

**МІНІСТЕРСТВО ФІНАНСІВ УКРАЇНИ
ДЕРЖАВНИЙ ПОДАТКОВИЙ УНІВЕРСИТЕТ**

**ФАКУЛЬТЕТ ФІНАНСІВ ТА ЦИФРОВИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І
СИСТЕМ**

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Дослідження методів забезпечення приватності транзакцій та користувачів блокчейну»

Здобувача вищої освіти другого (магістерського) рівня,
II курсу, групи КМІ-23-1, спеціальності 122 «Комп'ютерні науки»,
освітня програма «Інформаційні управляючі системи і технології в економіці»



(підпис)

Колеснікова Кирила
Віталійовича

(прізвище, ім'я, по батькові)

Науковий керівник



(підпис)

Редич Олександр
Володимирович

(прізвище, ім'я, по батькові, науковий
ступінь, вчене звання)

Керівник освітньо-
професійної програми



(підпис)

Горбовий Артур Юліанович,
д.т.н, професор

(прізвище, ім'я, по батькові, науковий
ступінь, вчене звання)

Допущено до захисту за рішенням кафедри протокол від «10» січня 2025 №11

Завідувач кафедри



(підпис)

Філоненко Михайло
Миколайович, к.ф.-м.н.,
доцент

(прізвище, ім'я, по батькові,
науковий ступінь, вчене звання)

Реєстрація № 11 24.01.2025
(дата)


(підпис відповідальної особи)

Результати
перевірки До захисту
(до захисту, на доопрацювання)

7.01.25 
(дата) (підпис)

Результат
захисту 4 (добре)
(оцінка)

06.02.2025 
(дата) (підпис)

Голова комісії: 
(підпис)

В.М. Франчук

Члени комісії: 
(підпис)

А.Ю. Горбовий


(підпис)

М.М. Філоненко


(підпис)

І.Д. Погореловська

БЛАНК ОЦІНЮВАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Об'єкти контролю	Діапазон оцінки в балах	Фактична кількість балів	Примітки	Дата, П.І.Б.: керівника, голови комісії членів комісії, секретаря, (підписи)
ОЦІНЮВАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ КОМІСІЄЮ ПО ЗАХИСТУ				
1. Зміст кваліфікаційної роботи (якість теоретичного матеріалу, його зв'язок з практичним, вміння представляти матеріал в логічній послідовності, наявність висновків та пропозицій)	від 0 до 30	24		06.02.2025 Франчук В.М. <i>В.М.</i> Горбовий А.Ю. <i>А.Ю.</i> Філоненко М.М. <i>М.М.</i> Погореловська І.Д. <i>І.Д.</i> Суховерхівка Л.П. <i>Л.П.</i>
2. Дотримання вимог щодо оформлення	від 0 до 10	8		
3. Вчасне виконання роботи на всіх етапах у встановлені кафедрою терміни	від 0 до 5	4		
4. Рецензент	від 0 до 5	4		
РАЗОМ БАЛІВ	від 0 до 50	40		

Оцінювання кваліфікаційної роботи науковим керівником - 40 балів.

ПЕРЕВЕДЕННЯ ОЦІНКИ ЗА 100-БАЛЬНОЮ ШКАЛОЮ ДО НАЦІОНАЛЬНОЇ ШКАЛИ ТА ШКАЛИ ЄКТС

Оцінка за 100-бальною шкалою	Оцінка за шкалою ЄКТС	Оцінка за національною шкалою	Дата	П.І.Б.: голови, членів комісії, секретаря, (підпис)
80	B	4 (добре)	06.02.2025	Франчук В.М. <i>В.М.</i> Горбовий А.Ю. <i>А.Ю.</i> Філоненко М.М. <i>М.М.</i> Погореловська І.Д. <i>І.Д.</i> Суховерхівка Л.П. <i>Л.П.</i>

ПРАВИЛА ПЕРЕВЕДЕННЯ ОЦІНОК ДО НАЦІОНАЛЬНОЇ ШКАЛИ ТА ШКАЛИ ЄКТС

Оцінка за 100-бальною шкалою	Оцінка за шкалою ЄКТС	Оцінка за національною шкалою
90-100	A	5 (відмінно)
80-89	B	4 (добре)
70-79	C	
60-69	D	
50-59	E	3 (задовільно)
35-49	FX	2 (незадовільно)

Факультет фінансів та цифрових технологій
Кафедра комп'ютерних та інформаційних технологій і систем
Галузь знань 12 «Інформаційні технології»
Спеціальність 122 «Комп'ютерні науки»
Освітня програма «Інформаційні управляючі системи і технології в економіці»

Затверджую

Завідувач кафедри  Філоненко М.М.

«23» грудня 2023

Завдання на кваліфікаційну роботу
здобувачу вищої освіти
Колеснікову Кирилу Віталійовичу
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи «Дослідження методів забезпечення приватності транзакцій та користувачів блокчейну»
Затверджена наказом ректора від «20» грудня 2023 № 1701
2. Строк здачі закінченої роботи «25» грудня 2024
3. Мета роботи: дослідження та формування пропозицій щодо використання методів забезпечення приватності транзакцій у прикладних завданнях
Об'єкт дослідження: платформи і технології блокчейн ресурсів.
Предмет дослідження: технології безпеки блок-чейн платформ та транзакцій
Орієнтовна база дослідження: блок-чейн мережі.
4. Вхідні дані до роботи: технічна інформація та описи технологій блокчейну, транзакцій, приватність транзакцій та користувачів.
5. Перелік графічного матеріалу :рисунки інтерфейсів системи, структурні схеми, алгоритми, лістинги програм.
6. Зміст кваліфікаційної роботи (перелік питань за кожним розділом)
 1. ЗАГАЛЬНИЙ ОГЛЯД ІСНУЮЧИХ ПРОБЛЕМ ТА МЕТОДІВ ЇХ
 - 1.1 Технологія Blockchain та забезпечення
 - 1.2 Аналіз методів та засобів захисту інформації
 - 1.3. Огляд існуючих рішень
 2. ДОСЛІДЖЕННЯ ТЕХНОЛОГІЇ BLOCKCHAIN
 - 2.1 Хешування у Blockchain і Алгоритми досягнення консенсусу
 - 2.2 Класифікація блокчейн мереж
 3. РОЗРОБКА ТА ТЕСТУВАННЯ ТЕХОЛОГІЙ
 - 3.2 Реалізація смарт-контракту та його захист
 - 3.3 Формування висновків і пропозицій

7. Календарний план виконання роботи

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	
		За планом	Фактично
1	2	3	4
1	Вибір теми, підбір літературних джерел, формування структури кваліфікаційної роботи	До 15.12.2023	15.12.2023
2	I розділ (розкрити сутність і стан дослідження наукової проблематики роботи, після чого навести її загальну характеристику; огляд літератури за темою і вибір конкретних напрямів досліджень)	30.05.2024	30.05.2024
3	II розділ (аналіз експериментальної частини про методики досліджень)	09.09.2024	09.09.2024
4	III розділ (проведені теоретичні і (або) експериментальні дослідження та проектні рішення)	22.11.2024	22.11.2024
5	Висновки	16.12.2024	16.12.2024
6	Доопрацювання роботи з урахуванням зауважень наукового керівника	26.12.2024	26.12.2024
7	Отримання відгуку керівника	25.12.2024	25.12.2024
8	Отримання зовнішньої рецензії	30.01.2025	30.01.2025

8. Дата видачі завдання «23» грудня 2023

9. Керівник кваліфікаційної роботи


(підпис)

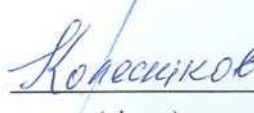
Редич О.В.

10. Керівник освітньо-професійної програми


(підпис)

Горбовий А.Ю.

11. Завдання до виконання отримав
здобувач вищої освіти


(підпис)

Колесніков К.

АНОТАЦІЯ

Актуальність теми. З розвитком засобів інформаційних комунікацій і зростанням ризиків, пов'язаних із пошкодженням або втратою даних, що зберігаються й передаються за їх допомогою, виникла потреба в інформаційній безпеці. Забезпечення інформаційної безпеки спрямоване на захист інформації та її інфраструктури від шкідливих впливів, які можуть бути випадковими або навмисними, внутрішніми або зовнішніми. Наслідком таких втручань може стати втрата важливих даних, їх несанкціонована зміна або використання третіми особами. Повноцінне та надійне забезпечення інформаційної безпеки можливе лише за умови використання комплексного й системного підходу. Система інформаційної безпеки має враховувати всі актуальні загрози та вразливості, а також передбачати потенційні ризики, що можуть виникнути в майбутньому. Тому важливо забезпечити безперервний і надійний контроль для захисту інформації.

Технологія Blockchain розпочинала свій розвиток як основа криптовалюти Bitcoin, але згодом стала перспективною технологією для забезпечення інформаційної безпеки. Блокчейн пропонує широкі можливості для підтримки високого рівня безпеки даних завдяки надійним механізмам шифрування, цілісності даних, стійкості мережі та масштабованості. У результаті перехід з традиційної системи інформаційної безпеки на систему на основі блокчейну може бути вигідним організаціям практично в будь-якій галузі.

Враховуючи численні переваги технології Blockchain та сучасні загрози інформаційних систем, розробка нових методів забезпечення інформаційної безпеки на основі блокчейну є актуальним і важливим завданням як з наукової, так і з практичної точки зору.

Об'єктом дослідження є методи та способи забезпечення інформаційної безпеки на основі технології блокчейн.

Предметом дослідження є способи захисту інформації від несанкціонованого доступу, а також уже реалізовані системи забезпечення інформаційної безпеки на основі технології блокчейн.

Мета роботи: підвищення інформаційної безпеки, за рахунок розробки модифікованого смарт-контракту на основі гібридної блокчейн-мережі.

Наукова новизна одержаних результатів роботи полягає у наступному.

23. Запропоновано спосіб забезпечення безпечної передачі даних на основі гібридної блокчейн-мережі, який являється покращеним методом уже існуючих та дозволяє забезпечити швидшу та надійнішу передачу даних.

24. Розроблено смарт-контракт на базі гібридної мережі Blockchain, який, на відміну від існуючих смарт-контрактів реалізованих на основі публічної мережі Blockchain дозволяє уникнути недоліків при передачі інформації та забезпечити надійний захист інформації.

Практична цінність отриманих в роботі результатів полягає в тому, що запропонований спосіб дає змогу забезпечити надійну передачу інформації, на основі технології блокчейн. Він може бути розширений та використовуватися в різноманітних сферах для забезпечення інформаційної безпеки.

Структура та обсяг роботи. Магістерська дисертація складається з вступу, трьох розділів та висновків.

У вступі подано загальну характеристику роботи, зроблено оцінку сучасного стану проблеми, обґрунтовано актуальність напрямку досліджень.

У першому розділі розглянуто основні поняття захисту інформації, проблеми інформаційної безпеки та існуючі методи забезпечення, а також проведено аналіз, який дає змогу визначити основні переваги та недоліки цих методів.

У другому розділі наведено теоретичні відомості щодо технології блокчейн, досліджено його архітектуру, види, проаналізовано основні переваги та недоліки.

У третьому розділі розробляється метод захисту інформації на основі гібридного ланцюга блокчейну, проведено тестування та порівняльний аналіз отриманих даних з результатами існуючих методів.

У висновках представлені результати проведеної роботи.

Ключові слова: захист інформації, інформаційна безпека, несанкціонований доступ, кібербезпека, технологія Blockchain, інформаційна безпека, смарт-контракти, децентралізована база даних.

ABSTRACT

Actuality of theme. With the development of information communications, and the ability to harm the information stored and transmitted through them, there was information security. Ensuring information security helps protect information and information infrastructure from negative impacts. Such influences may be accidental or intentional, internal or external, and may result in the loss of important information, unauthorized alteration, or use by third parties. Full and reliable information security can be ensured only with the use of an integrated and systematic approach. The information security system must be built taking into account all current threats and vulnerabilities, as well as taking into account those threats that may arise in the future, so it is important to ensure the maintenance of continuous and reliable monitoring.

Blockchain technology began as the basis of the cryptocurrency Bitcoin, but later became a promising technology for information security. Blockchain offers ample opportunities to maintain a high level of data security through robust encryption mechanisms, data integrity, network resilience and scalability. As a result, the transition from a traditional information security system to a blockchain-based system can be beneficial to organizations in virtually any industry.

Given all the advantages of Blockchain technology and the existing threats of modern information systems, the development of new methods of information security based on blockchain is an urgent and important task, both from a scientific and practical point of view.

The object of research are methods and ways to ensure information security based on blockchain technology.

The subject of the study are ways to protect information from unauthorized access, as well as already implemented information security systems based on blockchain technology.

Purpose: to increase information security by developing a modified smart contract based on a hybrid blockchain network.

The scientific novelty of the obtained results is as follows.

- A method for ensuring secure data transmission based on a hybrid blockchain network has been proposed, which is an improved method of existing ones and allows for faster and more reliable data transmission.
- A smart contract based on the Blockchain hybrid network has been developed, which, unlike existing smart contracts implemented on the basis of the public Blockchain network, avoids shortcomings in the transfer of information and provides reliable protection of information.

The practical value of the results obtained in this work is that the proposed method allows to ensure reliable transmission of information based on blockchain technology. It can be expanded and used in various fields to ensure information security.

Structure and scope of work. The master's dissertation consists of an introduction, three chapters and conclusions.

The introduction presents a general description of the work, assesses the current state of the problem, substantiates the relevance of research.

The first section discusses the basic concepts of information security, information security issues and existing security methods, as well as an analysis that identifies the main advantages and disadvantages of these methods.

The second section provides theoretical information on blockchain technology, explores its architecture, types, analyzes the main advantages and disadvantages.

The third section develops a method of information protection based on a hybrid blockchain chain, testing and comparative analysis of the obtained data with the results of existing methods.

The conclusions present the results of the work.

Keywords: information protection, information security, unauthorized access, cybersecurity, Blockchain technology, information security, smart contracts, decentralized database.

