Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Факультет комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)
Кафедра кібербезпеки
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр
(освітній рівень)

на тему: " Ризики безпеки NFT технології та методи їх вирішення "

Виконав: студент (ка)

Спеціальності:
125 «Кібербезпека та захист інформації»
(шифр і назва напряму підготовки, спеціальності)

Пилипів П.В.
підпис (прізвище та ініціали)

Керівник Лечаченко Т. А.
підпис (прізвище та ініціали)

Нормоконтроль Тимощук Д. І.
підпис (прізвище та ініціали)

Завідувач кафедри Загородна Н.В.
підпис (прізвище та ініціали)

Рецензент Стоянов Ю.М.
підпис (прізвище та ініціали)

м. Тернопіль – 2024

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра кібербезпеки
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Загородна Н.В.
(підпис) (прізвище та ініціали)
«__» _____ 2024 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Магістр
(назва освітнього ступеня)

за спеціальністю 125 Кібербезпека та захист інформації
(шифр і назва спеціальності)

Студенту Пилипіву Павлу Володимировичу
(прізвище, ім'я, по батькові)

1. Тема роботи Ризики безпеки NFT технології та методи їх вирішення

Керівник роботи Лечаченко Тарас Анатолійович, доктор філософії., старший викладач кафедри КБ
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «20» 11 2024 року № 4/7-1112

2. Термін подання студентом завершеної роботи

3. Вихідні дані до роботи Інтернет джерела, технічна документація Solidity, Openzeppelin .

4. Зміст роботи (перелік питань, які потрібно розробити)
Що таке невзаємозамінні токени, їх історія та особливості; Процеси створення та використання невзаємозамінного токена; Сфери використання NFT; Ринки NFT; Блокчейн; Смарт-контракти; Стадарти токенів на блокчейні Ethereum; Ризики безпеки невзаємозамінних токенів; Вразливості смарт-контрактів; Вразливості блокчейн-мереж; Фішингові атаки; Вразливості NFT-гаманців та гаманців; Методи вирішення ризиків смарт-контрактів; Методи вирішення вразливостей у блокчейн-мережах; Підвищення безпеки транзакцій; Підвищення безпеки під час користування NFT; Охорона праці та безпека в надзвичайних ситуаціях; Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
Титулка. Мета, задачі. Наукова новизна та практичне значення роботи. Історія невзаємозамінних токенів. Особливості NFT. Технології невзаємозамінних токенів: блокчейн. Технології невзаємозамінних токенів: смарт-контракти. Стандарти токенів. Ризики невзаємозамінних токенів. Алгоритм аудиту смарт-контрактів. Методи вирішення вразливостей смарт-контрактів. Методи вирішення вразливостей блокчейн-мереж. Підвищення безпеки транзакцій завдяки оракулу. Функції requestRiskLevel та fulfillRiskLevel. Рекомендації з підвищення безпеки під час користування NFT. Напрямки майбутніх досліджень.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Осухівська Г.М., к.т.н., доцент		
Безпека в надзвичайних ситуаціях	Клепчик В.М., проректор з адміністративно-господарської роботи та будівництва		

7. Дата видачі завдання 24.09.2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	24.09	Виконано
2.	Підбір джерел за темою кваліфікаційної роботи	25.09 – 29.09	Виконано
3.	Опрацювання джерел в галузі дослідження	01.10-07.10	Виконано
4.	Огляд невзаємозамінних токенів та задіяних технологій	08.10-20.10	Виконано
5.	Аналіз ризиків безпеки NFT	21.10-30.10	Виконано
6.	Розробка методів вирішення або пом’якшення виявлених ризиків	04.11-11.11	Виконано
7.	Розробка смарт-контракту для виявлення ризиків транзакцій	12.11-20.11	Виконано
8.	Оформлення розділу «Аналіз предметної області»	20.11-26.11	Виконано
9.	Оформлення розділу «Технологія невзамозамінних токенів»	26.11-03.12	Виконано
10.	Оформлення розділу «Методи вирішення ризиків безпеки NFT»	04.12-09.12	Виконано
11.	Оформлення розділу «Охорона праці та безпека в надзвичайних ситуаціях»	10.12-12.12	Виконано
12.	Підготовка звіту та оформлення кваліфікаційної роботи	13.12-14.12	Виконано
13.	Нормоконтроль	14.12-16.12	Виконано
14.	Перевірка на плагіат	17.12	Виконано
15.	Попередній захист кваліфікаційної роботи	20.12	Виконано
16.	Захист кваліфікаційної роботи	27.12	Виконано

Студент _____ Пилипів П.В.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Лечаченко Т. А.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Ризики безпеки NFT технології та методи їх вирішення// Кваліфікаційна робота ОР «Магістр» //Пилипів Павло Володимирович// Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра кібербезпеки, група СБм-62 // Тернопіль, 2024 // С. 85 , рис. – 1, табл. – __ , кресл. –__ , додат. – 6.

Ключові слова: аналіз, невзаємозамінні токени, блокчейн, смарт-контракт, Ethereum, ризик, вразливості.

Кваліфікаційна робота присвячена аналізу ризиків та розробці методів їх вирішення або пом'якшення завдяки покращенню безпеки смарт-контрактів, блокчейн-мереж, підвищенню безпеки транзакцій з використанням смарт-контракту для оцінки їх ризику.

