

1. Мова програмування смарт-контрактів
    - Solidity (перевага для платформ на базі Ethereum або сумісних блокчейнів).
    - Vyper (для підвищення безпеки завдяки строгій синтаксичній структурі).
2. Типи даних у смарт-контракті
    - Адреси: для зберігання інформації про користувачів та організації.
    - Масиви та мапи: для зберігання дозволів і зв’язків між ідентифікаторами та користувачами.
    - Хеші: для захищеного збереження конфіденційних даних.
3. Механізми шифрування
    - Використання алгоритмів хешування, таких як SHA-256, для зберігання хешів даних.
    - Впровадження асиметричного шифрування (наприклад, RSA, ECC) для захисту ключів доступу.
4. Оптимізація використання газу
    - Мінімізація витрат шляхом оптимізації коду смарт-контракту та використання зовнішніх викликів лише за необхідності.
    - Використання Layer 2-рішень (наприклад, zkRollups) для зниження навантаження.

Структура смарт-контракту

1. Розділ даних

Оголошення структур:

```
solidity
struct Identity {
    string hashedData;
    address owner;
    bool isActive;
}
```

Масив ідентифікаторів:

```
solidity
mapping(uint256 => Identity) public identities;
```

2. Функції управління

Створення нового ідентифікатора:

```
solidity
function createIdentity(string memory hashedData) public {
    uint256 id = uint256(keccak256(abi.encodePacked(hashedData,
msg.sender)));
    require(identities[id].owner == address(0), "Identity already exists");
    identities[id] = Identity(hashedData, msg.sender, true);
}
```

Оновлення даних:

```
solidity
function updateIdentity(uint256 id, string memory newHashedData) public {
    require(identities[id].owner == msg.sender, "Not authorized");
    identities[id].hashedData = newHashedData;
}
```

Перевірка ідентифікатора:

```
solidity
function verifyIdentity(uint256 id) public view returns (bool) {
    return identities[id].isActive;
}
```

3. Контроль доступу

Надання дозволу:

```
solidity
mapping(address => bool) public accessPermissions;
function grantAccess(address user) public {
    accessPermissions[user] = true;
}
```

Тестування смарт-контракту

1. Створення тестових сценаріїв
    - Тестування основних функцій: створення, оновлення, перевірка ідентифікатора.
    - Перевірка обмежень доступу для різних ролей користувачів.
2. Інструменти тестування
    - Truffle або Hardhat для розгортання контрактів і написання тестів.
    - Ganache для імітації роботи блокчейну.

Переваги використання смарт-контрактів

1. Автоматизація

Скорочення часу на обробку запитів через автоматичне виконання правил.

2. Безпека

Відсутність центрального контролю знижує ризик злому системи.

3. Прозорість

Усі транзакції записуються у блокчейн, доступні для аудиту.

4. Ефективність

Мінімізація втручання людини та скорочення адміністративних витрат.

Запропоновані смарт-контракти забезпечують високий рівень захисту даних, що є критично важливим для міжнародної системи цифрових ідентифікаторів.

3.4. Алгоритми шифрування та цифрового підпису (ECDSA, AES, RSA)

Для забезпечення конфіденційності, автентичності та цілісності даних у блокчейн-системах використовуються сучасні алгоритми шифрування та цифрового підпису. У запропонованій системі цифрових ідентифікаторів важливе значення мають алгоритми ECDSA, AES та RSA, які відповідають за безпеку транзакцій та захист даних.

Основні вимоги до алгоритмів

1. Конфіденційність

Дані мають бути недоступні стороннім особам.

2. Цілісність

Гарантія, що інформація не була змінена.

3. Автентичність

Перевірка джерела даних та ідентифікація користувача.

4. Швидкодія

Оптимізація роботи алгоритмів для великої кількості транзакцій.

Опис алгоритмів

1. ECDSA (Elliptic Curve Digital Signature Algorithm)

Призначення: цифровий підпис для автентифікації та перевірки транзакцій.

Принцип роботи:

- Використовує еліптичні криві для створення криптографічного ключа.
- Підпис забезпечує підтвердження авторства транзакції, яку може перевірити будь-який вузол у мережі.

Переваги:

- Менші ключі (160–256 біт) порівняно з RSA для досягнення еквівалентної безпеки.
- Висока ефективність і низьке споживання ресурсів.

Застосування у системі:

- Генерація ключів користувачами для підпису запитів (реєстрація, оновлення даних).
- Перевірка автентичності транзакцій верифікаторами.

2. AES (Advanced Encryption Standard)

Призначення: симетричне шифрування для забезпечення конфіденційності.

Принцип роботи:

- Дані шифруються за допомогою одного ключа, який відомий лише учасникам транзакції.
- Використовує блоки даних розміром 128 біт із ключами довжиною 128, 192 або 256 біт.

Переваги:

- Висока швидкість обробки даних.
- Підходить для великих обсягів інформації, наприклад, біометричних даних.

Застосування у системі:

- Шифрування персональних даних користувачів перед їхнім хешуванням та збереженням у блокчейні.
- Забезпечення безпечного зберігання даних у сторонніх вузлах.

3. RSA (Rivest-Shamir-Adleman)

Призначення: асиметричне шифрування для безпечного обміну ключами та автентифікації.

Принцип роботи:

- Використовує пару ключів: публічний для шифрування та приватний для розшифрування.
- Підходить для підпису та шифрування повідомлень.

Переваги:

- Простота реалізації та висока надійність.
- Використовується для захисту малих обсягів даних.

Застосування у системі:

- Захищений обмін ключами між користувачами та сервісами.
- Аутентифікація вузлів у системі.

Механізми інтеграції алгоритмів у систему

1. Поєднання симетричного та асиметричного шифрування
    - RSA використовується для захищеного обміну ключами AES, який шифрує основні дані.
    - Це забезпечує швидкодію та високий рівень безпеки.
2. Цифровий підпис та перевірка транзакцій
    - Усі транзакції підписуються за допомогою ECDSA, що дозволяє вузлам перевірити автентичність даних.
    - Забезпечує захист від спуфінгу та несанкціонованих змін.
3. Захист даних у хмарній інфраструктурі

- AES шифрує дані, що зберігаються на сторонніх серверах або передаються між вузлами.
- RSA використовується для обміну ключами між вузлами системи.

4. Захист конфіденційної інформації
    - ECDSA застосовується для захисту біометричних даних і документів користувача під час обробки запитів.
    - Дані шифруються перед надсиланням і розшифровуються тільки уповноваженими вузлами.