ЗМІСТ

Перелік скорочень, умовних позначень, термінів	5
ВСТУП.....	6
РОЗДІЛ 1. ЗАГАЛЬНИЙ ОГЛЯД ІСНУЮЧИХ ПРОБЛЕМ ТА МЕТОДІВ ВИРІШЕННЯ.	9
1.1. Актуальність проблеми та обґрунтування поставленої задачі.....	9
1.2. Основні поняття захисту інформації та забезпечення інформаційної безпеки	12
1.3. Основні загрози інформаційної безпеки.....	14
ВИСНОВКИ ДО РОЗДІЛУ 1	16
РОЗДІЛ 2. ДОСЛІЖЕННЯ ТЕХНОЛОГІЇ BLOCKCHAIN.....	18
2.1. Технологія Blockchain.....	18
2.2 Хешування у Blockchain	21
2.3 Алгоритми досягнення консенсусу	25
2.4 Класифікація блокчейн мереж	30
ВИСНОВКИ ДО РОЗДІЛУ 2	35
РОЗДІЛ 3. РОЗРОБКА ТА ТЕСТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ	36
3.1 Мова Solidity для розробки смарт-контракту.....	36
3.2 Створення блокчейн мережі.....	37
3.3 Реалізація смарт-контракту	43
ВИСНОВКИ ДО РОЗДІЛУ 3	51
ВИСНОВКИ.....	53
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	55

Перелік скорочень, умовних позначень, термінів

Blockchain	– вибудована за певними правилами безперервна послідовна ланцюжків блоків, децентралізована база даних, яка містить інформацію про всі транзакції, проведені учасниками системи.
DDoS	– Distributed denial-of-service attack. Хакерська атака на обчислювальну систему з метою довести її до відмови, тобто створення таких умов, за яких сумлінні користувачі системи не зможуть отримати доступ до системних ресурсів (серверів), що надаються, або цей доступ буде ускладнений.
DNS	– Domain Name System
EVM	– Ethereum virtual machine
POS	– Proof of Stake
POW	– Proof of Work
SHA	– Secure Hashing Algorithm
ІБ	– інформаційна безпека
Смарт-контракт	– комп'ютерний алгоритм, що зберігається мережі Blockchain, призначений для формування, контролю та надання інформації про володіння чимось.
КС	– комп'ютерна система.
ПЗ	– програмне забезпечення.

ВСТУП

Сучасне життя важко уявити без Інтернету та цифрових технологій. Вони глибоко інтегрувались у наше повсякдення, суттєво його полегшуючи. Сьогодні Інтернет — це одна з найвпливовіших складових у світі, що являє собою збірку різноманітних сервісів та ресурсів, з якими людина взаємодіє щодня.

З розвитком цифрових технологій ми отримуємо доступ до нових інструментів, які роблять звичні нам дії швидшими та комфортнішими. Однак, сучасний світ настільки залежний від комп'ютерних систем, що їх вихід із ладу або збій у роботі програмного забезпечення може призвести до серйозних наслідків.

Об'єктом дослідження є методи та способи забезпечення інформаційної безпеки на основі технології блокчейн.

Предметом дослідження є способи захисту інформації від несанкціонованого доступу, а також уже реалізовані системи забезпечення інформаційної безпеки на основі технології блокчейн.

Мета роботи: підвищення інформаційної безпеки, за рахунок розробки модифікованого смарт-контракту на основі гібридної блокчейн- мережі.

Практична цінність отриманих в роботі результатів полягає в тому, що запропонований спосіб дає змогу забезпечити надійну передачу інформації, на основі технології блокчейн. Він може бути розширений та використовуватися в різноманітних сферах для забезпечення інформаційної безпеки.

Забезпечення безпеки даних у комп'ютерних мережах є однією з ключових проблем у сфері інформаційних технологій. Виклики, пов'язані з новими загрозами та небезпеками, що виникають у процесі інформатизації, привертають увагу науковців сучасного світу.

«Інформаційна безпека (ІБ) - це збереження і захист інформації, а також її найважливіших елементів, в тому числі пристроїв, систем та обладнання, призначених для використання, заощадження та передачі інформації. Іншими

словами, це набір технологій, стандартів і методів управління, які необхідні для захисту інформаційної безпеки»[1].

«Лише системний та комплексний підхід до захисту може забезпечити інформаційну безпеку. У системі інформаційної безпеки потрібно враховувати всі актуальні та ймовірні загрози та вразливості. Для цього необхідний безперервний контроль в реальному часі. Контроль повинен проводитися постійно та охоплювати весь життєвий цикл інформації - від моменту, коли вона надходить в організацію, і до її знищення або втрати актуальності»[2].

Всі вище наведені вимоги може вирішити технологія блокчейн.

Блокчейн — це інноваційна технологія, популярність якої значно зросла разом із розвитком криптовалют. Ці цифрові активи є результатом складних обчислювальних процесів та представлені у вигляді електронних записів, що містять інформацію про баланс користувачів і їх фінансові операції. Зберігається криптовалюта у спеціальних цифрових гаманцях, а її створення та передача здійснюються за допомогою криптографічних методів, зокрема з використанням блокчейн-технології.

«Ідея створення блокчейну виникла 2008-го року та належить людині, відомій як Сатоші Накамото. Задум було втілено 2009-го року, а саме, було розроблено основну складову криптовалюти Bitcoin. Завдяки блокчейну, Bitcoin став першою криптовалютою»[8].

Блокчейн (blockchain або block chain) — це структура, що складається з послідовно з'єднаних блоків, кожен з яких містить певну інформацію. Блоки з'єднані за допомогою криптографії, що забезпечує їх незмінність і безпеку. Зазвичай копії ланцюга блоків зберігаються на багатьох комп'ютерах незалежно один від одного, що дозволяє створювати децентралізовані системи. Конфігурація блокчейну може змінюватись залежно від області застосування. Наприклад, блокчейн може бути загальнодоступним, з відкритим вихідним кодом, доступним для будь-якого користувача, або приватним, де необхідно чітко регулювати права доступу на читання та запис даних. Ці різні типи

блокчейн-мереж дозволяють адаптувати технологію до різноманітних потреб і вимог.

Технологію Blockchain можна уявити як велику книгу, що містить усю історію транзакцій з криптовалютою. Кожен власник цифрових активів має свою копію цього реєстру, яка повністю співпадає з іншими. Усі записи в цих копіях є автентичними та незмінними. Жодні фінансові установи, державні органи чи навіть власники криптовалюти не можуть внести корективи до цих даних.

Розвиток блокчейн-технології став ключовим фактором у створенні смарт-контрактів. Через п'ять років після її появи платформа Ethereum першою впровадила можливість практичного використання цих цифрових угод. Сьогодні блокчейн використовується не лише для транзакцій із цифровими активами, а й у багатьох інших сферах. Досвід криптовалюти Bitcoin продемонстрував, що поєднання криптографії та ефективних економічних механізмів може забезпечити надійний спосіб управління та зберігання даних.

В умовах зростання загроз інформаційній безпеці виникає нагальна потреба у впровадженні нових стратегій і технологій для захисту конфіденційної інформації.

Блокчейн та його унікальні властивості забезпечують безпечну передачу даних, що робить цю технологію ефективним інструментом для гарантування інформаційної безпеки. Завдяки незмінності блоків, криптографічному шифруванню та децентралізованій структурі, блокчейн може стати ідеальною системою захисту даних, втручання у яку майже неможливе. Ця технологія надає користувачам можливість самостійно зберігати та контролювати свої особисті дані, усуваючи необхідність довіряти їхню безпеку великим корпораціям. При цьому інформація залишається захищеною за допомогою криптографічних методів.

РОЗДІЛ 1. ЗАГАЛЬНИЙ ОГЛЯД ІСНУЮЧИХ ПРОБЛЕМ ТА МЕТОДІВ ВИРІШЕННЯ.

1.1 Актуальність проблеми та обґрунтування поставленої задачі

У сучасному світі стрімко розвиваються технології зберігання, обробки та передачі інформації. Використання інформаційних технологій потребує особливої уваги до аспектів інформаційної безпеки. Порушення безпеки, що призводять до втрати, недоступності або несанкціонованого використання даних, можуть спричинити серйозні наслідки для окремих громадян, соціальних груп, бізнесу та державних структур.

Ще 25-30 років тому завдання захисту інформації можна було ефективно вирішити переважно за допомогою організаційних заходів (наприклад, виконання режимних процедур, використання засобів охорони та сигналізації) і окремих програмно-апаратних засобів, таких як обмеження доступу та шифрування. Однак з появою персональних комп'ютерів, локальних та глобальних комп'ютерних мереж, супутникових каналів зв'язку та ефективних засобів технічної розвідки, які дозволяють отримувати конфіденційну інформацію, проблема захисту інформації стала набагато складнішою і актуальнішою. Ці технологічні досягнення значно збільшили можливості для несанкціонованого доступу до даних, що створило нові виклики для забезпечення безпеки інформації.

Збільшення кількості та складності загроз інформаційній безпеці в комп'ютерних системах не завжди супроводжується адекватним розвитком ефективних методів захисту даних та впровадженням безпечних інформаційних технологій. У більшості державних і комерційних структур, а також серед індивідуальних користувачів, заходи захисту часто обмежуються лише антивірусними програмами, які не завжди оновлюються вчасно, та використанням паролів для контролю доступу. У зв'язку з цим необхідно не

лише збільшувати кількість фахівців у сфері інформаційної безпеки, а й активно розробляти інноваційні методи та засоби захисту даних.

З розвитком сучасних технологій проблеми інформаційної безпеки стають дедалі актуальнішими. Фахівці у сфері захисту даних прогнозують, що загрози набудуть більш цілеспрямованого та закритого характеру, а впровадження передових технологій, таких як машинне навчання та нейромережі, значно ускладнить механізми кібератак. Зокрема, в найближчому майбутньому може зрости ризик витоку конфіденційної інформації, включаючи біометричні дані користувачів. Персональні відомості стають важливим інструментом для зловмисників, які вдосконалюють методи соціальної інженерії, що робить атаки ще більш переконливими. Через це інтерес кіберзлочинців до особистих даних постійно зростає. Крім того, існує ймовірність використання штучного інтелекту для створення детальних профілів жертв та розробки інформаційних фальсифікацій, які вже сьогодні викликають значне занепокоєння.

Протягом останнього року технології досягли значного прогресу. Пандемія COVID-19 справила суттєвий вплив на суспільство, змусивши переглянути підходи до інформатизації, організації робочих процесів та змін у звичках користувачів. Інформаційна безпека також зазнала трансформацій. Масовий перехід на віддалену роботу змусив компанії зосередитися на підтримці основних технологічних процесів, через що багато організацій пом'якшили заходи безпеки, щоб надати співробітникам доступ до необхідних ресурсів. Водночас значну загрозу почали становити пристрої для віддаленого доступу — домашні комп'ютери та смартфони, які часто не мають належного рівня захисту. Додатковим фактором ризику є брак часу та технічних знань у працівників для належного забезпечення безпеки цих пристроїв, що ще більше ускладнює ситуацію.

«Різке зростання використання хмарних додатків позначився на збільшенні числа DDoS-атак, які спрямовані на припинення доступу

користувачів до хмарних ресурсів і маскують більш небезпечні впливи на обчислювальні системи організацій. В опитуванні директорів , проведеному компанією Microsoft, 72% респондентів повідомили про те, що за останні 12 місяців в організації мали місце проблеми витоку інформації, і компанії акцентують увагу на управління ризиками інсайдерських атак у зв'язку з пандемією COVID-19»[5].

Цілі кіберзлочинців значно змінилися з часом. Раніше їхні атаки були зосереджені переважно на фінансових організаціях, оскільки ці структури давали можливість отримати максимальну фінансову вигоду. Проте, за даними Positive Technologies, у 2024 році частка таких атак знизилася до 8,1%, що пов'язано з підвищенням рівня захисту фінансових установ і зменшенням їх привабливості для зловмисників. Раніше промислові підприємства не були в центрі уваги кіберзлочинців, але ситуація змінилася. Кількість атак на цей сектор значно зросла. Якщо в IV кварталі 2023 року фахівці PT Expert Security Center зафіксували 31 атаку на промислові підприємства, то вже в I кварталі 2024 року число атак на цей сектор збільшилося до 71. Цей тренд свідчить про зміну стратегії кіберзлочинців, які тепер активно орієнтуються на промислові організації.

На сьогоднішній день трійка галузей, що найбільше постраждали від хакерських атак, виглядає доволі неочікувано: державні установи (13% усіх атак), промисловий та енергетичний сектори (12%), а також медичні заклади (9%). При цьому половина всіх атак була здійснена із застосуванням вірусів-вимагачів.

Оскільки у 2024 році загрози інформаційній безпеці продовжують зростати, а кіберзлочинці вдосконалюють свої методи проникнення в корпоративні мережі та компрометації даних, необхідно шукати нові підходи до захисту інформації. Комплексна стратегія забезпечення кібербезпеки дозволить суттєво зменшити кількість інцидентів як серед користувачів, так і в організаціях загалом.

Забезпечення інформаційної безпеки є безперервним процесом, що включає в себе розвиток систем захисту, постійний моніторинг, виявлення слабких місць, а також аналіз можливих каналів витоку інформації. У відповідь на зростаючі кіберзагрози з'являється новий підхід — технологія блокчейн. Вона має великий потенціал у різних областях, особливо в захисті пристроїв інтернету речей та децентралізованій ідентифікації.

1.2 Основні поняття захисту інформації та забезпечення інформаційної безпеки

У нинішню епоху цифрової трансформації питання збереження та захисту даних стало одним із найактуальніших. Здається, що кожен аспект нашого життя тепер піддається моніторингу через різноманітні інформаційні системи, а персональні дані опинилися під загрозою витоку та несанкціонованого використання, як ніколи раніше.

«Захист інформації — це діяльність, спрямована на запобігання її витоку, сукупність методів та інструментів, що забезпечують цілісність, конфіденційність, достовірність, автентичність і доступність даних у разі впливу на них загроз природного або штучного походження. Витоком вважається неконтрольоване розповсюдження інформації шляхом її розголошення, несанкціонованого доступу або отримання третіми сторонами»[2].

Розголошення — це процес доведення захищеної інформації до неконтрольованої кількості отримувачів (наприклад, публікація інформації на відкритому сайті в Інтернеті або в загальнодоступних виданнях). Несанкціонованим доступом вважається отримання захищеної інформації особами, які не мають на це права, з порушенням встановлених правил доступу. Несанкціонований вплив на захищену інформацію означає втручання в її зміст з порушенням правил, наприклад, впровадження шкідливого програмного коду в

інформаційні ресурси або навмисна заміна електронного документа. Ненавмисний вплив на захищену інформацію відбувається через помилки користувача, збої технічних або програмних засобів, природні явища або інші нецілеспрямовані впливи (наприклад, втрата документів через відмову накопичувача на жорсткому диску комп'ютера).

Основними цілями захисту інформації є:

- Запобігання витоку, розкрадання, втрати, спотворення і підробки інформації. Це включає захист будь-якої інформації, в тому числі відкритої.
- Запобігання загрозам безпеки особистості, суспільства і держави. Іншими словами, захист інформації є важливим елементом у забезпеченні інформаційної безпеки держави.
- Захист конституційних прав громадян на збереження особистої таємниці та конфіденційності персональних даних, що зберігаються в інформаційних системах.
- Збереження державної таємниці та конфіденційності документованої інформації відповідно до законодавства.

Поняття ефективності захисту інформації визначається як ступінь відповідності досягнутих результатів поставленим завданням. Об'єктом безпеки можуть бути не лише самі дані, а й носії інформації чи процеси її обробки, які потребують захисту. Якість інформації трактується як сукупність характеристик, що визначають її відповідність потребам користувачів та її функціональному призначенню. Одним із ключових показників якості інформації є її захищеність, що означає підтримку необхідного рівня безпеки під час зберігання, обробки та використання даних.

Основними характеристиками інформаційної безпеки, які підлягають захисту, є конфіденційність (запобігання несанкціонованому доступу), цілісність (гарантія незмінності даних) і доступність (можливість отримання інформації авторизованими користувачами).

Усі ці поняття формують основу системи забезпечення інформаційної безпеки.

1.3 Основні загрози інформаційної безпеки

«Під загрозою безпеки інформації в комп'ютерній системі (КС) розуміють подію або дію, яка може викликати зміну функціонування цієї системи, пов'язану з порушенням захищеності інформації, що обробляється в ній. Вразливість інформації - це можливість виникнення на будь-якому етапі життєвого циклу КС такого її стану, при якому створюються умови для реалізації загроз безпеці інформації»[5].

Атакою на комп'ютерну систему називають дію, яку здійснює зловмисник, використовуючи певну вразливість системи. Іншими словами, атака на КС — це реалізація потенційної загрози безпеці інформації, що зберігається або обробляється у системі.

Загрози інформаційної безпеки можна класифікувати на ті, що виникають незалежно від людської діяльності (природні загрози, спричинені стихійними явищами, які фізично впливають на інформацію), та загрози, викликані людською діяльністю (штучні загрози), які є значно небезпечнішими.

«Штучні загрози виходячи з їх мотивів поділяються на ненавмисні (випадкові) і навмисні.

До ненавмисних загроз відносяться:

- помилки в проектуванні КС;
- помилки в розробці програмних засобів;
- енергопостачання;
- помилки користувачів системи;
- вплив на апаратні засоби фізичних полів інших електронних пристроїв (при недотриманні умов їх електромагнітної сумісності) та ін.

Умисними загрозами вважають:

- несанкціоновані дії обслуговуючого персоналу системи;
- несанкціонований доступ до ресурсів КС з боку користувачів КС та сторонніх осіб, шкода від якого визначається отриманими порушником

повноваженнями»[7].

«Залежно від цілей навмисних загроз безпеці інформації, загрози можна поділити на три основні групи:

- Загроза порушення конфіденційності — це витік інформації з обмеженим доступом, яка зберігається в системі або передається між різними комп'ютерними системами.
- Загроза порушення цілісності — це навмисний вплив на інформацію, що зберігається в комп'ютерній системі, з метою її спотворення або фальсифікації.
- Загроза порушення доступу до інформації — це відмова в обслуговуванні, викликана навмисними діями порушника, внаслідок яких блокується доступ до певного ресурсу для інших користувачів.

Опосередкованою загрозою для безпеки інформації є загроза розкриття характеристик підсистеми захисту, що може створити умови для реалізації вищезгаданих прямих загроз. Наслідками таких загроз можуть бути: витік (неконтрольоване копіювання) даних, їх втрата (знищення), спотворення (фальсифікація) або блокування доступу до інформації»[7].

«Каналами витоку інформації є різні шляхи, через які інформація може потрапити до несанкціонованих осіб або систем. Безпосередні канали витоку — це канали, що пов'язані з фізичним доступом до елементів комп'ютерної системи, і для їх використання не потрібно змінювати самі елементи системи. До таких безпосередніх каналів витоку відносяться:

- розкрадання носіїв інформації;
- видалених файлів, помилково збережених тимчасових файлів);
- копіювання носіїв інформації;
- маскуванню під інших користувачів шляхом викрадення їх ідентифікуючої інформації (паролів, карт тощо) ;
- обхід засобів розмежування доступу до інформаційних ресурсів внаслідок недоліків у їх програмному забезпеченні
- зловмисна зміна програм для виконання несанкціонованого копіювання

інформації при її обробці»[7].

Окрім витоку інформації, також існують інші загрози для інформаційної безпеки, такі як несанкціоноване знищення або спотворення даних. До прикладів таких загроз належать: Зараження комп'ютерними вірусами, що може призвести до знищення або пошкодження файлів, програм або системи в цілому. Віруси можуть модифікувати, пошкоджувати або стерти критично важливу інформацію.

Оскільки найбільш небезпечні загрози інформаційної безпеки часто викликані намісними діями порушників, важливо застосовувати комплексний підхід до забезпечення захисту. Виходячи з цього, системно-концептуальний підхід є одним із найефективніших способів вирішення завдань захисту інформації.

ВИСНОВКИ ДО РОЗДІЛУ 1

Основну мету забезпечення інформаційної безпеки в контексті передачі даних у комп'ютерних мережах було детально проаналізовано. Розглянуто основні загрози інформаційній безпеці, їх класифікацію, канали витоку даних та проблеми, пов'язані з несанкціонованим доступом. У процесі дослідження було проведено аналіз існуючих методів захисту інформації, їх типів та класифікації. На основі виявлених недоліків сучасних методів захисту та враховуючи швидкий розвиток нових загроз, зроблено висновок, що система інформаційної безпеки потребує нових підходів і рішень.

У результаті аналізу існуючих методів, обґрунтовано необхідність впровадження технології Blockchain як одного з найефективніших інструментів для захисту даних. Технологія Blockchain має серйозний потенціал для посилення кібербезпеки, оскільки вона забезпечує незмінність, прозорість та децентралізоване зберігання інформації, що робить її надзвичайно стійкою до маніпуляцій і атак. Розглянуто можливі приклади застосування Blockchain для посилення кібербезпеки в майбутньому, включаючи захист персональних даних, безпечне зберігання та передачу інформації в різних сферах, таких як

фінансові установи, охорона здоров'я, урядові структури та інші важливі області, де захист інформації є критично важливим.

Встановлено, що впровадження блокчейну наразі ускладнюють кілька ключових проблем, серед яких обмежена масштабованість, висока енерговитратність, відсутність нормативно-правової бази та недостатня кількість кваліфікованих спеціалістів.

РОЗДІЛ 2. ДОСЛІЖЕННЯ ТЕХНОЛОГІЇ BLOCKCHAIN

2.1 Технологія Blockchain

Розробка криптовалюти Bitcoin програмістом Сатоші Накамото у 2009 році стала початком ери блокчейн-технологій. Це стало можливим завдяки тому, що Bitcoin забезпечив незалежність від централізованого контролю, що реалізується за допомогою блокчейну.

Блокчейн — це децентралізований реєстр транзакцій, який є складовою ширшої обчислювальної інфраструктури, що також включає функції зберігання даних, передачі інформації, управління файлами та архівування. Блокчейн — це безперервний і послідовний ланцюг блоків, створений за чіткими правилами, що містять інформацію. На рис. 2.1 представлено схематичну структуру блоків у блокчейн-мережі.

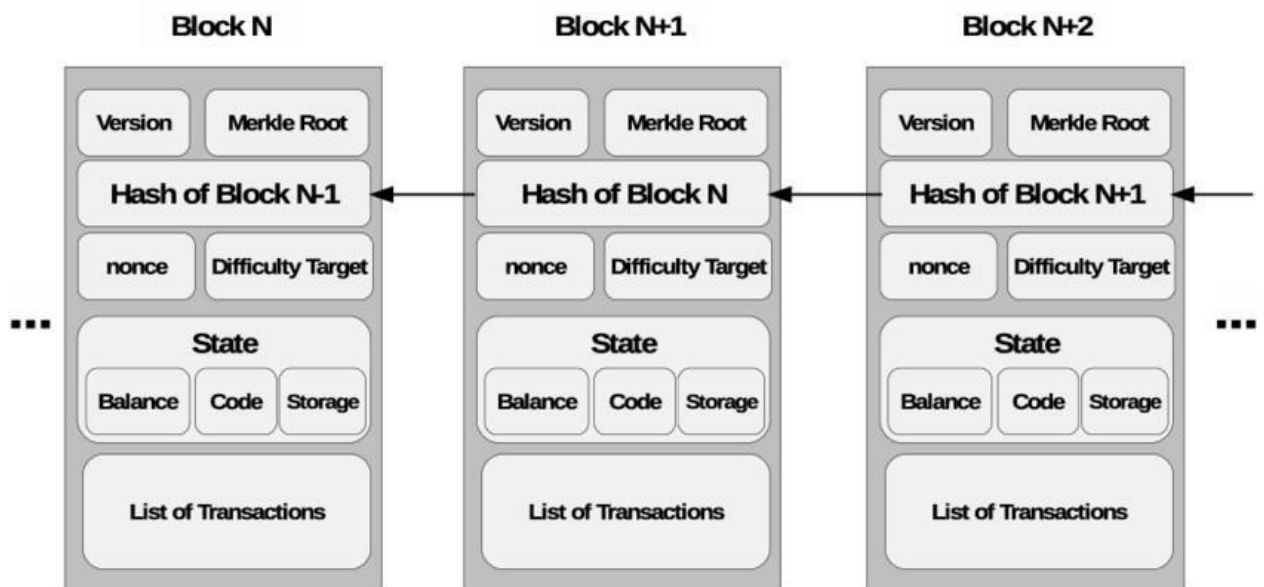


Рис 2.1 – Послідовність блоків у блокчейн мережі

«Блок являє собою контейнерну структуру даних, що об'єднує транзакції для їх подальшого включення у блокчейн. Він складається із заголовка, містить

метадані, а також довгий перелік здійснених транзакцій. Ідентифікація блоку можлива двома способами: за його унікальним хешем або шляхом визначення висоти у ланцюзі блоків. Особливе місце у блокчейні займає генезисний блок (*genesis block*), який є першим блоком у мережі. Його унікальність полягає в тому, що він єдиний у структурі блокчейну, який не містить посилання на попередній блок»[11].

Кожен блок у блокчейн-мережі складається з двох основних компонентів – заголовка (*head*) та тіла (*payload*). Заголовок (*head*) містить важливі метадані, що забезпечують стабільність та незмінність мережі. Тіло (*payload*) включає список усіх транзакцій, які підлягають збереженню в даному блоці та подальшому внесенню до блокчейн-мережі. На рис. 2.2 представлено структурну схему блоку блокчейну, що демонструє його основні елементи та взаємозв'язки.

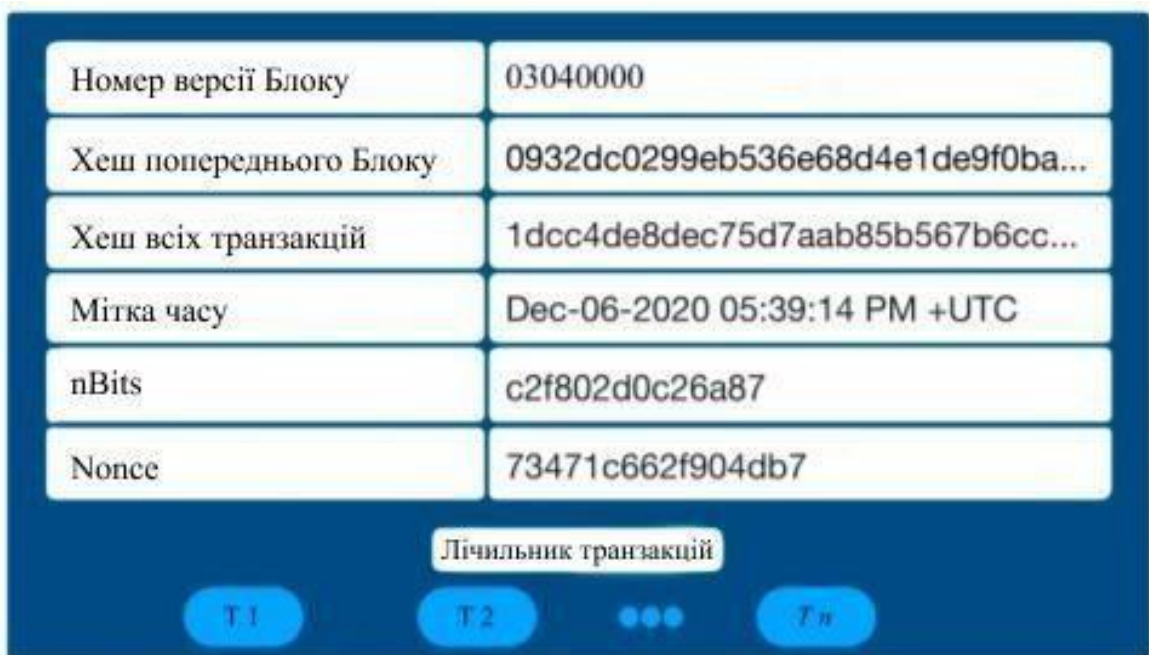


Рис 2.2 – Структура блоку

«У класичній блокчейн мережі Head містить такі поля:

- Номер версії блоку (*ver_block*) — вказує на версію блоку, що дозволяє ідентифікувати, які зміни були внесені в алгоритм блокчейну в певний

момент часу.

- Хеш попереднього блоку (`prev_block`) — хеш попереднього блоку в ланцюзі, що забезпечує незмінність і послідовність блоків.
- Хеш усіх транзакцій у поточному блоці (`mrkl_root`) — корінь Меркле-дерева.
- Тимчасова мітка (`timestamp`) — вказує час, коли блок був створений, що необхідно для визначення послідовності блоків у ланцюзі.
- «Bits» і «nonce» — параметри, що використовуються при майнінгу. Вони служать для знаходження відповідного хешу блоку під час процесу майнінгу.

Як раніше згадувалося, `payload` складається з лічильника транзакцій та списку всіх транзакцій, що входять до поточного блоку. Максимальна кількість транзакцій, що може містити блок, визначається розміром транзакції. Оскільки кожен блок має обмеження за розміром, кількість транзакцій у ньому може варіюватися залежно від їхнього обсягу. Для того, щоб перевірити справжність транзакції використовується механізм асиметричної криптографії»[12].

«Цифровий підпис — це криптографічний алгоритм, що застосовується користувачем до документа (повідомлення), який використовується для перевірки справжності та цілісності документа (повідомлення), а також для визначення авторства стосовно документа (повідомлення)»[11]. Криптографія з відкритими ключами (`public-key cryptography`) та хеш- функції надають математичні засоби, що дозволяють ефективно використовувати цифровий підпис. Використання шифрування та цифрових підписів є необхідним для функціонування блокчейн мережі. Хешування дає можливість кожному з учасників мережі визначити поточний стан блокчейна, а цифрові підписи забезпечують доказ того, що всі транзакції здійснено тільки справжніми власниками.