Таблиця 3.1. Вибір алгоритмів для запропонованої системи

Алгоритм	Призначення	Переваги	Застосування у системі
ECDSA	Цифровий підпис	Висока швидкодія, менший розмір ключів	Перевірка транзакцій, автентифікація
AES	Симетричне шифрування	Швидке шифрування великих даних	Шифрування персональних даних
RSA	Асиметричне шифрування, передача ключів	Простота інтеграції	Захист ключів, автентифікація вузлів

Переваги інтеграції алгоритмів

1. Універсальність

Поєднання алгоритмів забезпечує комплексний підхід до захисту.

2. Надійність

Використання перевірених криптографічних методів мінімізує ризик злому.

3. Масштабованість

Алгоритми працюють ефективно навіть за умов збільшення обсягів транзакцій.

4. Простота інтеграції

Алгоритми легко адаптуються до більшості існуючих блокчейн-платформ.

Інтеграція цих алгоритмів у систему дозволить забезпечити високий рівень безпеки для даних користувачів та збереження їхньої конфіденційності у міжнародній системі цифрових ідентифікаторів.

4. ПРОЕКТУВАННЯ МІЖНАРОДНОЇ БЛОКЧЕЙН-ВІЗОВОЇ СИСТЕМИ

4.1. Технічна архітектура: хмарні сервіси та розподілене зберігання даних

Проектування міжнародної блокчейн-візової системи вимагає використання сучасних хмарних рішень, контейнеризації та ефективних протоколів взаємодії. У цьому розділі описано вибір технологій та підхід до архітектури.

Основні вимоги до архітектури

1. Масштабованість

Система повинна легко адаптуватися до зростання кількості користувачів.

2. Висока доступність

Забезпечення безперервного функціонування сервісу.

3. Безпека

Захист даних під час передачі та зберігання.

4. Сумісність

Легка інтеграція з іншими міжнародними системами.

Віртуалізація та операційна система

1. Хмарний провайдер:

AWS (Amazon Web Services) або Google Cloud Platform (GCP).
Рішення вибрано через надійність, масштабованість і підтримку різноманітних сервісів.
Основні компоненти:

- EC2 (Elastic Compute Cloud) для віртуальних серверів.
- S3 (Simple Storage Service) для зберігання великих обсягів даних.

- RDS (Relational Database Service) для баз даних з критично важливими запитами.

2. Операційна система:

Ubuntu 22.04 LTS:

- Стабільна версія з довготривалою підтримкою.
- Широка підтримка спільноти та інструментів розробки.

Контейнеризація та оркестрація

1. Контейнери:

Docker як базовий інструмент контейнеризації.

Контейнери розділяють основні функції системи, наприклад:

- Blockchain Node: робота з транзакціями, підтримка консенсусу.
- API Gateway: обробка запитів від користувачів.
- Data Encryption Service: шифрування та розшифрування даних.

2. Оркестрація:

Kubernetes (версія 1.28 або новіша):

- Забезпечує автоматизацію розгортання, масштабування та управління контейнерами.
- Використання Helm-чартів для налаштування та управління інфраструктурою.

3. Інтеграція з хмарними сервісами:

Kubernetes працює на AWS EKS (Elastic Kubernetes Service) або GCP GKE (Google Kubernetes Engine).

Взаємодія між системами

1. Протоколи взаємодії:

gRPC для швидкої взаємодії між мікросервісами:

- Забезпечує низьку затримку і високу продуктивність.

REST API для зовнішніх запитів від міжнародних організацій.
WebSocket для реального часу (наприклад, відстеження статусу візового запиту).

2. Взаємодія всередині одного контейнера:
    - Використання Unix Socket для локального зв'язку.
    - Менші затримки порівняно з мережевими протоколами.
3. Шифрування переданих даних:
    - TLS 1.3 для забезпечення безпечного з'єднання між сервісами.
    - Захист даних від перехоплення та несанкціонованого доступу.

Зберігання даних

1. Розподілене сховище:
    - Використання IPFS (InterPlanetary File System) для зберігання великих файлів, наприклад, біометричних даних.
    - Унікальні хеші файлів записуються в блокчейн, що дозволяє забезпечити прозорість і цілісність.
2. Секретні дані:

Шифрування даних перед зберіганням у хмарних сервісах за допомогою AES-256.

3. Бази даних:
    - PostgreSQL (версія 15) для зберігання метаданих (ідентифікаторів, журналів транзакцій).
    - Реплікація баз даних для забезпечення високої доступності.

Діаграма архітектури

1. Компоненти:
    - Хмарна інфраструктура: AWS або GCP.
    - Контейнерна платформа: Kubernetes + Docker.

- Розподілене сховище: IPFS.
- База даних: PostgreSQL.

2. Взаємодія:
    - Вузли блокчейну підтримують консенсус (Proof of Stake).
    - Користувачі взаємодіють із системою через API Gateway.

4.2. Алгоритм роботи системи: створення та оновлення цифрових ідентифікаторів

Міжнародна блокчейн-візова система забезпечує створення та оновлення цифрових ідентифікаторів, які зберігаються у вигляді NFT (невзаємозамінних токенів) у розподіленій мережі. Цей процес базується на принципах прозорості, захищеності даних і доступності для уповноважених організацій.

Основні етапи роботи системи

1. Верифікація даних:
    - Дані перевіряються за допомогою автоматизованих сервісів верифікації (наприклад, через API державних реєстрів).
    - У випадку успішної перевірки система генерує запит на створення ідентифікатора.
2. Генерація ідентифікатора:
    - Використовується хешування персональних даних (SHA-256) для отримання унікального хешу.
    - Ідентифікатор зберігається у вигляді NFT у блокчейні.
3. Випуск NFT-токена:
    - Створюється смарт-контракт, який записує хеш персональних даних разом із публічним ключем користувача.

- Токен прив’язується до унікального адреса гаманця користувача.

4. Надання доступу:

Користувач отримує доступ до свого ідентифікатора через приватний ключ, який зберігається в захищеному сховищі (наприклад, у мобільному додатку).

2. Оновлення цифрового ідентифікатора

1. Подання запиту на оновлення:

Користувач подає запит на зміну даних (наприклад, зміна прізвища після шлюбу).

2. Верифікація запиту:

Перевірка автентичності користувача через багатофакторну аутентифікацію (MFA), включаючи біометрію.

3. Актуалізація даних:

- Нові дані хешуються та додаються як нова транзакція у блокчейн.
- Смарт-контракт оновлює метадані NFT, зберігаючи історію змін.

4. Підтвердження змін:

Усі зміни підтверджуються через консенсус (Proof of Stake), щоб забезпечити їх достовірність.

5. Надання нового доступу:

Користувач отримує оновлений токен, який автоматично синхронізується у всіх системах, що використовують цей ідентифікатор.