«Коли відбувається нова транзакція чи модифікація існуючої транзакції, більшість вузлів у мережі Blockchain здійснюють алгоритми для перевірки цього блоку. Якщо транзакція проходить перевірку, вона додається у вигляді

нового окремого блоку у базу даних (рис. 2.3)

Для того, щоб додати новий блок, необхідно отримати відповідь на криптографічну хеш-функцію»[12].

Після завершення транзакції для неї генерується унікальний хеш-код за допомогою криптографічного хешування, яке забезпечує зв'язок між поточною транзакцією та наступним блоком у ланцюзі.

Якщо більшість вузлів мережі відхиляє внесення нової або зміненої транзакції, вона не буде включена до блокчейну та не збережеться в базі даних, що гарантує незмінність і цілісність інформації.

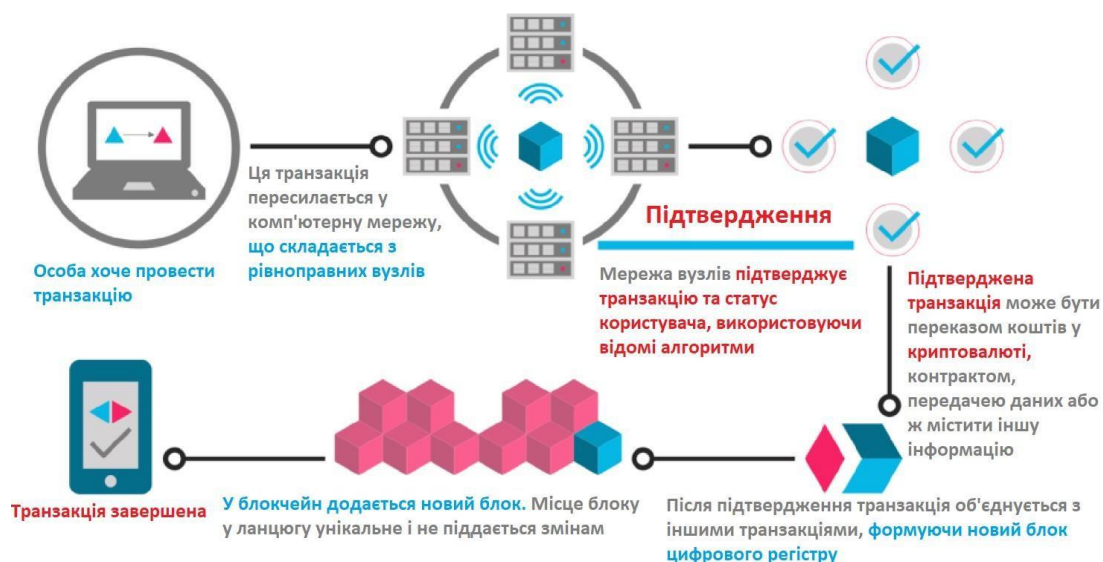


Рис 2.3 – Проведення транзакцій

2.2 Хешування у Blockchain

«Хеш-функції – це одні із найбільш широко використовуваних криптографічних алгоритмів у технології блокчейн. Це криптографічні (але не шифрувальні) алгоритми, призначені для захисту цілісності даних. Будь-яка частина даних може бути хешована, незалежно від її розміру або типу»[13]. У

традиційному хешуванню, незалежно від розміру, типу чи довжини вихідних даних, згенерований хеш завжди має фіксовану довжину. Хеш-функція є односторонньою та призначена для створення унікального цифрового відбитка вхідної інформації»[12].

Хеш-функція приймає вхідний рядок будь-якої довжини (числові значення, текст, мультимедійні файли) та перетворює його у вихідний рядок фіксованої довжини. Ця довжина може змінюватися залежно від обраного алгоритму хешування, наприклад, 32-бітний, 64-бітний, 128-бітний або 256-бітний. Отриманий результат називається хешем, і він є кінцевим значенням роботи хеш-алгоритму (рис. 2.4).

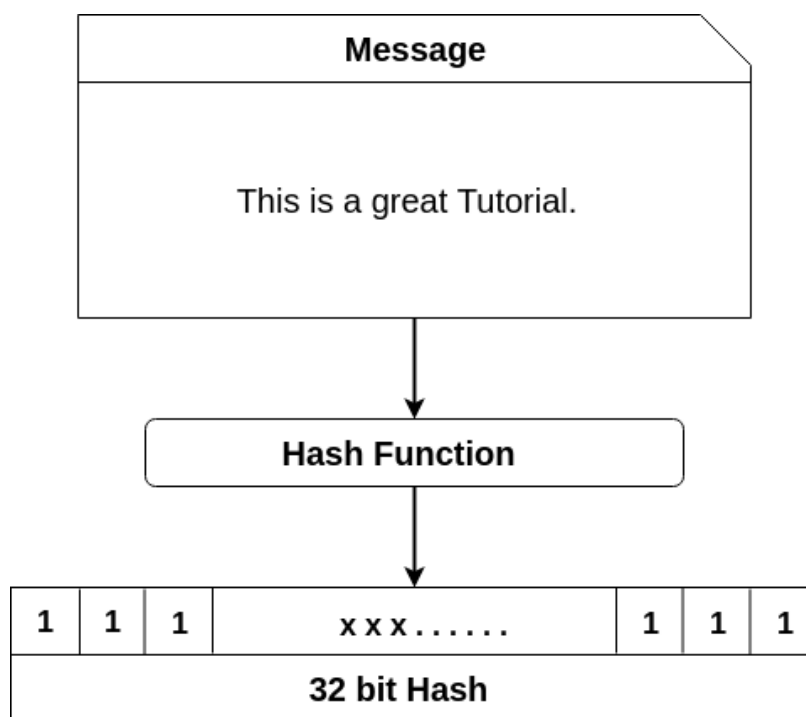


Рис 2.4 – Алгоритм хешування

«Хеш-функції виникли через необхідність зробити контент одноманітним та для використання унікальних ідентифікаторів. Хеш- функція забезпечує криптовалюті високий рівень безпеки, і хоча теоретично ніщо не захищене від зламів повною мірою, такий підхід дає максимально високий із доступних нині рівнів складності. При використанні хеш-функції можна досягати

високого рівня безпеки систем. У тому числі, з її допомогою можливе хешування паролів та шифрування інших даних. Щоб бути безпечною та використовуватися для криптовалют, хеш-функція повинна бути стійкою до колізій, а це означає, що неможливо знайти два вхідних рядка, які дадуть однаковий результат»[12].

Хеш-функції у блокчейні гарантують «незворотність» всього ланцюжка транзакцій. Кожен новий блок транзакцій посиляється на хеш попереднього блоку в реєстрі. Хеш самого блоку залежить від усіх транзакцій у блоці, але замість того, щоб послідовно передавати транзакції хеш-функції, вони збираються в одне хеш-значення за допомогою двійкового дерева з хешами (дерево Меркла). Таким чином, хеші використовуються як заміна покажчикам у звичайних структурах даних: пов'язаних списках та двійкових деревах»[13].

Будь-яке, навіть незначне, коригування вхідних даних спричиняє суттєві зміни у вихідному хеші, що є основою безпеки блокчейн-технології. Якщо в мережі відбудеться зміна хоча б одного запису, всі пов'язані хеші також зміняться, що зробить їх недійсними та застарілими. Завдяки децентралізованій природі блокчейну подібні маніпуляції майже неможливі.

Перший блок у блокчейні, відомий як генезисний блок (genesis block), містить первинні транзакції, які після перевірки формують унікальний хеш. Таким чином, кожен блок у мережі пов'язаний із попереднім, формуючи нерозривний ланцюг, що веде до блоку genesis. Така структура гарантує безпечне та незмінне додавання нових транзакцій.

Безпека алгоритму хешування зберігається доти, доки не буде виявлено колізію — випадок, коли різні вхідні дані дають однаковий хеш. Якщо алгоритм піддається колізіям, він вважається компрометованим і непридатним для використання. Так сталося з застарілими алгоритмами MD5 і SHA-1, які більше не вважаються безпечними через можливість підробки хешів.

Для генерації хешу з масиву інформації у криптовалютах застосовується алгоритм SHA-256. Цей алгоритм забезпечує надійне хешування даних, який

практично неможливо підібрати або змінити без коригування вихідної інформації (рис. 2.5).

Input	Output
Hello	185F8DB32271FE25F561A6FC938B2E264306EC304EDA518007D1764826381969
hello	2CF24DBA5FB0A30E26E83B2AC5B9E29E1B161E5C1FA7425E73043362938B9824
BANKEX Proof-of-Asset protocol	6C3C86FD24B044BC19FD1764EEB5D04E7EC8833199A62C85150B92B39D2C8CF5

Рис 2.5 – Приклад хешування SHA-256

«SHA-256 (Secure Hash Algorithm 256-bit) є односпрямованою функцією хешування, що генерує цифрові рядки фіксованої довжини — 256 біт (або 32 байти) — з вхідних даних розміром до 2,31 ексабайт (2^{64} біт). Це означає, що алгоритм може обробляти величезні обсяги інформації, перетворюючи їх у короткий, фіксований хеш. Хеш-функції сімейства SHA-2 побудовані на основі структури Меркла-Дамгарда.

Основні характеристики алгоритму:

- індикатор розміру блоку: 64 байти;
- максимально допустима довжина повідомлення: 32 байти;
- характеристика розміру дайджесту повідомлення: 32 байти;
- стандартний розмір слова: 4 байти;
- параметр довжини внутрішньої позиції: 32 байти;
- кількість ітерацій в одному циклі: 64;
- швидкість, досягнута за протоколом: приблизно 140 Мб/с.

Вхідне повідомлення розбивається на блоки, кожен блок – на 16 слів. Алгоритм пропускає кожен блок повідомлення через цикл із 64 ітераціями. На кожній ітерації 2 слова перетворюються, функцію перетворення задають інші

слова. Результати обробки кожного блоку складаються, сума є значенням хеш-функції. На рисунку 2.6 зображено графічне подання однієї ітерації обробки блоку даних»[14].

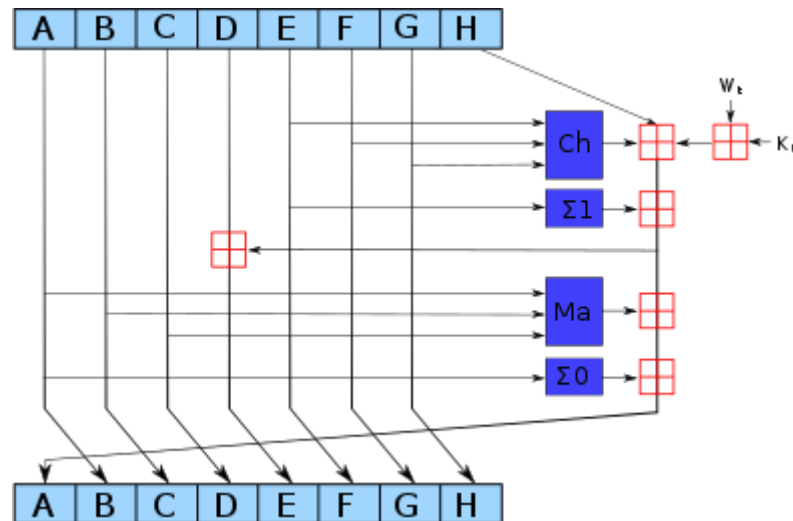


Рис. 2.6 - Графічне подання однієї ітерації обробки блоку даних у алгоритмі SHA-256

2.3 Алгоритми досягнення консенсусу

На відміну від традиційних баз даних, що знаходяться під управлінням адміністратора, публічний блокчейн функціонує як децентралізована однорангова мережа, до якої може долучитися будь-який користувач. Оскільки в такій розподіленій системі потенційно бере участь велика кількість вузлів, для її стабільного функціонування необхідний механізм консенсусу. Він забезпечує узгодженість стану мережі між усіма учасниками, що гарантує її коректну та безперебійну роботу.

«У блокчейні, що є децентралізованою системою, яка не має єдиного управляючого органу, задля досягнення консенсусу розроблені різні алгоритми. У блокчейн-мережі немає значення, довіряють учасники системи одне одному чи ні. У блокчейн-мережах учасники повинні домовитися

про певні принципи роботи, які стосуватимуться всіх, що забезпечить узгодженість та спільну роботу мережі. Це є прямою функцією механізму консенсусу, який дозволяє учасникам досягти угоди, не маючи централізованого управління. Алгоритмом консенсусу в блокчейні є набір певних математичних правил і функцій, які дозволяють досягти угоди між усіма учасниками та забезпечити працездатність мережі. Досягнення консенсусу стало основою математичної моделі побудови та розвитку блокчейн-технології»[16].

У децентралізованій мережі блокчейн можуть одночасно генеруватися кілька дійсних блоків, що призводить до утворення розгалужень в ланцюзі. Такі розгалуження називаються форками. Форк — це зміна протоколу або відхилення від попередньої версії блокчейна, яке може статися в результаті різних рішень або реалізації нових правил у мережі. Вибір однієї з гілок ланцюга блоків здійснюється за допомогою алгоритму консенсусу, що дозволяє учасникам мережі обрати єдину правдиву гілку. Якщо кілька гілок блоків мають однакову вірогідність, консенсусний алгоритм допомагає вирішити, яка з них буде продовжена. На рисунку 2.8 продемонстровано сценарій, у якому відбувається розгалуження даних на різних вузлах мережі блокчейн, що дозволяє побачити, як створюються форки і як вони впливають на подальше функціонування блокчейн-мережі.

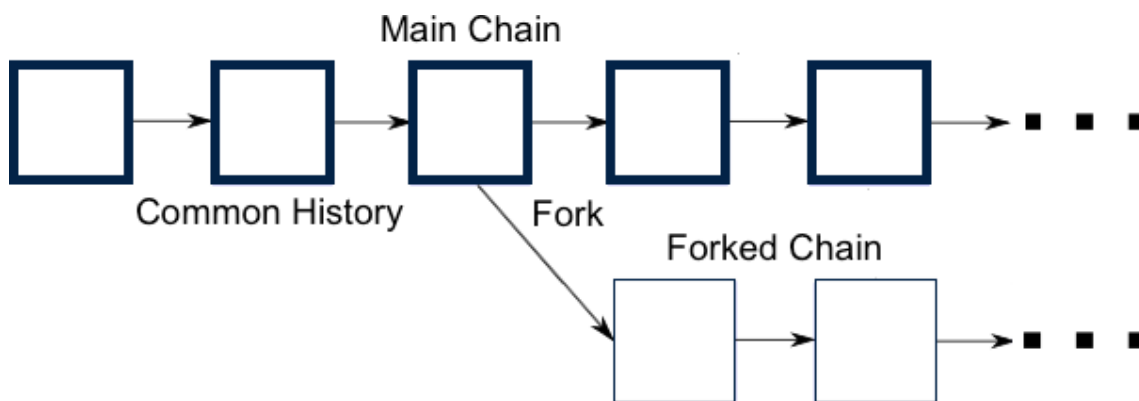


Рис. 2.8 – Розгалуження у блокчейні

Існують два основні види розгалужень в блокчейн-мережах:

м'які форки (soft forks) та жорсткі форки (hard forks).