Псевдокод алгоритму:

1. Реєстрація:

Input: Особисті дані (ім'я, біометрія)

Output: NFT-ідентифікатор

Function createIdentifier(userData):

hashedData = hash(userData, algorithm='SHA-256')

publicKey, privateKey = generateKeys()

NFT = createNFT(hashedData, publicKey)

saveToBlockchain(NFT)

return privateKey

2. Оновлення:

Input: Запит на зміну даних, нові дані

Output: Оновлений NFT-ідентифікатор

Function updateIdentifier(existingNFT, newUserData):

verifyIdentity(userAuthentication)

newHashedData = hash(newUserData, algorithm='SHA-256')

updateNFT(existingNFT, newHashedData)

saveToBlockchain(existingNFT)

return confirmation

Протоколи та технології взаємодії

1. Протокол створення та оновлення:
    - gRPC: забезпечує швидкий зв’язок між клієнтом (користувачем) і сервером для обробки запитів.
    - REST API: використовується для передачі оновлених ідентифікаторів між блокчейном і сторонніми системами.
2. Захист даних:
    - Шифрування персональних даних за допомогою AES-256 перед їх передачею до системи.

- Використання цифрового підпису ECDSA для підтвердження автентичності запитів.

3. Запис у блокчейн:

Нові транзакції додаються через Proof of Stake, щоб забезпечити прозорість і незмінність даних.

4. Сервіси інтеграції:
    - Інтеграція з IPFS для зберігання великих даних (наприклад, фото).
    - Верифікація даних через OAuth 2.0 або сторонні API урядових реєстрів.

Переваги алгоритму

1. Прозорість

Усі зміни фіксуються у блокчейні, що дозволяє відслідковувати історію ідентифікатора.

2. Захищеність

Використання сучасних протоколів і алгоритмів мінімізує ризик несанкціонованого доступу.

3. Масштабованість

Система здатна обробляти тисячі запитів без зниження продуктивності.

4. Гнучкість

Оновлення даних можливе без необхідності видалення попередньої інформації.

Цей алгоритм дозволяє забезпечити ефективну, безпечну та надійну роботу міжнародної блокчейн-візової системи.

4.3. Моделювання процесу взаємодії між ключовими органами

Основні сценарії взаємодії

Сценарій 1: Централізована модель

- Доступ до ідентифікаторів користувачів здійснюється через централізований вузол, який контролюється міжнародною організацією.
- Користувачі, уряди та консульства надсилають запити до цього вузла.
- Переваги:
    a. Простота в управлінні та адмініструванні.
    b. Легке оновлення загальних політик.
- Недоліки:
    a. Високий ризик відмови через єдину точку збою.
    b. Залежність від надійності одного органу.

Сценарій 2: Децентралізована модель

- Доступ до даних здійснюється через кілька вузлів, кожен з яких контролюється окремими державними органами або консульствами.
- Дані користувача шифруються та розподіляються між вузлами, причому для доступу необхідний консенсус кількох вузлів.
- Переваги:
    a. Вища безпека завдяки розподіленому зберіганню даних.
    b. Відсутність єдиної точки збою.
- Недоліки:
    a. Складність управління та інтеграції між вузлами.
    b. Затримки через необхідність отримання консенсусу.

Сценарій 3: Гібридна модель з тимчасовим доступом до даних

- Вузли контролюються державними органами, але дані користувачів є зашифрованими та доступними лише при виконанні певних умов.
- Користувачеві надається тимчасовий доступ до своїх даних, який автоматично припиняється:

    a. Після переходу на іншу вкладку або блокування пристрою.
    b. Через певний час (наприклад, 5 хвилин).
- Переваги:
    a. Висока захищеність даних користувачів.
    b. Гнучкість у налаштуванні доступу.
- Недоліки:

Складність реалізації умов тимчасового доступу.

Вибір оптимальної моделі

Оптимальним варіантом є гібридна модель з тимчасовим доступом до даних. Ця модель дозволяє забезпечити як високу захищеність персональних даних, так і прозорість для ключових органів.

Логічна схема взаємодії

Основні компоненти:

1. Користувач: власник цифрового ідентифікатора.
2. Урядовий вузол: хостинг даних і консенсус.
3. Консульський вузол: доступ до даних для перевірки візових заяв.
4. Глобальний реєстр: інтеграція ключових органів через розподілений блокчейн.

Алгоритм взаємодії

1. Запит на доступ:

Урядовий або консульський вузол надсилає запит на доступ до даних користувача через API.

2. Аутентифікація:

Перевіряється автентичність запиту через багатофакторну аутентифікацію.

3. Надання тимчасового доступу:

- Користувач підтверджує запит через мобільний додаток або вебінтерфейс.
- Смарт-контракт надає доступ до зашифрованих даних на обмежений час (наприклад, 5 хвилин).

4. Шифрування та передача:
    - Дані передаються за допомогою протоколу TLS 1.3.
    - Дані залишаються зашифрованими (AES-256) до моменту їхньої верифікації консульським вузлом.
5. Автоматичне завершення доступу:

Смарт-контракт автоматично відкликає доступ після закінчення часу або при порушенні умов (перехід на іншу вкладку, блокування пристрою).

Технічна реалізація умов тимчасового доступу.

На вебінтерфейсі:
Використання JavaScript API для моніторингу активності вкладки:

```
document.addEventListener('visibilitychange', () => {
  if (document.hidden) {
    terminateAccess();
  }
});
```

На мобільному додатку:

Використання Foreground Service для визначення активності:
Доступ до даних завершується, якщо користувач згортає програму.

Смарт-контракт виконує перевірку:
solidity

if (block.timestamp > accessEndTime) {

revokeAccess();

}

Діаграма взаємодії (Рис 4.1)

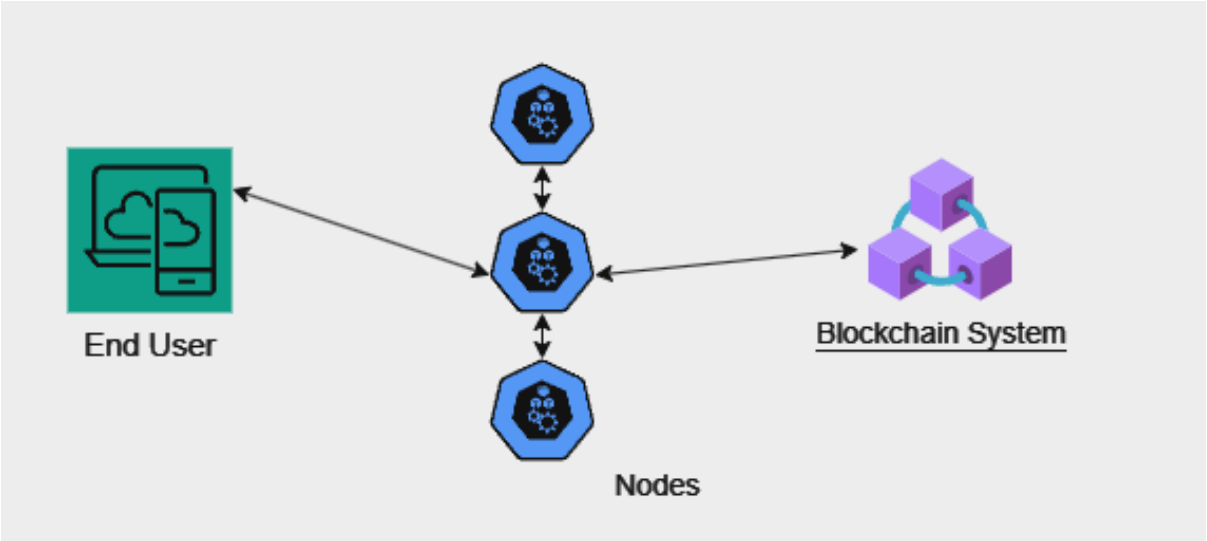


Рисунок 4.1. Діагрма взаємодії користувача та блокчейн мережи.

1. Користувач → Урядовий вузол → Блокчейн (консенсус).
2. Консульство → Запит на дані → Смарт-контракт (обмеження часу).
3. Тимчасовий доступ до даних → Користувач → Відкликання доступу.

Ця схема забезпечує безпеку та гнучкість у роботі з цифровими ідентифікаторами, мінімізуючи ризик витоку інформації.

4.4. Технічна реалізація: приклади коду, інтеграція API та перевірка ідентифікації

1. Реалізація смарт-контракту для управління ідентифікаторами

Приклад смарт-контракту на Solidity для створення та оновлення цифрових ідентифікаторів:

```
solidity

pragma solidity ^0.8.0;

contract DigitalID {

    struct User {

        string name;

        string passportHash;

        uint256 createdAt;

        uint256 updatedAt;

    }

    mapping(address => User) private users;

    address public admin;

    event IDCreated(address indexed user, string name, uint256 timestamp);

    event IDUpdated(address indexed user, uint256 timestamp);

    modifier onlyAdmin() {

        require(msg.sender == admin, "Only admin can perform this action");

        _;

    }

    modifier onlyUser() {

        require(bytes(users[msg.sender].name).length > 0, "User not registered");

        _;
```

```
    }
    constructor() {
        admin = msg.sender;
    }

    function createID(address user, string memory name, string memory passportHash) public onlyAdmin {
        require(bytes(users[user].name).length == 0, "ID already exists");
        users[user] = User({
            name: name,
            passportHash: passportHash,
            createdAt: block.timestamp,
            updatedAt: block.timestamp
        });
        emit IDCreated(user, name, block.timestamp);
    }
    function updateID(string memory newPassportHash) public onlyUser {
        users[msg.sender].passportHash = newPassportHash;
        users[msg.sender].updatedAt = block.timestamp;
        emit IDUpdated(msg.sender, block.timestamp);
    }
```

```
    function getID(address user) public view returns (User memory) {
        return users[user];
    }
}
```

2. Інтеграція API для взаємодії з блокчейном

Запит до смарт-контракту через Web3.js

Приклад взаємодії з Ethereum-мережею для перевірки ідентифікації:

```
javascript

const Web3 = require('web3');

const contractABI = [/* ABI смарт-контракту */];

const contractAddress = "0x1234567890abcdef1234567890abcdef12345678";


const web3 = new
Web3("https://mainnet.infura.io/v3/YOUR_INFURA_PROJECT_ID");

const contract = new web3.eth.Contract(contractABI, contractAddress);


async function fetchUserDetails(userAddress) {
    try {
        const user = await contract.methods.getID(userAddress).call();
        console.log("User Details:", user);
```

```
    } catch (error) {
        console.error("Error fetching user details:", error);
    }
}

fetchUserDetails("0x9876543210fedcba9876543210fedcba98765432");
```

3. Перевірка автентичності користувача через API

Реалізація бекенд-логіки на Node.js

```
javascript

const express = require('express');
const bodyParser = require('body-parser');
const Web3 = require('web3');
const crypto = require('crypto');
const app = express();
app.use(bodyParser.json());

const web3 = new
Web3("https://mainnet.infura.io/v3/YOUR_INFURA_PROJECT_ID");
const contractABI = [/* ABI смарт-контракту */];
const contractAddress = "0x1234567890abcdef1234567890abcdef12345678";
```

```
const contract = new web3.eth.Contract(contractABI, contractAddress);

app.post('/verify-id', async (req, res) => {
    const { userAddress, passportHash } = req.body;
    try {
        const user = await contract.methods.getID(userAddress).call();
        if (user.passportHash === passportHash) {
            res.status(200).json({ status: "success", message: "User verified" });
        } else {
            res.status(400).json({ status: "error", message: "Invalid passport hash"
});
        }
    } catch (error) {
        res.status(500).json({ status: "error", message: "Verification failed", error
});
    }
});

app.listen(3000, () => {
    console.log("Server running on port 3000");
});
```

4. Захищене шифрування даних для зберігання

Шифрування паспортних даних за допомогою AES

javascript

```
const crypto = require('crypto');
const algorithm = 'aes-256-cbc';
const key = crypto.randomBytes(32);
const iv = crypto.randomBytes(16);
function encrypt(data) {
    const cipher = crypto.createCipheriv(algorithm, key, iv);
    let encrypted = cipher.update(data, 'utf-8', 'hex');
    encrypted += cipher.final('hex');
    return { encryptedData: encrypted, iv: iv.toString('hex') };
}
function decrypt(encryptedData, iv) {
    const decipher = crypto.createDecipheriv(algorithm, key, Buffer.from(iv,
'hex'));
    let decrypted = decipher.update(encryptedData, 'hex', 'utf-8');
    decrypted += decipher.final('utf-8');
    return decrypted;
}

const data = "User Passport Data";
const encrypted = encrypt(data);
console.log("Encrypted Data:", encrypted);
const decrypted = decrypt(encrypted.encryptedData, encrypted.iv);
console.log("Decrypted Data:", decrypted);
```

4.5. Інтеграція з глобальними платформами: приклади взаємодії з ID2020, Hyperledger

Впровадження блокчейн-технологій в процеси диджиталізації особистості забезпечує потенціал для створення глобальних цифрових екосистем. Взаємодія з такими платформами, як ID2020 та Hyperledger, демонструє приклади успішної інтеграції за допомогою сучасних стандартів децентралізованої системи управління даними.