Soft fork — це оновлення програмного забезпечення, яке має зворотну сумісність з більш старими версіями програмного забезпечення. Це означає, що учасники, які не оновили своє програмне забезпечення, все одно можуть брати участь у процесах валідації та перевірки транзакцій. Оновлення проводиться без порушення роботи старих версій, тобто, вони можуть продовжувати працювати, але з обмеженнями. Реалізувати soft fork набагато простіше, ніж hard fork, оскільки достатньо оновити лише клієнтське ПЗ, і всі учасники, незалежно від того, чи оновили вони ПЗ, продовжуватимуть розпізнавати нові блоки та підтримувати сумісність із мережею.

Однак є важливий момент: за старими версіями ПЗ учасники можуть перевіряти транзакції та бачити, що нові транзакції є дійсними, але коли новий блок додається, старі версії програмного забезпечення не зможуть визнати цей блок, і він буде відхилений мережею блокчейн. Це обмежує функціональність учасників, які не оновлюють своє програмне забезпечення, і може спричинити тимчасові проблеми з інтеграцією нових блоків у старі версії.

Hard fork — це оновлення програмного забезпечення, яке не має зворотної сумісності з попередніми версіями. Усі учасники повинні перейти на нове програмне забезпечення, щоб продовжити брати участь у створенні нових блоків. Ті, хто не оновиться, будуть відокремлені від мережі і більше не зможуть брати участь у перевірці нових транзакцій.

Такий розподіл призводить до постійного розходження ланцюгів блокчейну, що означає, що два або більше ланцюгів можуть існувати одночасно. Наприклад, після hard fork в мережі Bitcoin було створено нову мережу — Bitcoin Cash, що стало результатом розбіжностей у підходах до розвитку технології та зміни правил консенсусу.

«Proof of Work (PoW, доказ виконання роботи). Протокол PoW

пов'язаний з появою криптовалюти Bitcoin. PoW є одним із ресурсномістких протоколів консенсусу. У цьому протоколі кожному вузлу необхідно обчислювати значення хеша змінного заголовку блоку. У розподіленому середовищі всі вузли безперервно обчислюють значення хешу, використовуючи різні випадкові послідовності, які називаються "nonce". Загалом, рішенням є таким: $H(n \cup H(b)) < t$, де: H - це криптографічна хеш-функція (наприклад, SHA-256), n - значення "nonce", b - це поточний вміст блоку і t - час, що витрачається на знаходження нового блоку.

Наскільки швидко створюється блок залежить від того, наскільки складний ланцюг. Значення t не може бути дуже маленьким, так як це призводить до зайвих розгалужень у блокчейн мережі. Форки не тільки призводять до марнотратного витрачання ресурсів, але й мають негативні наслідки для безпеки, оскільки дозволяють реалізувати атаку подвійного витрачання.

Після отримання цільового значення хешу іншим учасникам необхідно взаємно підтвердити коректність хешу. Тоді транзакції, що зберігаються в новому блоці можуть бути перевірені у разі шахрайства. Таким чином з'являється новий блок у ланцюжку, який тісно пов'язаний із попереднім»[18].

Вузли, що займаються обчисленням хеш-функцій, називаються майнерами (miners), а сам процес перевірки та додавання нових блоків у мережу відповідно до алгоритму Proof of Work (PoW) отримав назву майнінг (mining). Оскільки майнінг вимагає значних обчислювальних потужностей, що супроводжується високими витратами електроенергії, були розроблені більш ефективні алгоритми консенсусу, які забезпечують енергозбереження та оптимізацію процесу перевірки транзакцій, такі як Proof of Stake (PoS) та його варіації.

Proof of Stake (PoS) (доказ володіння часткою) є більш ефективною альтернативою до Proof of Work (PoW). У протоколі PoS користувачі замість того, щоб обчислювати коректний "nonce", як це відбувається в PoW, повинні

довести право власності на певну кількість цифрових токенів. Це означає, що для того, щоб бути обраним для створення нового блоку, користувачі повинні заморожувати (стейкати) свої токени як доказ володіння. Цей підхід вважається менш енергоємним та ефективнішим, оскільки не потребує великих обчислювальних потужностей, як у випадку з PoW. Також у PoS вважається, що користувачі з більшою кількістю токенів мають більшу ймовірність бути обраними для додавання нових блоків, і, таким чином, менш схильні до атак на блокчейн-мережу, оскільки вони мають велику зацікавленість у підтримці безпеки мережі.

Для вирішення проблеми централізації в механізмах консенсусу, таких як Proof of Stake (PoS), були розроблені нові модифікації відбору вузла для створення наступного блоку. Однією з таких модифікацій є механізм, що використовується в Blackcoin. У Blackcoin застосовується механізм рандомізації для прогнозування наступного вузла, який отримає право створювати новий блок. Цей механізм використовує формулу, яка має на меті знаходження найменшого значення хешу в комбінації з розміром частки (тобто кількістю токенів, що належать користувачу). У роботі вузли з великою кількістю старих токенів мають велику ймовірність для створення нового блоку.

«Багато блокчейн мереж спочатку використовували PoW для досягнення консенсусу, але поступово почали переходити на PoS. Наприклад, Ethereum планується перевести до нової реалізації Casper, яка в кінцевому підсумку перетворює Ethereum на блокчейн PoS (також відомий як Ethereum2.0)»[18].

«Також можна відзначити протокол Nxt – гібрид двох протоколів PoW та PoS. Відмінною особливістю є те, що відбір вузла для генерації наступного блоку та право на підтвердження попереднього блоку обчислюється алгоритмом, який залежить від:

- хеша ідентифікатора попереднього блоку та відкритого ключа, який створив попередній блок;

— кількості токенів на рахунку користувача, підключеного до блокчейн мережі.

Основний принцип роботи протоколу DPoS виглядає наступним чином: реалізовано поділ голосуючих та валідуючих вузлів, тобто учасники мережі, які мають право голосу в системі (утримувачі токенів) не є валідаторами транзакцій. Таким чином, одну підмножину учасників обирає інша підмножина, яке в свою чергу буде формувати блоки.

Умови, в яких працює даний алгоритм консенсусу, кардинально відрізняються від умов, у яких працюють протоколи PoW та PoS. А саме, валідаторам необхідно розкрити свої особи та заявити про готовності безперебійно підтримувати роботу повноцінного вузла мережі, своєчасно виконувати верифікацію транзакцій та формувати нові блоки»[19].

Кожен механізм консенсусу має як переваги, так і недоліки. Ідеальний протокол повинен поєднувати високу ефективність, надійність та зручність у використанні. Однак навіть у найпопулярніших алгоритмах консенсусу дослідники періодично виявляють вразливості, що можуть негативно впливати на стабільність і продуктивність блокчейн-мережі.

2.4 Класифікація блокчейн мереж

Блокчейн-проекти постійно видозмінюються та вдосконалюються, що дозволяє їм адаптуватися до нових вимог і технологічних змін. Оскільки більшість блокчейн-систем (БС) базуються на відкритому вихідному коді, вони часто використовують розробки інших проектів, що сприяє швидкому розвитку цієї технології. Відкритий код дозволяє команді розробників брати за основу вже наявні рішення, покращувати їх та інтегрувати нові функціональності, що прискорює інноваційний процес. Кожна блокчейн-система має свої особливості та ознаки, які визначають її функціональність, безпеку та способи взаємодії з

іншими системами.

Найкоректнішим поділом блокчейн-мереж є поділ за рівнем відкритості, що дозволяє класифікувати їх на публічні (відкриті), приватні та гібридні мережі. Цей поділ відображає рівень доступу до системи та її функціональності, а також визначає, хто може брати участь у створенні нових блоків або верифікації транзакцій. Тупінь відкритості мережі часто залежить від кількох факторів, серед яких одним з найважливіших є доступність вихідного коду протоколу блокчейн-системи.

Для підключення нових користувачів до публічних блокчейн-мереж достатньо завантажити клієнтський додаток, сумісний із відповідною версією протоколу, та встановити зв'язок із вузлами мережі. Рівень участі визначається самим користувачем і залежить виключно від його фінансових та технічних можливостей. Важливо зазначити, що жоден учасник публічного блокчейну не має можливості відключити інших користувачів, оскільки всі вузли мережі є рівноправними.

Публічний блокчейн нерозривно пов'язаний із криптовалютами, які стали основою розвитку цієї технології. Саме цифрові активи стимулювали подальшу еволюцію блокчейну, дозволяючи долати обмеження ранніх реалізацій. Водночас фінансова складова завжди привертає увагу зловмисників, через що публічні блокчейн-мережі регулярно стають об'єктами різноманітних кібератак.

Приватний ключ створюється користувачем і забезпечує доступ до блокчейн-адреси, на якій можуть зберігатися цифрові активи. Володіючи приватним ключем, користувач отримує повний контроль над активами, що йому належать. Публічний ключ, у свою чергу, використовується як ідентифікатор гаманця або блоку, а також для верифікації цифрового підпису в інших блоках мережі. Таким чином, пара приватного та публічного ключа фактично формує блокчейн-гаманець.

Атаки на приватні ключі зазвичай не пов'язані безпосередньо з

блокчейн-мережею, але залишаються одним із головних ризиків для користувачів. Для мінімізації цих загроз розробляються децентралізовані платформи, які функціонують за іншими принципами, ніж традиційні криптобіржі. Вони не зберігають приватні ключі та персональні дані на своїх серверах, а лише виступають посередниками у процесі купівлі та продажу цифрових активів.

Додатковий рівень захисту цифрових активів забезпечує технологія мульти-підпису (multi-signature). Вона передбачає використання кількох приватних ключів для підтвердження транзакції, що значно підвищує безпеку гаманця. Оскільки для виконання операції необхідне схвалення від кількох незалежних користувачів, ризик несанкціонованого доступу суттєво знижується.

«Атака 51%. Концепція, при якій контроль над понад 50% ресурсами блокчейн-мережі, дозволяє створити свій форк, який замінить поточний реєстр блокчейну та стане актуальним. Окремим підвидом є атака дальньої дії, сенс якої полягає в тому, щоб почати збирати хибний форк задовго до актуального блоку. У деяких видах консенсусу це коштуватиме менше ресурсів. Залежно від алгоритму консенсусу, реалізація атаки відрізнятиметься. На даний момент виділяється дві основні групи алгоритмів: на основі доказу роботи (Proof-of-Work) та на основі підтвердження частки (Proof-of-Stake). У разі першого алгоритму (напр., Bitcoin та Ethereum) атакуючому потрібно заволодіти 51% обчислювальних ресурсів мережі. У випадку з Proof-of-Stake, для реалізації атаки 51% атакуючого потрібно заволодіти 51% основного активу мережі. Варто відзначити, що для розвинених і справді децентралізованих мереж, таких як Bitcoin і Ethereum, подібна атака залишається дуже дорогою – для її успішної реалізації потрібна величезна обчислювальна потужність, порівнянна з сотнями суперкомп'ютерів. Однак у менш масштабних мережах такі атаки є цілком реальними. Зокрема, влітку 2020 року було проведено дві атаки 51% на блокчейн Ethereum Classic – форк оригінального Ethereum, який стався після

зламування DAO у 2016 році»[20].

Атака Сивілли є видом загрози для однорангових (peer-to-peer) мереж, коли зловмисник створює велику кількість фальшивих вузлів, що перебувають під його контролем. Ці вузли підключаються до мережі та поступово оточують цільового користувача, отримуючи контроль над усіма його вхідними та вихідними транзакціями. У великих децентралізованих мережах реалізувати таку атаку вкрай складно, оскільки при підтвердженні транзакції учасники вибирають випадкові вузли для взаємодії. Однак у менш захищених або централізованих системах атака Сивілли може бути вкрай ефективною.

Блокчейн-мережі використовують криптографічні алгоритми, такі як SHA-256 (хеш-функція) та ECDSA (цифровий підпис), які на сьогоднішній день вважаються високостійкими до злому навіть із застосуванням сучасних обчислювальних потужностей. Однак розвиток квантових комп'ютерів може поставити під загрозу безпеку існуючих криптографічних механізмів. Завдяки своїм унікальним можливостям квантові обчислення здатні ефективно розшифровувати криптографічні алгоритми, що використовуються в блокчейнах, зокрема електронні підписи (ЕП), які є ключовим елементом перевірки транзакцій. З метою підготовки до постквантових загроз починається розробка нових криптографічних рішень. Наприклад, Quantum Resistant Ledger (QRL)

Приватний блокчейн — це закрита блокчейн-мережа, у якій доступ обмежується системою контролю, що визначає коло учасників. Управління такою мережею здійснюється однією або кількома організаціями, через що транзакції можуть вимагати участі посередників. На відміну від публічного блокчейну, в приватному блокчейні лише безпосередні учасники транзакції мають доступ до інформації про неї, тоді як сторонні користувачі залишаються без можливості перегляду даних. Основна перевага приватного блокчейну полягає у кращому контролі над інфраструктурою та гнучкості в керуванні мережею. Це дає організаціям можливість швидко оновлювати

функціональність, оптимізувати ІТ-процеси та посилювати інформаційну безпеку. Завдяки контрольованій та довірній архітектурі, приватні блокчейн-мережі менш вразливі до загроз, характерних для публічних мереж. Наприклад, атаки 51% або Сивілли стають майже неможливими, оскільки всі вузли належать перевіреним учасникам. Хоча зловмисник теоретично може спробувати отримати доступ до вузла приватної мережі, вартість такої атаки значно зростає. Це пояснюється необхідністю подолання корпоративного периметра безпеки, що включає захисні механізми інформаційної безпеки (ІБ), які використовують різні організації.

Контроль доступу є ключовою особливістю приватних блокчейн-мереж. На відміну від публічних блокчейнів, новий учасник не може приєднатися без дозволу оператора мережі. Це гарантує обмежений доступ до інформації, що відповідає класичним принципам інформаційної безпеки (ІБ), коли лише авторизовані користувачі мають право переглядати та використовувати дані.

Продуктивність блокчейн-мереж оцінюється за показником транзакцій за секунду (TPS). У публічних блокчейнах, що використовують Proof-of-Work (PoW), швидкість обробки обмежується десятками транзакцій за секунду. Водночас приватні блокчейни з ефективнішими алгоритмами консенсусу здатні досягати сотень або навіть тисяч транзакцій на секунду, що значно покращує їхню масштабованість.

Приватні блокчейн-мережі також забезпечують високий рівень конфіденційності, використовуючи гнучкі механізми керування доступом. Наприклад, контроль прав читання записів у реєстрі дозволяє обмежити доступ до транзакцій лише для певних учасників. Додатково, деякі платформи підтримують розширені методи захисту даних. Це дозволяє реалізовувати сценарії, коли навіть у межах однієї блокчейн-мережі певні транзакції можуть бути повністю приховані від сторонніх учасників, навіть у зашифрованому вигляді. Такий підхід значно підвищує рівень конфіденційності та безпеки фінансових та комерційних операцій.

ВИСНОВКИ ДО РОЗДІЛУ 2

У другому розділі роботи детально розглянуто та проаналізовано технологію Blockchain, її архітектуру та специфічні особливості. На основі отриманих результатів було виявлено ряд ключових переваг блокчейн-технології. Ці переваги роблять блокчейн-платформи потужним інструментом для забезпечення безпеки і довіри в багатьох сферах, включаючи фінанси, логістику, управління даними та інші.

Проведено порівняльний аналіз трьох типів протоколів досягнення консенсусу, а також публічних, приватних і гібридних блокчейн-мереж. Визначено основні переваги, проблеми та недоліки кожного виду блокчейну. У ході дослідження встановлено, що найдоцільнішим для забезпечення інформаційної безпеки є гібридний блокчейн. Цей тип мережі поєднує переваги як публічних, так і приватних блокчейнів, мінімізуючи їхні недоліки. Завдяки цьому гібридний блокчейн ефективно забезпечує всі аспекти інформаційної безпеки, що робить його надійним і безпечним засобом захисту даних.

Також у розділі розглянуто смарт-контракти, створені на основі публічних блокчейн-мереж, які мають певні недоліки в забезпеченні інформаційної безпеки. На підставі цього зроблено висновок, що смарт-контракти, розроблені на базі гібридної блокчейн-мережі, здатні усунути ці недоліки та стати надійним інструментом для безпечної передачі даних.

РОЗДІЛ 3. РОЗРОБКА ТА ТЕСТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ

3.1 Мова Solidity для розробки смарт-контракту

Solidity — це мова програмування високого рівня, розроблена для реалізації смарт-контрактів. Смарт-контракти є програмами, що виконуються в мережі Ethereum і дозволяють автоматизувати процес виконання угод, забезпечуючи контроль над їх виконанням. Мова була сильно впливала на мови програмування, такі як C++, Python і JavaScript, і була спеціально створена для роботи з Ethereum Virtual Machine (EVM). EVM можна описати як віртуальний комп'ютер, що працює в ланцюгу блоків, і який перетворює ідеї користувачів у код, що запускає програми безпосередньо в блокчейн-мережі.

Мова програмування була створена з урахуванням зручності для розробників, які вже мають досвід роботи з іншими сучасними мовами. Це дозволяє їм швидко опанувати синтаксис і принципи роботи нового інструменту.

Фактично, мова використовує компілятор, який перетворює зрозумілий для людини високорівневий код у низькорівневі інструкції, що можуть виконуватися мікропроцесором. Це робить розробку інтуїтивно зрозумілою, позбавляючи необхідності оперувати машинним кодом у вигляді нулів та одиниць.

Однією з найпопулярніших мов, що мають схожий синтаксис із Solidity, є JavaScript. Завдяки цьому програмісти можуть легко адаптуватися до нової мови та створювати блокчейн-програми без глибокого занурення у низькорівневі обчислення.

Смарт-контракти, написані на Solidity, є інструментом для автоматизації бізнесових та небізнесових процесів між різними учасниками. Вони виконують роль програмних угод, які забезпечують прозорість, безпеку та

виконання домовленостей без необхідності довіри між сторонами.

3.2 Створення блокчейн мережі

Сьогодні існує безліч реалізацій блокчейнів, проте найпопулярнішими залишаються Bitcoin та Ethereum. Якщо Bitcoin був створений як криптовалюта на основі технології блокчейн, то головною метою Ethereum є розробка універсальної платформи, яка дозволяє виконувати різноманітні завдання за допомогою смарт-контрактів. Крім того, Ethereum активно використовується для тестування та експериментів у сфері блокчейну.

Як і всі блокчейн-системи, Ethereum функціонує завдяки мережі вузлів, які працюють на спеціальному програмному забезпеченні. Основним компонентом цієї платформи є Ethereum Virtual Machine (EVM) – унікальне середовище виконання смарт-контрактів.

«EVM можна назвати середовищем виконання контрактів Ethereum, оскільки їх код виконується саме нею. Віртуальна машина Ethereum повністю ізольована від глобальної мережі, що не тільки робить значний внесок у безпеку, але й дозволяє створювати та тестувати смарт-контракти локально. Основне завдання EVM – виконання коду смарт-контракту. Однак EVM працює не з кодом, написаним мовою високого рівня, а з так званим EVM байт-кодом, який попередньо компілюється вихідний код контракту. Саме байт-код EVM зберігається у реєстрі. Цей байт-код є 16-терічним рядком, який є представленням кодів операцій та операндів.

Кожній операції відповідає вартість її виконання, так званий gas. Ця особливість EVM, вимога плати за кожну операцію, гарантує захист від DoS атак, що є найбільш популярним способом атаки в даний час. Кожен вузол у мережі Ethereum виконує одні й ті ж обчислення і зберігає одні й самі значення. Така надмірність якраз і дозволяє досягти консенсусу всередині мережі. Однак ця надмірність також є причиною, через яку виконання цих обчислень стає

сильно ресурсомістким. Для того щоб мотивувати вузли, виконувати такі обчислення в мережі Ethereum, існує поняття плати за обчислення: кожна операція, що виконується має свою конкретну вартість, що виражається у так званій кількості gas'a»[26].

Для запуску блокчейну Ethereum, комп'ютер повинен стати вузлом у мережі, що передбачає завантаження всього блокчейну та синхронізацію з основним ланцюгом Ethereum. Для підключення до мережі Ethereum використовуються спеціальні клієнти – програмне забезпечення, яке дозволяє взаємодіяти з блокчейном. Ці інструменти надають доступ до консолі (терміналу) – інтерактивного середовища JavaScript, що містить усі необхідні функції для управління блокчейном та виконання смарт-контрактів. Важливим етапом запуску блокчейну є отримання доступу до консолі, що забезпечує можливість взаємодії з мережею та розгортання смарт-контрактів. У рамках цієї роботи було обрано клієнт Go Ethereum (Geth)

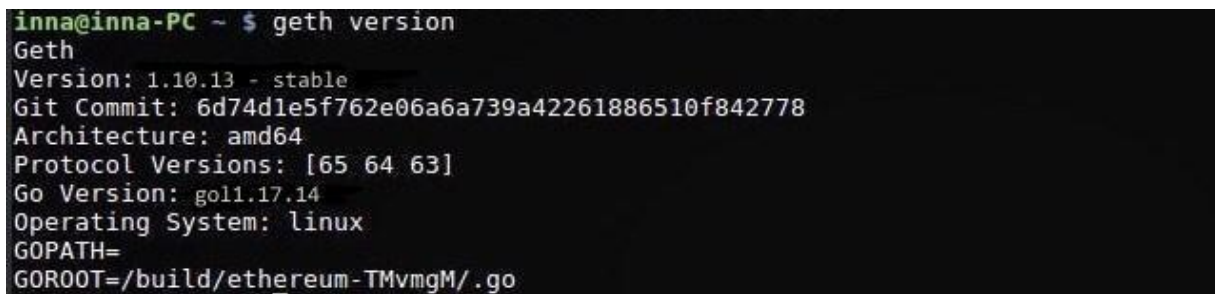
Go Ethereum - це одна з трьох основних реалізацій протоколу Ethereum (поряд з C++ та Python). Вона написана на мові Go ліцензована під GNU LGPL v3. Go Ethereum можна завантажити як окремий клієнт, що працює на різних операційних системах, або як бібліотеку для вбудовування у проекти Go, Android чи iOS. З Geth можна здійснювати транзакції, створювати смарт-контракти, децентралізовані програми, створювати приватні блокчейни та досліджувати історію блоків, а також багато іншого

Розробка відбувається на комп'ютері з Ubuntu Linux – операційною системою для персональних комп'ютерів. На комп'ютер встановлюється Go Ethereum за допомогою команд зображених на рисунку 3.1.

```
$ sudo apt-get install software-properties-common  
$ sudo apt-get install build-essential  
$ sudo add-apt-repository -y ppa:ethereum/ethereum  
$ sudo apt-get update  
$ sudo apt-get install ethereum
```

Рис. 3.1 – Установка Go Ethereum

Після написання вищеперерахованих команд було встановлено Geth 1.11.14 та Go 1.18.15. (рис. 3.2).



```
inna@inna-PC ~ $ geth version
Geth
Version: 1.10.13 - stable
Git Commit: 6d74d1e5f762e06a6a739a42261886510f842778
Architecture: amd64
Protocol Versions: [65 64 63]
Go Version: go1.17.14
Operating System: linux
GOPATH=
GOROOT=/build/ethereum-TMvmgM/.go
```

Рис. 3.2 – Версії встановлених утиліт

Для реалізації смарт-контракту необхідно створити приватний блокчейн, який слугуватиме для генерації хешу транзакцій. У майбутньому ці транзакції можуть бути перевірені через загальнодоступну блокчейн-мережу. Після встановлення та синхронізації Geth із мережею Ethereum можна запустити приватну тестову мережу. Як відомо, кількість блоків у блокчейні є необмеженою, проте існує перший блок, що започатковує весь ланцюжок. Він називається генезис-блоком (genesis block) і є відправною точкою для будь-якої блокчейн-мережі.

Щоб створити приватний блокчейн, необхідно спочатку згенерувати генезис-блок. Для цього потрібно виконати наступні кроки: створити файл Genesis, прописати до нього потрібні команди (атрибути) та використовувати його за допомогою Geth (рис.3.3).

```
{
  "config": {
    "chainId": 777,
    "homesteadBlock": 0,
    "eip155Block": 0,
    "eip158Block": 0
  },
  "difficulty": "0x400",
  "gasLimit": "0x8000000",
  "alloc": {}
}
```

Рис. 3.3 – Початкові установки для створення генезисного блоку

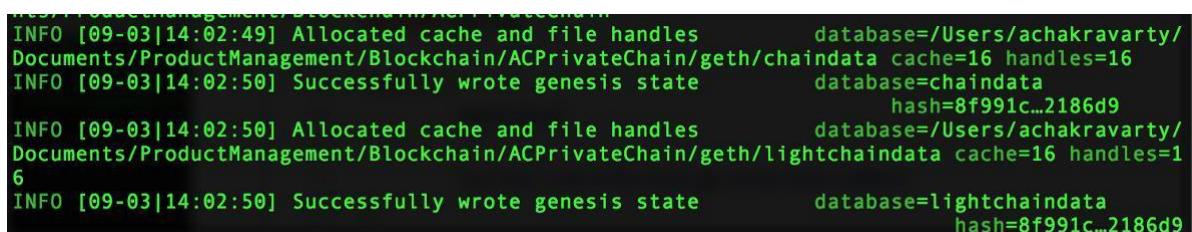
Нижче наведено основні параметри та їх значення: :

- config. визначає загальні параметри конфігурації блокчейну. Він не налаштовує сам генезис-блок, але визначає правила роботи мережі, включаючи підтримку оновлень протоколу Ethereum.
- chainId. Ідентифікатор використовуваного блокчейна (щось на зразок адреси), унікальний ідентифікатор блокчейну, що допомагає відрізнити одну мережу від іншої. Може бути встановлений у довільне значення, якщо воно не збігається з основними ланцюгами (наприклад, основний Ethereum має chainId = 1). У поточній конфігурації використовується ідентифікатор 777.
- HomesteadBlock. Вказує на перехід мережі до першої стабільної версії протоколу Ethereum – Homestead. Значення "0" означає, що мережа відразу запускається з підтримкою Homestead. Додатково можна підключити інші оновлення, такі як Byzantium, eip155B та eip158.
- difficulty. Визначає складність майнінгу блоків. Чим вище значення, тим більше обчислювальних ресурсів потрібно для створення нового блоку. Для тестової мережі встановлюється низьке значення, щоб уникнути довгого очікування під час генерації блоків.
- gasLimit. Вказує максимальну кількість газу, що може бути витрачена на виконання транзакцій у блоці. Газ є своєрідним «паливом» для

роботи мережі Ethereum, що використовується для оплати комісій за транзакції та виконання смарт-контрактів.

- alloc. Дозволяє створити криптовалютний гаманець у приватному блокчейні та наповнити його ефіром для подальшого тестування.

Після створення файлу `genesis.json`, для ініціалізації приватного блокчейну через Geth, необхідно ввести команду, що виглядає наступним чином: `“geth –identity –init/folder/genesis.json datadir/data_directory/ACPrivateChain”`. У цій команді параметр `--identity "MyPrivateChain"` визначає ім'я приватного блокчейну, яке можна вибрати довільно. Параметр `--datadir /path/to/data_directory/ACPrivateChain` вказує шлях до директорії, в якій будуть зберігатися дані приватного ланцюга. Команда `init /path/to/folder/genesis.json` визначає шлях до файлу `genesis.json`, що містить налаштування для запуску блокчейну. Виконуючи цю команду, Geth використовує файл `genesis.json` для створення першого блоку приватного блокчейну, який відповідатиме зазначеним налаштуванням. Після запуску даної команди Geth підключається до `genesis`-файлу і підтверджує успішне завершення дії (рис. 3.4).



```
INFO [09-03|14:02:49] Allocated cache and file handles      database=/Users/achakravarty/
Documents/ProductManagement/Blockchain/ACPrivateChain/geth/chaindata cache=16 handles=16
INFO [09-03|14:02:50] Successfully wrote genesis state       database=chaindata
                                         hash=8f991c...2186d9
INFO [09-03|14:02:50] Allocated cache and file handles      database=/Users/achakravarty/
Documents/ProductManagement/Blockchain/ACPrivateChain/geth/lightchaindata cache=16 handles=1
6
INFO [09-03|14:02:50] Successfully wrote genesis state       database=lightchaindata
                                         hash=8f991c...2186d9
```

Рис. 3.4 – Успішне підключення генезис блоку

Після створення генезис-блоку, можна запустити приватну блокчейн-мережу, що дозволить вузлам взаємодіяти та додавати нові блоки. Для цього в консолі Geth необхідно виконати команду: `“geth –datadir/data_directory/ACPrivateChain –networkid 7755”`. Ця команда вказує директорію, в якій зберігаються дані

блокчейну, включаючи генезис-блок та подальші блоки мережі. А також задає унікальний ідентифікатор мережі (network ID), який допомагає відокремити приватний блокчейн від інших Ethereum-мереж. Існує список ідентифікаторів існуючих мереж блокчейна Ethereum, які не можна використовувати при створенні власної мережі(рис. 3.5).