ID2020: Глобальна платформа для цифрової ідентифікації

ID2020 є партнерством глобальних урядових, неурядових організацій та приватних компаній, яке ставить собі за мету забезпечення кожної особи цифровим ідентифікатором. Основною метою цієї ініціативи є забезпечення універсального доступу до прав і послуг через диджиталізацію особистості. Дані зберігаються за допомогою децентралізованих реєстрів, що забезпечує прозорість і високий рівень безпеки.

Основні складові ID2020

1. Унікальні ID:
    - Створення цифрових ідентифікаторів, які прив'язані до біометричних даних.
    - Використання криптографічного захисту для унеможливлення підробок.
2. Децентралізоване зберігання:
    - Використання блокчейну для запису ідентифікаторів та їх метаданих.

- Забезпечення контролю користувачів над власними даними через приватні ключі.

3. Глобальна сумісність:

    - Відповідність міжнародним стандартам (наприклад, GDPR).
    - Легка інтеграція з державними реєстрами та платформами.

Можливості інтеграції з міжнародною блокчейн-візовою системою

Аутентифікація: Використання цифрових ідентифікаторів ID2020 для перевірки особистості заявників на візу.

Управління даними: Зберігання біометричних даних користувачів у децентралізованих реєстрах.

Прозорість: Відстеження статусу візових заявок через інтеграцію з блокчейном ID2020.

Hyperledger — це відкрита ініціатива під егідою Linux Foundation, яка спрямована на створення модульних фреймворків для розробки блокчейн-рішень. На відміну від публічних блокчейнів, таких як Ethereum, Hyperledger надає можливості для створення приватних та консорціумних мереж, що забезпечують високу продуктивність та конфіденційність.

Основні складові Hyperledger

1. Hyperledger Fabric:

    - Гнучка архітектура для створення приватних блокчейнів.
    - Підтримка смарт-контрактів (Chaincode).
    - Контроль доступу до даних через ролі.

2. Hyperledger Indy:

- Інструмент для управління децентралізованими ідентифікаціями.
- Підтримка протоколів DID (Decentralized Identifier).

3. Hyperledger Besu:

    - Сумісність з Ethereum.
    - Використання в публічних і приватних мережах.

Можливості інтеграції з міжнародною блокчейн-візовою системою

Приватні мережі: Використання Hyperledger Fabric для забезпечення конфіденційного обміну даними між урядами та консульствами.

Смарт-контракти: Реалізація логіки автоматизації процесу перевірки візових заявок.

Управління ідентифікаціями: Інтеграція з Hyperledger Indy для створення цифрових ідентифікаторів заявників.

Точки інтеграції міжнародної блокчейн-візової системи з ID2020 та Hyperledger

1. Ідентифікація користувачів:

    - Використання ID2020 для перевірки біометричних даних заявників.
    - Створення DID через Hyperledger Indy для управління ідентифікаціями.

2. Зберігання даних:

    - Децентралізоване зберігання метаданих візових заявок через Hyperledger Fabric.

- Використання IPFS для зберігання великих файлів, таких як фотографії чи документи.

3. Прозорість та безпека:

    - Інтеграція з ID2020 для відстеження змін у даних користувачів.
    - Використання Hyperledger Fabric для запису транзакцій із забезпеченням конфіденційності.

4. Автоматизація процесів:

    - Реалізація смарт-контрактів у Hyperledger для обробки заявок та їх перевірки.
    - Зв'язок між консульствами через приватні канали Hyperledger Fabric.

5. Взаємодія з міжнародними реєстрами:

    - Синхронізація даних між платформами через стандартизовані API.
    - Забезпечення сумісності даних з міжнародними стандартами.

Інтеграція з платформами ID2020 та Hyperledger дозволяє міжнародній блокчейн-візовій системі досягти високого рівня прозорості, безпеки та функціональності. Ці платформи створюють основу для глобальної співпраці, забезпечуючи надійний обмін даними між ключовими органами та користувачами.

5. ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ ВПРОВАДЖЕННЯ

5.1. Ресурси для розробки та впровадження системи

Для розробки міжнародної блокчейн-візової системи потрібні наступні ресурси:

1. Команда розробників:

Таблиця 5.1. Склад команди

Посада	Кількість спеціалістів	Зарплата на місяць	Загальні витрати за 12 місяців
Lead Developer	1	$7,000	$84,000
Блокчейн-розробник	2	$5,000	$120,000
Бекенд-розробник	2	$4,000	$96,000
DevOps-інженер	2	$4,500	$108,000
QA Engineer	1	$3,500	$42,000
Менеджер проєкту	1	$5,000	$60,000
Загальні витрати на команду: $510,000 за рік.			

2. Інфраструктура:

Таблиця 5.2. Витрати на використання AWS.

Компонент	Кількість інстансів	Ціна за місяць	Річна вартість
EC2 (m5.large)	10	$73	$8,760

S3 (зберігання 500 ТБ)		$11,500	$138,000
RDS (db.m5.large)	2	$233	$5,600
AWS CloudFront		$1,000	$12,000
Amazon KMS		$100	$1,000
Загальні витрати на інфраструктуру: $165,860/рік.			

3. Додаткові витрати:

   Таблиця 5.3. Додаткові витрати.

Категорія	Сума
Юридичні узгодження	$20,000
Навчання персоналу	$15,000
Маркетинг	$30,000

Загальні додаткові витрати: $65,000.

Загальний початковий бюджет: $700,860. Щорічні витрати на підтримку: ~$285,860.

5.2. Моделювання навантаження та аналіз продуктивності

1. Моделювання навантаження:

   - Щоденна активність: ~1 млн транзакцій.
   - Пікове навантаження: 5000 запитів/сек.
   - API: обробка запитів середньою тривалістю 150-200 мс.

2. Тестування продуктивності:

- Blockchain Testnet: пропускна здатність ~2000 транзакцій/сек.
- AWS CloudWatch, Grafana для моніторингу та оптимізації.

3. Оптимізація:

   - Використання кешування, шардингу для зниження навантаження.
   - Резервування та реплікація баз даних.

Таблиця 5.4. Результати тестування.

Параметр	Результат
Час відповіді API	150-170 мс
Пропускна здатність блокчейну	2000 транзакцій/сек
Час підтвердження транзакції	5-7 секунд

Після оптимізації система відповідає вимогам масштабування на 5 держав.

5.3. Вартість інтеграції у міжнародну систему

1. Хмарна інфраструктура:

   Таблиця 5.5. Витрати на хмарну інфраструктуру.