- **0:** Olympic, Ethereum public pre-release PoW testnet
- **1:** Frontier, Homestead, Metropolis, the Ethereum public PoW main network
- **1:** Classic, the (un)forked public Ethereum Classic PoW main network, chain ID 61
- **1:** Expanse, an alternative Ethereum implementation, chain ID 2
- **2:** Morden Classic, the public Ethereum Classic PoW testnet
- **3:** Ropsten, the public cross-client Ethereum PoW testnet
- **4:** Rinkeby, the public Geth-only PoA testnet
- **5:** Goerli, the public cross-client PoA testnet
- **6:** Kotti Classic, the public cross-client PoA testnet for Classic
- **8:** Ubiq, the public Gubi main network with flux difficulty chain ID 8
- **42:** Kovan, the public Parity-only PoA testnet
- **60:** GoChain, the GoChain networks mainnet
- **77:** Sokol, the public POA Network testnet
- **99:** Core, the public POA Network main network
- **100:** xDai, the public MakerDAO/POA Network main network
- **31337:** GoChain testnet, the GoChain networks public testnet
- **401697:** Tobalaba, the public Energy Web Foundation testnet
- **7762959:** Musicoin, the music blockchain
- **61717561:** Aquachain, ASIC resistant chain
- **[Other]:** Could indicate that your connected to a local development test network.

Рис. 3.5 - Ідентифікаторів існуючих мереж Ethereum

Після успішного виконання цієї команди, термінал Geth виведе повідомлення, яке підтверджує, що блокчейн був успішно ініціалізований і готовий до використання. Рисунок 3.6 демонструє вигляд виведеного терміналу після успішного запуску приватного блокчейну.

```

INFO [09-03|15:35:07] Starting peer-to-peer node           instance=Geth/v1.6.7-stable-ab5646c5/darwin-amd64/go1.9
INFO [09-03|15:35:07] Allocated cache and file handles       database=/Users/achakravarty/Documents/ProductManagement/Blockch
INFO [09-03|15:35:07] Initialised chain configuration         config="{ChainID: 15 Homestead: 0 DAO: <nil> DAOSupport: false E
P150: <nil> EIP155: 0 EIP158: 0 Metropolis: <nil> Engine: unknown}"
INFO [09-03|15:35:07] Disk storage enabled for ethash caches  dir=/Users/achakravarty/Documents/ProductManagement/Blockchain/A
PrivateChain/geth/ethash count=3
INFO [09-03|15:35:07] Disk storage enabled for ethash DAGs    dir=/Users/achakravarty/.ethash
count=2
INFO [09-03|15:35:07] Initialising Ethereum protocol         versions="[63 62]" network=6136
INFO [09-03|15:35:07] Loaded most recent local header         number=1493 hash=700fb2...0d9621 td=274732247
INFO [09-03|15:35:07] Loaded most recent local full block     number=1493 hash=700fb2...0d9621 td=274732247
INFO [09-03|15:35:07] Loaded most recent local fast block     number=1493 hash=700fb2...0d9621 td=274732247
WARN [09-03|15:35:07] Blockchain not empty, fast sync disabled
INFO [09-03|15:35:07] Starting P2P networking
INFO [09-03|15:35:07] Mapped network port                    proto=udp extport=30303 intport=30303 interface=NAT-PMP(192.168.
.1)
INFO [09-03|15:35:07] UDP listener up                       self=enode://eb9dc1560e03868ec7b4986db8878277125d1a119f2453c592e
715871498fe74a002e6ad6c9901699f0edd73e259d57c37568941146311fb81dbb0ff1e66ce4@24.207.134.58:30303
INFO [09-03|15:35:07] RLPx listener up                     self=enode://eb9dc1560e03868ec7b4986db8878277125d1a119f2453c592e
715871498fe74a002e6ad6c9901699f0edd73e259d57c37568941146311fb81dbb0ff1e66ce4@24.207.134.58:30303
INFO [09-03|15:35:07] IPC endpoint opened: /Users/achakravarty/Documents/ProductManagement/Blockchain/ACPrivateChain/geth.ipc
INFO [09-03|15:35:07] Mapped network port                    proto=tcp extport=30303 intport=30303 interface=NAT-PMP(192.168.
.1)

```

Рис. 3.6 – Успішний запуск блокчейн мережі

Однією з важливих особливостей Geth є те, що він не зберігає налаштування приватної блокчейн-мережі після завершення роботи. Це означає, що при кожному перезапуску потрібно заново ініціювати підключення до приватного ланцюжка за допомогою відповідних команд у консолі.

3.3 Реалізація смарт-контракту

Смарт-контракт, написаний мовою Solidity, за своєю структурою нагадує класи в об'єктно-орієнтованому програмуванні, оскільки містить набір змінних стану та функцій, що реалізують основну логіку. Смарт-контракти зберігають свої змінні стану у постійному сховищі, а кожна функція виконує певну програмну операцію, визначену розробником. У цьому контракті реалізовано простий гаманець, що дозволяє керувати балансом та фіксувати відтік коштів. Головні змінні включають `owner`, який зберігає адресу власника гаманця, та `outflow`, що являє собою асоціативний масив, де зберігається інформація про виведені кошти на кожну адресу. Окрім цього,

контракт неявно містить `balance`, значення якого відповідає загальній сумі отриманих контрактом коштів. Також договір неявно містить змінну `balance`, значення якої відповідає сумі переведених на договір грошових одиниць.

Функції смарт-контракту виступають як «вхідні точки», які дозволяють ініціювати виконання певного коду в контракті. Окрім стандартних функцій, що мають чітко визначену сигнатуру, смарт-контракт також містить конструктор, який викликається при створенні контракту, а також `fallback`-функцію з порожньою сигнатурою, яка активується в особливих випадках. Наприклад, `fallback`-функція спрацьовує, коли на контракт переводяться тільки грошові одиниці або якщо викликається функція, яка не існує в контракті. Для початку опису смарт-контракту необхідно установити початкові параметри (рис.3.7).

```
pragma solidity ^0.8.10;

contract Ownable {
    address public owner;

    mapping (address => uint) public outflow;

    constructor() public {
        owner = msg.sender;
    }
}
```

Рис. 3.7 – Початкові установки смарт-контракту

Рядок `pragma solidity 0.8.11` вказує на те, що для компіляції смарт-контракту потрібно використовувати версію компілятора 0.8.11 або новішу. Це важливо, оскільки саме ця версія містить виправлення помилок та нові функції, які покращують стабільність і функціональність мови Solidity.

Ім'я контракту задається після ключового слова `contract`. У тілі договору контракту задається після ключового слова `contract`, а у тілі контракту описуються всі дані, що зберігаються. Маніпуляції з цими даними

здійснюються через сеттери (функції для зміни значень) та гетери (функції для отримання значень). Важливо, що гетери працюють миттєво і безкоштовно, оскільки не змінюють стан контракту, а просто зчитують інформацію. Натомість, сеттери змінюють дані контракту, тому вони вимагають оплати за газ і виконуються не миттєво — їх виконання відбувається тільки після того, як транзакція буде включена в блокчейн.

Оскільки цей смарт-контракт створений для надійної та захищеної передачі даних, він розроблений таким чином, щоб виключити вразливості, характерні для більшості смарт-контрактів. Наприклад, у цьому контракті не використовуються функції низького рівня виклику, що може призвести до потенційних проблем із безпекою. Для відправки засобів використовуються функції `send` і `transfer`, оскільки вони гарантують надійну передачу коштів і не можуть бути повторно викликані, що підвищує рівень безпеки.

Для захисту від DDos-атак, у смарт-контракті великі цикли поділяються на кілька дрібніших частин, щоб кожен з цих циклів виконувався в окремих транзакціях, що дозволяє уникнути перевантаження мережі та покращити ефективність.

Щодо видимості функцій, всі функції в Solidity мають один із чотирьох специфікаторів видимості: `public`, `private`, `external` та `internal`. Якщо функція не має явно вказаного специфікатора видимості, за замовчуванням вона є `public`, що означає, що її можна викликати з будь-якого місця. Тому для забезпечення обмеженого доступу до певних функцій необхідно чітко вказувати відповідний специфікатор видимості.

Також в Ethereum існує стандарт токенів, і якщо смарт-контракт відповідає цьому стандарту, його можна вважати токеном або монетою, що дає змогу гаманцям і біржам працювати з таким контрактом.

Контракт повинен містити три поля, зображених на рисунку 3.8.

```
string public constant name = "Coin Token";//повна назва монети
string public constant symbol = "СТ";//скорочена назва монети
uint32 public constant decimals = 18;//кількість знаків після коми
uint public totalSupply = 0;//початковий баланс
```

Рис. 3.8 – обов’язкові поля смарт-контракту

У смарт-контрактах Solidity поля часто визначають такі важливі аспекти, як назва монети, кількість знаків після коми (найпоширеніше значення — 18), а також початковий баланс у гаманці. Ці параметри є доволі простими і складають основні вимоги для створення токена.

Ще однією важливою частиною є поле `mapping`. У Solidity `mapping` функціонує як хеш-таблиця, яка зберігає дані у вигляді пар ключ-значення. Ключем може бути будь-який з вбудованих типів даних, проте посилання на інші типи даних не дозволяються. Значення у `mapping` може бути будь-якого типу, що дозволяє зберігати різноманітні дані, включаючи інші структури або навіть функції. `Mapping` в Solidity використовується для зв'язування унікальних адрес Ethereum з відповідними значеннями. Це є типом даних, що дозволяє створювати асоціативні масиви, де кожен ключ (в даному випадку — адреса Ethereum) відповідає певному значенню (наприклад, баланс гаманця або інша інформація). (Рисунок 3.9)

```
mapping (address => mapping (address => uint)) allowed;//поле дозволу
```

Рис. 3.9 – Поле `mapping`

Смарт-контракт містить ряд основних функцій, зображених на рисунку 3.10.

```

function totalSupply() constant returns (uint256 totalSupply)
function balanceOf(address _owner) constant returns (uint256 balance)
function transfer(address _to, uint256 _value) returns (bool success)
function transferFrom(address _from, address _to, uint256 _value) returns (bool success)
function approve(address _spender, uint256 _value) returns (bool success)
function allowance(address _owner, address _spender) constant returns (uint256 remaining)
function totalSupply() constant returns (uint256 totalSupply)
function balanceOf(address _owner) constant returns (uint256 balance)
function transfer(address _to, uint256 _value) returns (bool success)
function transferFrom(address _from, address _to, uint256 _value) returns (bool success)
function approve(address _spender, uint256 _value) returns (bool success)
function allowance(address _owner, address _spender) constant returns (uint256 remaining)

```

Рис. 3.10 – Основі функції смарт-контракту

Функція `totalSupply` – повертає сумарну кількість випущених монет.

Цю функцію може викликати будь-хто.

Функція `balanceOf` – повертає кількість монет, що належать конкретному `owner`. Ця функція може бути викликана будь-ким, і вона дозволяє гаманцям перевіряти баланс монет, що зберігаються за певною адресою. Функція `transfer` – використовується для передачі монет на адресу `_to`. Коли користувач хоче перемістити свої монети на іншу адресу, саме ця функція буде викликана.

Функція `transferFrom` – дозволяє передавати `_value` монет від однієї адреси `_from` до іншої адреси `_to`. Однак користувач `_from` повинен попередньо дати дозвіл на таку операцію, щоб запобігти несанкціонованим транзакціям.

Функція `approve` – дозволяє користувачеві `_owner` надати дозвіл іншому користувачеві `_spender` на зняття монет з його рахунку, але не більше ніж `_value` монет.

Функція `allowance` – повертає кількість монет, що дозволив знімати `_owner` користувачу `_spender`.

Solidity визначає події за допомогою ключового слова `event`. Після виклику подій їх аргументи розміщуються в ланцюжок блоків. Щоб спочатку використати події, їх потрібно спочатку оголосити. У даному контракті оголошено два типи подій (рис. 3.11).

```
event Transfer(address indexed _from, address indexed _to, uint _value); //подія для функцій transfer, transferFrom
event Approval(address indexed _owner, address indexed _spender, uint _value); //подія для функції approve
```

Рис. 3.11 – Оголошення подій

Тестування смарт-контракту здійснювалось у середовищі Remix, яке є потужним інструментом для розробки та тестування смарт-контрактів. Remix IDE — це програма з відкритим вихідним кодом, яка працює як для локальних, так і для веб-додатків. Вона сприяє швидкому циклу розробки завдяки інтегрованим плагінам та інтуїтивно зрозумілому графічному інтерфейсу. Remix підтримує весь процес розробки смарт-контрактів: від написання та тестування до налагодження та розгортання. Remix IDE є частиною Remix Project, що є платформою для розробки, яка використовує архітектуру плагінів для інтеграції різних інструментів. Вона дозволяє писати контракти Solidity прямо в браузері без необхідності в додатковому програмному забезпеченні. Remix містить модулі для тестування, налагодження та розгортання смарт-контрактів, що робить цей інструмент надзвичайно зручним для розробників блокчейн-рішень.

Після перенесення смарт-контракту у Remix, його можна скомпілювати та провести тестування функцій (рис. 3.12).

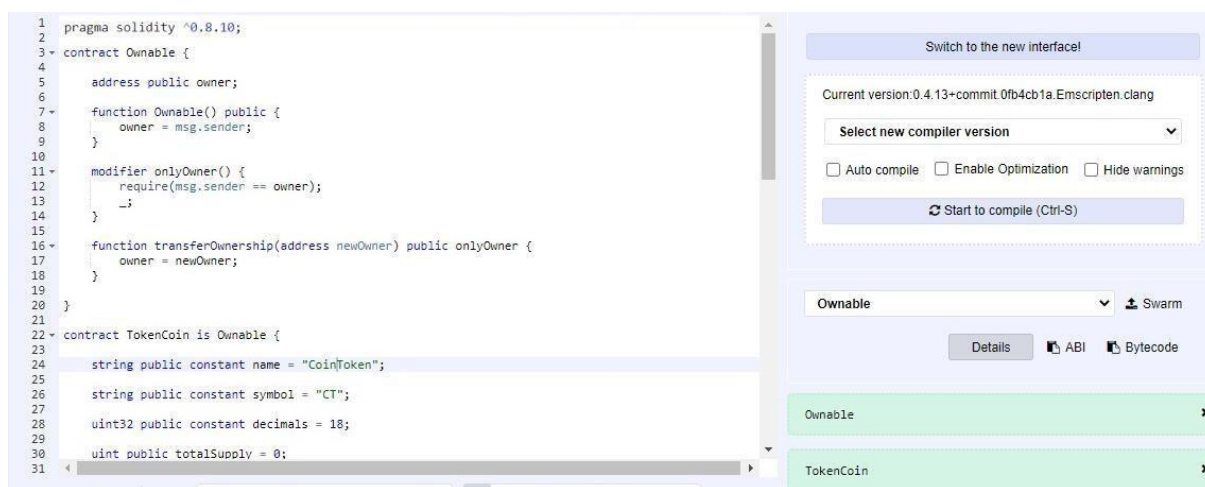


Рис. 3.12 – Смарт-контракт у середовищі Remix

За допомогою інтерфейсу взаємодії з функціями можна протестувати їх, та переказати кошти з одного гаманця на інший (рис.3.13).