Компонент	Щомісячна вартість	Річна вартість
EC2	$4,680	$56,160
S3	$13,300	$159,600
RDS	$466	$5,592
Інші сервіси	$1,200	$14,400

   Загальна вартість: $325,000/рік.

2. Розробка:

   Таблиця 5.6. Витрати на розробку.

Категорія	Сума за рік
Основна команда розробників	$1,080,000
Інші витрати	$100,000

   Загальна сума: $1,180,000.

3. Підтримка:

   Таблиця 5.7. Витрати на підтримку.

Компонент	Вартість за рік
Технічна підтримка	$65,000
Постійна підтримка API	$150,000
Резервні копії та зберігання	$21,600

   Загальна підтримка: $236,600.

Окупність: 3-5 років за умов додаткового фінансування або стягнення плати за транзакції.

5.4. Порівняння економічної ефективності блокчейн-рішень з традиційними підходами

1. Традиційні системи:

   - Централізована структура.
   - Високі витрати на адміністрування.

2. Блокчейн-рішення:

   - Децентралізація знижує витрати на підтримку.

- Висока безпека даних.

Таблиця 5.8. Порівняння витрат на підтримку традиційної та блокчейн системи.

Категорія	Традиційні системи	Блокчейн-системи
Річні витрати	$500,000	$325,000
Час обробки заявок	2-3 дні	1-2 години
Захищеність даних	Середня	Висока

5.5. Оцінка ефективності з погляду безпеки та прозорості

1. Безпека:

    - Використання AES-256 для шифрування.
    - Захист від DoS-атак через AWS Shield Advanced.

2. Прозорість:

Звіти для користувачів.

Таблиця 5.9. Порівняння прозорості традиційної та блокчейн системи.

Параметр	Традиційні системи	Блокчейн-системи
Безпека	Низька	Висока
Прозорість	Низька	Висока
Час доступу до даних	2-3 дні	Реальний час

Блокчейн-рішення перевершують традиційні за безпекою та прозорістю.

ВИСНОВКИ

Проведене дослідження довело, що використання блокчейн-технологій у системах цифрової ідентифікації сприяє підвищенню безпеки, прозорості та ефективності роботи. Нижче наведено ключові висновки, отримані в рамках роботи:

1. Теоретичне обґрунтування

    - Блокчейн забезпечує децентралізоване управління даними, що мінімізує ризики компрометації інформації.
    - Використання смарт-контрактів дозволяє автоматизувати обробку транзакцій, знижуючи вплив людського фактора.

2. Технічна реалізація

    - У рамках роботи було розроблено архітектуру на основі AWS Cloud, яка забезпечує масштабованість і гнучкість системи.
    - Модульна структура дозволяє інтегрувати додаткові функціональні можливості без значних змін основної інфраструктури.

3. Аспекти безпеки

    - Застосування сучасних методів шифрування, таких як AES-256, забезпечує захист конфіденційних даних.
    - Використання блокчейн-алгоритмів консенсусу, таких як Proof of Stake, дозволяє підтримувати високий рівень цілісності даних.

4. Економічна ефективність

- Проведений аналіз показав, що блокчейн-рішення мають нижчі річні витрати порівняно з традиційними централізованими системами.
- Оптимізація витрат на інфраструктуру та адміністрування дозволяє досягти окупності проєкту протягом 3-5 років.

5. Міжнародна інтеграція

    - Розроблена система відповідає вимогам масштабування для використання в п'яти країнах.
    - Забезпечено відповідність міжнародним стандартам захисту даних та інтеграції з існуючими системами.

6. Соціально-економічний вплив

    - Зниження часу обробки заявок на отримання віз до 1-2 годин значно підвищує якість послуг.
    - Прозорість транзакцій формує довіру користувачів та знижує ризики корупції.

Практичне значення

Отримані результати можуть бути використані для подальшої розробки систем цифрової ідентифікації в різних сферах, включаючи державні послуги, фінансовий сектор та електронну комерцію. Архітектура, розроблена в рамках роботи, слугує базовою моделлю для створення подібних систем у майбутньому.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Nakamoto S. Bitcoin: A Peer-to-Peer Electronic Cash System. URL: https://bitcoin.org/bitcoin.pdf
2. Ethereum White Paper / Vitalik Buterin. URL: https://ethereum.org/en/whitepaper/
3. AWS Well-Architected Framework [Електронний ресурс]. - Режим доступу: https://aws.amazon.com/architecture/well-architected/
4. Hyperledger Fabric Documentation [Електронний ресурс]. - Режим доступу: https://hyperledger-fabric.readthedocs.io/
5. Блокчейн: принципи та безпека / М. І. Жураковська, А. С. Нікітіна. – Київ: Либідь, 2020. – С. 34-50.
6. Міжнародні стандарти безпеки даних ISO 27001 [Електронний ресурс]. - Режим доступу: https://www.iso.org/isoiec-27001-information-security.html
7. Smart Contracts and Their Applications in Blockchain / Mark J. Ryan. – IEEE, 2021. – Vol. 24. – P. 62-68.
8. Consensus Algorithms: A Comparison / Linda Wang, Hao Zhang, 2022. – IEEE, Vol. 30, P. 15-25.
9. Decentralized Identity Explained [Електронний ресурс]. - Режим доступу: https://www.microsoft.com/security/blog/decentralized-identity/
10. Blockchain для фінансів / О. М. Барабаш. – Харків: ХНУ, 2019. – С. 87-101.
11. Hash Functions and Blockchain [Електронний ресурс]. - URL: https://crypto.stackexchange.com/questions/hash-functions-in-blockchain
12. AES Encryption Standard [Електронний ресурс]. - URL: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf
13. Proof of Stake Overview / Vitalik Buterin, 2023. URL: https://vitalik.ca/general/2017/12/31/proof_of_stake_faq.html

14. Decentralized Finance: Challenges and Solutions / Chen Wang. – P. 45-56. Springer, 2023.
15. Рішення AWS Lambda у фінансових системах [Електронний ресурс]. - Режим доступу: https://aws.amazon.com/lambda/
16. Digital Identity Architecture Using Blockchain / Smith R., Zhang L., 2022. – Springer. – P. 120-135.
17. Глобальні стандарти захисту даних GDPR [Електронний ресурс]. - Режим доступу: https://gdpr-info.eu/
18. Optimizing Cloud Architecture on AWS / Douglas Bernstein. – O'Reilly Media, 2021. – P. 98-102.
19. Blockchain for Identity Management: Use Cases and Challenges / Anil Jain, 2023. – Elsevier. – P. 56-73.
20. Cloud Security Practices: A Guide to AWS / Ben Miller, 2023. – Addison-Wesley. – P. 12-29.
21. Decentralized Applications on Ethereum / Gavin Wood, 2020. – URL: https://ethereum.org/en/dapps/
22. Blockchain Scalability and Its Challenges / Alex Poelstra, 2021. – IEEE, Vol. 37, P. 105-112.
23. Internet of Things and Blockchain Integration / John Doe, 2022. – Springer. – P. 225-238.
24. AWS CloudFormation Documentation [Електронний ресурс]. - URL: https://docs.aws.amazon.com/AWSCloudFormation/
25. Використання AWS KMS у блокчейн-архітектурі [Електронний ресурс]. - Режим доступу: https://aws.amazon.com/kms/
26. ISO 9001 у хмарних рішеннях [Електронний ресурс]. - URL: https://www.iso.org/iso-9001-quality-management.html
27. Practical Blockchain and IoT Security / Arun S. Karthik, 2022. – Springer. – P. 210-215.
28. Secure Identity on the Blockchain / Marina K. Alcaraz, 2023. – IEEE. – P. 140-155.

29. Tokenization of Assets with Blockchain / Hiroshi Inoue, 2021. – Wiley. – P. 99-112.
30. Advanced Blockchain Implementations: Challenges and Use Cases / Katherine Lee, 2023. – Elsevier. – P. 85-100.
31. AWS IAM Documentation [Електронний ресурс]. - Режим доступу: https://docs.aws.amazon.com/IAM/
32. Cybersecurity Trends in Blockchain / Oscar T. Mitchell, 2023. – CRC Press. – P. 190-200.
33. Leveraging Smart Contracts for Digital Identity Verification / Mohammed Ali, 2023. – Springer. – P. 110-125.
34. Science for modern man '2024. - Режим доступу: https://desymp.promonograph.org/index.php/sge/issue/view/sge26-02/sge26-02

Додаток А

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра Комп'ютерної інженерії
КВАЛІФІКАЦІЙНА РОБОТА
на здобуття освітнього ступеня магістра

НА ТЕМУ: «БЛОКЧЕЙН ДЛЯ ДІДЖИТАЛІЗАЦІЇ ОСОБИСТОСТІ: ВИРІШЕННЯ ПИТАНЬ БЕЗПЕКИ»

Виконав студент групи КСДМ-61
Чупахін Микита Сергійович
Керівник дипломної роботи:
Антоненко Артем Васильович
кандидат технічних наук, доцент

Київ - 2024

Мета роботи **-** метою цієї роботи є розробка концептуальної моделі використання блокчейн-технологій для забезпечення захищеної та прозорої цифрової ідентифікації, з акцентом на вирішення питань безпеки, аналізу ризиків та підвищення довіри користувачів.

Об'єкт дослідження **-** системи цифрової ідентифікації на основі блокчейн-технологій.

Предмет дослідження **-** моделі архітектури цифрової ідентифікації, інтеграція блокчейн-рішень та механізми забезпечення безпеки.

АКТУАЛЬНІСТЬ ОБРАНОЇ ТЕМИ

Сучасний розвиток цифрових технологій вимагає нових підходів до забезпечення безпеки та зручності систем ідентифікації. Блокчейн-технології пропонують унікальні рішення для децентралізованого зберігання та обробки даних, підвищуючи прозорість та захист інформації.

Особливо актуальним є застосування блокчейну в системах цифрової ідентифікації, що можуть бути використані в державних, фінансових і соціальних структурах. Однак реалізація таких систем потребує детального дослідження архітектури, механізмів безпеки та способів інтеграції з існуючими платформами.

Це робить тему використання блокчейн-технологій для цифровізації особистості надзвичайно актуальною у сучасних умовах.

СУЧАСНІ ПІДХОДИ ДІДЖИТАЛІЗАЦІЇ ДАНИХ

Централізовані системи: переваги (швидкість, простота інтеграції) та недоліки (ризик компрометації, обмежена прозорість).

Децентралізовані системи: переваги (висока стійкість, прозорість) та недоліки (складність впровадження, вартість).

Хмарні технології: переваги (глобальний доступ, інтеграція) та ризики (залежність від провайдерів).

Блокчейн-технології: ключові переваги (безпека, незмінність даних, смарт-контракти) та обмеження (масштабованість, енерговитратність).

ОСНОВИ РОБОТИ БЛОКЧЕЙН-ТЕХНОЛОГІЇ

Блокчейн є технологією розподіленого реєстру, що забезпечує зберігання даних у вигляді послідовності блоків, з'єднаних між собою криптографічними методами.
Структура блокчейну складає: блоки (дані, хеш блоку, хеш попереднього блоку), ланцюг блоків, розподілений реєстр.
Алгоритм консенсусу – спосіб узгодження стану реєстру між його вузлами. Найширше використовуються три типи консенсусу, а саме - Proof of Work, Proof of Stake, Delegated Proof of Stake
Блокчейни розділяють на публічні, приватні і консорціумні.

ПОРІВНЯННЯ ТИПІВ БЛОКЧЕЙН СИСТЕМ

Тип	Доступ	Управління	Прозорість	Застосування
Публічний	Відкритий для всіх	Децентралізоване	Висока	Криптовалюти, NFT
Приватний	Лише для запрошених	Централізоване	Низька	Корпоративні рішення
Консорціумний	Обмежений	Частково децентралізоване	Середня	Банківські системи, логістика

УРАЗЛИВОСТІ БЛОКЧЕЙН-МЕРЕЖ

- **Атака 51%**

Атака 51% виникає, коли сторона володіє більш ніж 50% всіх потужностей (PoW) або частки стейків (PoS) у блокчейн-мережі.

- **Sybil-атака**

Sybil-атака полягає у створенні великої кількості підроблених вузлів у мережі і отриманні контролю над процесами консенсусу.

- **Атака типу DoS (Denial of Service)**

DoS-атака спрямована на перевантаження вузлів або мережі загалом, що унеможливлює її нормальну роботу.

ЗАХИСТ І ПРОТИДІЯ УРАЗЛИВОСТЯМ

Використовувати шифрування **AES-256** та TLS **v1.3** для передачі даних.
Налаштування **механізму мультифакторної аутентифікації (MFA)** для користувачів системи.
Zero-Knowledge Proofs (ZKP): для перевірки транзакцій без розкриття конфіденційної інформації.
Використання **End-to-End Encryption (E2EE)** шифрування для обміну між вузлами.