✓ [vm] from:0xca3...a733c to:TokenCoin.transfer(address,uint256) 0x35e...450cf value:0 wei data:0xa90...00064 logs:0 hash:0xc8d...7d9a4

status	0x1 Transaction mined and execution succeed
transaction hash	0xc8d3e88599ed37980f78e3825148772756cadbe61e1d6110e845b3be7d9a4
from	0xca35b7d915458ef540ade6068dfe2f44e8fa733c
to	TokenCoin.transfer(address,uint256) 0x35ef07393b57464e93deb59175ff72e6499450cf
gas	3000000 gas
transaction cost	23661 gas
execution cost	789 gas
hash	0xc8d3e88599ed37980f78e3825148772756cadbe61e1d6110e845b3be7d9a4
input	0xa90...00064
decoded input	{ "address _to": "0xEA15Adb66DC92a48bCcC88F32Fd25E2e86a2A770", "uint256 _value": "100" }
decoded output	{ "0": "bool: success false" }
logs	[]
value	0 wei

Рис. 3.13 – Тестування функції переказу коштів

Функція смарт-контракту може не виконатися успішно, якщо на рахунку недостатньо коштів. У такому випадку, для забезпечення успішної операції, створюються монети, які потім додаються на рахунок, після чого можна перевірити загальний баланс гаманця.

Отримавши монети на рахунку, можна перевірити роботу функції переказу коштів на інший рахунок, для цього викликається функція transfer. Після того як тестування функцій смарт-контракту завершено, його можна додати до створеної блокчейн-мережі для використання реальних переказів.

У випадку з Ethereum Virtual Machine (EVM), виклик функції та її параметрів відбувається наступним чином:

1. Спочатку потрібно отримати хеш-сигнатуру функції за допомогою **алгоритму Кессак256.

Опублікувавши смарт-контракт, можна провести транзакцію та перевірити працездатність функцій (рис. 3.16).

```
transaction                                fullhash=0x429eb0bbbc50
cca55c8446adca8bdd5ed4c7df6fdc32930f7e14ce7ccb51ea51
recipient=0x1C766C3CD0114771C0e9E45e54F4924B3F81f0d8
```

Рис. 3.16 – Проведення транзакції у мережі

Смарт-контракт також опубліковано у публічній мережі, та проведено тестування.

ВИСНОВКИ ДО РОЗДІЛУ 3

У третьому розділі роботи розглянута мова програмування Solidity та зроблено висновок про те, що вона являється найкращою для написання смарт-контрактів.

На основі мови Solidity було створено смарт-контракт, що має на меті виконання конкретних функцій у блокчейн-мережі. Як і будь-які програми, смарт-контракти можуть містити помилки, які потенційно можуть бути використані для викрадення коштів. Тому було проведено детальний аналіз коду, щоб виявити і виправити можливі вразливості, що можуть загрожувати безпеці.

Після того, як смарт-контракт був створений, він був опублікований у приватній блокчейн-мережі. Після публікації було проведено серію тестувань, які виявили значні відмінності між контрактом, опублікованим у приватній мережі, і контрактом, опублікованим у гібридній мережі. Зокрема, смарт-контракт, що працює в гібридній мережі, проводить транзакції значно швидше, з меншою середньою комісією, а також з меншим часом проведення угоди. Крім того, робота ланцюга в гібридній мережі потребує менше енергоресурсів для забезпечення своєї дієздатності. Це робить гібридну

мережу більш ефективною з точки зору швидкості транзакцій, економії коштів і енергоспоживання, порівняно з приватною мережею. Ці результати підтверджують переваги використання гібридних блокчейн-мереж для проектів, які потребують високої продуктивності та зниження витрат на проведення операцій.

ВИСНОВКИ

У даній магістерській роботі було проаналізовано основну мету забезпечення інформаційної безпеки, методи захисту даних від несанкціонованого доступу, а також існуючі системи, що забезпечують інформаційну безпеку на основі блокчейн-технології. На основі проведених досліджень було запропоновано метод підвищення інформаційної безпеки шляхом розробки модифікованого смарт-контракту, що функціонує в межах гібридної блокчейн-мережі.

У першому розділі роботи було проведено дослідження основних загроз інформаційної безпеки, їх класифікацію, вивчено канали витоку інформації та проблеми, пов'язані з несанкціонованим доступом. Крім того, було проаналізовано існуючі методи захисту інформації, їх види та класифікацію. З огляду на недоліки чинних методів і швидкий розвиток нових загроз, зроблено висновок про необхідність впровадження нових підходів і рішень для забезпечення інформаційної безпеки. Як основу для розробки нового методу захисту було обрано технологію Blockchain.

У другому розділі роботи детально розглянуто та проаналізовано технологію Blockchain, зокрема її ключові переваги, такі як незмінність, децентралізація, прозорість та конфіденційність. На основі отриманих результатів зроблено висновок про те, що ці властивості технології є важливими для забезпечення високого рівня безпеки та довіри при передачі даних.

Проведено порівняльний аналіз видів блокчейн-мереж, у якому визначено їхні основні переваги, проблеми та недоліки. Зокрема, гібридний blockchain був визначений як найдоцільніший тип мережі для забезпечення інформаційної безпеки. У цьому типі блокчейн-мережі поєднуються переваги обох видів технології, що дозволяє обмежити їх недоліки та забезпечити усі аспекти інформаційної безпеки. Також розглянуто смарт-контракти, які працюють на різних типах блокчейн-мереж. Зроблено висновок, що смарт-контракти на основі

гібридної блокчейн-мережі здатні забезпечити відсутність вразливостей, що притаманні іншим типам блокчейн-систем, і є відмінним методом для безпечної передачі інформації.

Аналізуючи результати, отримані у першому та другому розділах роботи, у практичній частині було розроблено смарт-контракт мовою Solidity, використовуючи гібридну блокчейн-мережу. Після публікації контракту було проведено серію тестувань, які показали, що смарт-контракт, реалізований у такій мережі, забезпечує швидші та надійніші транзакції, а також є енергоефективним. Смарт-контракт гарантує безпечну передачу інформації, і будь-які спроби несанкціонованого доступу миттєво фіксуються і стають видимими для всіх учасників мережі. Завдяки своїй конструкції, цей контракт може бути розширеним і застосованим в компаніях для забезпечення надійної передачі даних. Організації зможуть скористатися перевагами блокчейн-технології, такими як висока продуктивність, повний контроль і прозорість, підтверджуючи транзакції в ланцюгу блоків, що сприяє покращенню перевірки та збереження даних. Це дозволяє значно підвищити рівень безпеки та ефективності в обробці даних у корпоративному середовищі.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Харченко В. С. Інформаційна безпека. Глосарій. — К.: КНТ, 2005.
(Last accessed: 27.12.2024).
2. Сідак В. С., Артемов В. Ю. Забезпечення інформаційної безпеки в країнах НАТО та ЄС: Навчальний посібник. — К.: КНТ, 2007. (Last accessed: 10.12.2024).
3. Chris Dannen. Introducing Ethereum and Solidity. — Brooklyn, New York, USA: Apress, 2017. — 197 с. (Last accessed: 11.12.2024).
4. Information Security Problems and Solutions. URL <https://www.deltanet.com/compliance/information-security/faqs/information-security-problems-and-solutions-deltanet>. (Last accessed: 25.12.2024).
5. Ярочкін В.І. Інформаційна безпека 2003. – 640 с.
(Last accessed: 19.12.2024).
6. Гатчін Ю. А., Сухостат В. В. Теорія інформаційної безпеки
(Last accessed: 25.12.2024).
7. Технологія блокчейн. URL: <https://cryptor.net/bitkoin-dlya-chaynikov/osnovnye-problemy-povsednevnogo-vnedreniya-blokcheyn-tehnologii>.
(Last accessed: 27.12.2024).
8. Основні проблеми запровадження технологій блокчейн URL: <https://cryptor.net/bitkoin-dlya-chaynikov/osnovnye-problemy-povsednevnogo-vnedreniya-blokcheyn-tehnologii>. (Last accessed: 20.12.2024).
9. Blockchain Guide Documentation URL: <https://readthedocs.org/projects/blockchain-guide/downloads/pdf/latest/>.
10. Лоран Лелу. Блокчейн від А до Я. Все про технології десятиріччя. — 2018. — 256 с. (Last accessed: 02.12.2024).
11. What Is a Block in the Blockchain? URL: <https://medium.com/datadriveninvestor/what-is-a-block-in-the-blockchain-c7a420270373>. (Last accessed: 09.12.2024).
12. Лекторіум, Криптографічні Хеш-функції URL:

<https://www.lektorium.tv/course/22746>. (Last accessed: 27.12.2024).

13. [Лапони́на О. Р. Криптографічні основи безпеки.](#) (Last accessed: 05.01.2024). (Last accessed: 26.12.2024).

14. What Is a Merkle Tree? URL: <https://www.investopedia.com/terms/m/merkle-tree.asp>. (Last accessed: 28.12.2024).

15. Consensus Algorithm URL: <https://whatis.techtarget.com/definition/consensus-algorithm>. (Last accessed: 25.12.2024).

16. Puddu I., Dmitrienko A., Capkun S. How to forget without hard forks. — IACR Cryptology, 2017. URL: <https://eprint.iacr.org/2017/106.pdf>. (Last accessed: 25.12.2024).

17. [Delegated Proof-of-stake](#) URL: <https://distributedlab.com/blog/ru/description-of-delegated-proof-of-stake>. (Last accessed: 27.12.2024).

18. Kwon J. Tendermint: Consensus without mining. URL: <https://github.com/tendermint/tendermint/wiki>. (Last accessed: 04.01.2024).

19. Chase B., Macbrough E. Analysis of the XRP ledger consensus protocol. Cornell University, 2018 URL: <https://arxiv.org/abs/1802.07242>. (Last accessed: 25.12.2024).

20. Класифікація блокчейнів URL: <https://www.bitbon.space/ru/knowledge-base/distributed-ledger-technologies-blockchain/technological-aspects-of-blockchain/classification-of-blockchains> (Last accessed: 01.01.2024).

22. D. Puthal, N. Malik, S. P. Mohanty, E. Kougianos, and G. Das, "Everything you Wanted to Know about the Blockchain", IEEE Consumer Electronics Magazine, Volume 7, Issue 4, July 2018, pp. 06–14. (Last accessed: 27.12.2024).

23. The Idea of Smart Contracts URL: https://web.archive.org/web/20060615044959/http://szabo.best.vwh.net/smart_contracts_idea.htm/. (Last accessed: 21.12.2024).

24. Solidity URL : <https://docs.soliditylang.org/en/v0.8.10/>. (Last accessed:

21.12.2024).

25. Ethereum virtual machine URL: <https://ethereum.org/en/developers/docs/evm/>. (Last accessed: 21.12.2024).

26. Geth Documentation URL: <https://geth.ethereum.org/> (Last accessed: 23.12.2024).

27. Narayanan, A., Bonneau, J., Felten, E., Miller, A., Goldfeder, S. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction. — Princeton University Press, 2016. — 336 с. (Last accessed: 25.12.2024).

28. Zcash Company. Zcash Protocol Specification URL: <https://z.cash/technology/protocol.html>. (Last accessed: 24.12.2024).

29. Monero Project. How Monero Provides Transaction Privacy URL: <https://www.getmonero.org/resources/moneropedia/ringCT.html>. (Last accessed: 05.01.2024).

30. Бондаренко В.В. Приватність у блокчейн-мережах: аналіз сучасних методів забезпечення. — Київ: Науковий журнал "Інформаційна безпека", 2020. — С. 56-72. (Last accessed: 05.01.2024).

31. Parizi, R. M., Homayoun, S., Yazdinejad, A., Dehghantanha, A., Choo, K.-K. R. Integrating Privacy Enhancing Techniques into Blockchains Using Sidechains // arXiv preprint arXiv:1906.04953, 2019. URL: <https://arxiv.org/abs/1906.04953>. (Last accessed: 05.01.2025).

32. Mödinger, D., Kopp, H., Kargl, F., Hauck, F. J. A Flexible Network Approach to Privacy of Blockchain Transactions // arXiv preprint arXiv:1807.11338, 2018. URL: <https://arxiv.org/abs/1807.11338>. (Last accessed: 04.01.2024).

33. Сміюн І. В. Способи забезпечення захисту інформації від несанкціонованого доступу на основі сучасних технологій Blockchain: магістерська дисертація. — Київ: НТУУ "КПІ ім. Ігоря Сікорського", 2021. — 86 с. URL: https://ela.kpi.ua/bitstream/123456789/45919/1/Smiun_magistr.pdf. (Last accessed: 04.01.2024).

34. Терещенко Г., Кириченко І. Аналіз і обґрунтування використання

наявних блокчейн-рішень для захисту цифрових активів // Innovative technologies and scientific solutions for industries. — 2024. — №1 (27). — С. 164-172. URL:<https://journals.uran.ua/itssi/article/view/301056/293267>., (Last accessed: 25.12.2024).

35. Ворса А. В. Дослідження методів забезпечення безпеки інформації в блокчейн-системах: дипломна робота. — Київ: НАУ, 2022. — 75 с. URL: https://er.nau.edu.ua/bitstream/NAU/55899/1/Дипломна%20робота_Ворса.pdf. (Last accessed: 25.12.2024).

36. Dannen, C. Introducing Ethereum and Solidity. — Apress, 2017. — 197 с. (Last accessed: 25.12.2024).

37. Shubin, I. , Karmanenko, O., Gorbach, T., Umyarov, K. The methods of adaptation in computer-based training systems Shubin, I. , Karmanenko, O., Gorbach, T., Umyarov, K. 54 2015 Information Technologies in Innovation Business Conference, ITIB 2015 - Proceedings, 2015, с. 64-67, 7355054 (Last accessed: 22.12.2024).

38. R. Caruana and A. Niculescu-Mizil, “An empirical comparison of supervised learning algorithms,” in Proceedings of the 23rd international conference on Machine learning, pp. 161–168, ACM, 2006. (Last accessed: 25.12.2024).

39. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “Smote: synthetic minority over-sampling technique,” Journal of artificial intelligence research, vol. 16, pp. 321–357, 2002. (Last accessed: 26.12.2024).

40. M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, “A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches,” IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 4, pp. 463–484, 2012. (Last accessed: 28.12.2024).

41. Фаулер М. Архітектура програмних додатків. — Київ: Знання, 2007 (Last accessed: 25.12.2024).