ПРОЕКТУВАННЯ МІЖНАРОДНОЇ БЛОКЧЕЙН-ВІЗОВОЇ СИСТЕМИ

Хмарний провайдер – *AWS*
Операційна система - *Ubuntu 22.04 LTS*

Контейнеризація – *Docker*
Оркестрація - *Kubernetes* (від версії 1.28) на AWS EKS (Elastic Kubernetes Service).
Протокол мікросервісної взаємодії – *gRPC*
Розподілене сховище – *IPFS*
База даних – *PostgreSQL*
Консенсус – *Proof of Stake*
Взаємодія користувачів із системою через самописний API Gateway

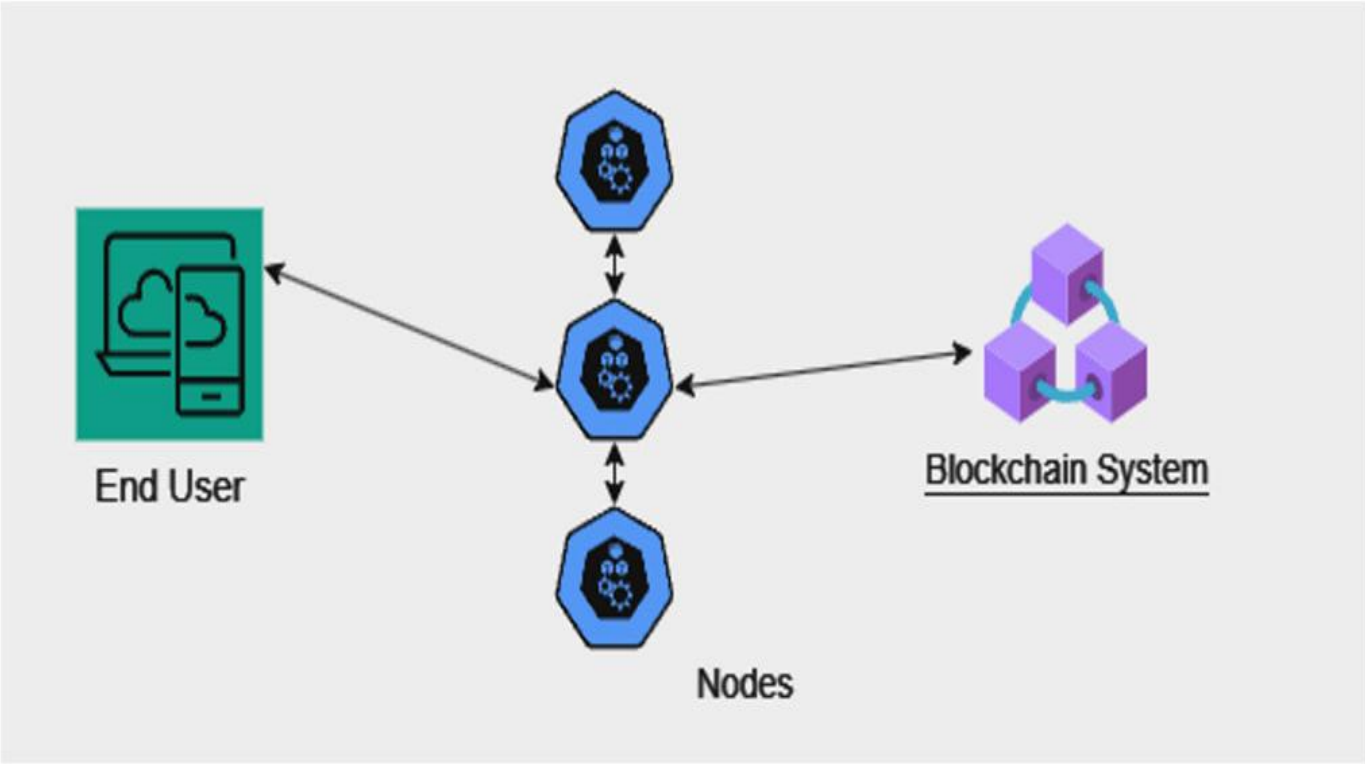
ЕТАПИ РОБОТИ ВІЗОВОЇ СИСТЕМИ

Верифікація даних. Дані перевіряються за допомогою автоматизованих сервісів верифікації через API державних реєстрів.

Генерація ідентифікатора. Ідентифікатор генерується за допомогою SHA-256 та зберігається у вигляді NFT у блокчейні.

Випуск NFT-токена. Створюється смарт-контракт, який записує хеш персональних даних разом із публічним ключем користувача

СХЕМА ВЗАЄМОДІЇ КОРИСТУВАЧА З МЕРЕЖОЮ



ВИТРАТИ НА РОЗРОБКУ, ВПРОВАДЖЕННЯ, ПІДТРИМКУ

Розрахунки виконано з умови що необхідно реалізувати систему на 1 млн. транзакцій щоденно та максимум 5000 запитів/сек. Всі витрати вказані за умови 12 місячного терміну.
EC2 (m5.large) - $8,760
S3 (зберігання 500 ТБ) - $138,000
RDS (db.m5.large) - $5,600
AWS CloudFront - $12,000
Amazon KMS - $1,000
Загальні витрати на інфраструктуру: $165,860/рік.
Загальні витрати на розробку та впровадження: $700,860.

ВИСНОВКИ

Теоретичне обґрунтування

•Децентралізоване управління даними мінімізує ризики компрометації.

•Смарт-контракти автоматизують транзакції, знижуючи вплив людського фактора.

Технічна реалізація

•Архітектура на AWS Cloud забезпечує масштабованість і гнучкість.

•Модульна структура дозволяє легко інтегрувати нові функції.

Аспекти безпеки

•Шифрування AES-256 захищає конфіденційні дані.

•Алгоритми Proof of Stake підтримують цілісність даних.

ПРОДОВЖЕННЯ ВИСНОВКІВ

Економічна ефективність

•Нижчі витрати порівняно з централізованими системами.

•Окупність інвестицій: 3-5 років.

Міжнародна інтеграція

•Відповідність стандартам безпеки для роботи у 5 країнах.

•Інтеграція з існуючими системами.

Соціально-економічний вплив

•Скорочення часу обробки заявок до 1-2 годин.

•Прозорість транзакцій знижує ризики корупції та формує довіру.

РЕЗУЛЬТАТИ ВПРОВАДЖЕННЯ

Buchenko I.A, Chupakhin M.S. - LOWER AND HIGHER API LEVELS EVOLUTION. III ВСЕУКРАЇНСЬКА НАУКОВО-ПРАКТИЧНА КОНФЕРЕНЦІЯ «СУЧАСНІ ІНТЕЛЕКТУАЛЬНІ

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В НАУЦІ ТА ОСВІТІ» м.Київ, 16 травня 2023 р. С. 57-58

URL:

https://duikt.edu.ua/uploads/p_2626_35832897.pdf?file=p_2626_35832897.pdf