У першому розділі кваліфікаційної роботи висвітлено історію NFT технології, процеси створення та використання невзаємозамінних токенів, сфери їх використання та торгові майданчики для даних токенів.

У другому розділі кваліфікаційної роботи розглянуто технології, застосовані у невзаємозамінних токенах, такі як блокчейн, смарт-контракти та стандарти ERC-20, ERC-721, ERC-1155 та BRC-721E. Проведено аналіз ризиків безпеки невзаємозамінних токенів, зокрема вразливості смарт-контрактів, блокчейн-мереж, бірж NFT та гаманців, а також небезпека фішингових атак.

У третьому розділі кваліфікаційної роботи подані методи вирішення або пом'якшення ризиків безпеки NFT. Окремо висвітлено методи вирішення ризиків пов'язаних зі смарт-контрактами та блокчейн-мережами. Запропоновано смарт-контракт для оцінки ризиків транзакцій на основі даних, отриманих від оракула. Подано методи покращення безпеки під час використання NFT, зокрема вибір безпечного гаманця, підвищення його безпеки, критерії вибору надійного торгового майданчика та методи уникнення фішингових атак.

ABSTRACT

Security Risks of NFT Technology and Methods for Addressing Them// Thesis of educational level "Master" // Pylypiv Pavlo Volodymyrovych // Ternopil Ivan Puluj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Cybersecurity, SBm-62 group // Ternopil, 2024 // P. 85 , fig. – 1, table. – __ , chair. – ___ , added. - 6.

Keywords: analysis, non-fungible tokens, blockchain, smart contract, Ethereum, risk, vulnerability.

The qualification work is dedicated to analyzing risks and developing methods for resolving or mitigating them by improving the security of smart contracts, blockchain networks, and increasing the security of transactions using a smart contract to assess their risk.

The first section of the qualification work covers the history of NFT technology, the processes of creating and using non-fungible tokens, their areas of use, and trading platforms for these tokens.

The second section of the qualification paper examines the technologies used in non-fungible tokens, such as blockchain, smart contracts, and the ERC-20, ERC-721, ERC-1155, and BRC-721E standards. An analysis of the security risks of non-fungible tokens is conducted, including the vulnerabilities of smart contracts, blockchain networks, NFT exchanges, and wallets, as well as the risk of phishing attacks.

The third section of the qualification work presents methods for addressing or mitigating NFT security risks. Methods for addressing risks associated with smart contracts and blockchain networks are separately highlighted. A smart contract is proposed to assess transaction risks based on data received from an oracle. Methods for improving security when using NFT are presented, in particular, choosing a secure wallet, increasing its security, criteria for choosing a reliable trading platform, and methods for avoiding phishing attacks.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

NFT (Non-Fungible Token) – вид криптографічних токенів, кожен екземпляр якого унікальний та не може бути замінений іншим аналогічним токеном.

RFID (Radio frequency identification) – радіочастотна ідентифікація.

IoT (Internet of Things) – інтернет речей.

PoW (Proof of Work) – протокол консенсусу у блокчейн мережах.

PoS (Proof of Stake) – протокол консенсусу у блокчейн мережах.

EVM (Ethereum Virtual Machine) – віртуальна машина Ethereum.

CVL (Certora Verification Language) – мова для написання специфікацій для смарт-контрактів в інструменті Cetora Prover.

BGP (Border Gateway Protocol) – протокол маршрутизації між автономними системами в глобальній мережі Інтернет.

RPKI (Resource Public Key Infrastructure) – ієрархічна система відкритих ключів.

JSON (JavaScript Object Notation) – текстовий формат обміну даними, заснований на JavaScript.

ВСТУП

Технологія NFT стала новим кроком у розвитку криптографічних токенів. На відміну від того ж Bitcoin, кожен NFT-токен є унікальним та неподільним активом, який неможливо просто обміняти на інший NFT без втрати його якості та ціни.

Дані токени можна використовувати для підтвердження та реєстрації права власності на цифрові та фізичні активи стабільним та прозорим способом. Головною функцією та особливістю NFT є унікальність, яка може бути легко підтверджена за допомогою блокчейну, завдяки якому такі токени використовуються у якості цифрових ідентифікаторів (ID), які підтверджують справжність та унікальність цифрового або токенізованого фізичного активу. Завдяки NFT можна автентифікувати витвір цифрового мистецтва або цифровий предмет колекціонування, а також ідентифікувати його законного власника та авторські чи комерційні права, якими він володіє.

Після стрімкого підйому у 2020-2021 роках, ринок NFT зазнав спаду, проте даний токен досі користується достатньою популярністю. На даний момент NFT-транзакції є досі поширеними про що свідчить кількість унікальних гаманців за день – більше 300 тисяч. Також на торгових майданчиках продовжують відбуватись транзакції з високою вартістю, що демонструє постійний інтерес до цифрових активів вищого рівня. Станом на вересень 2024 року загальна вартість продажів за 30 днів склала приблизно 12 мільйонів доларів США.

Оскільки величезний і різноманітний простір для технологій штучного інтелекту та блокчейн продовжує розвиватися, NFT стали потужним інструментом для перевірки автентичності даних та ідентифікації власності, існує великий потенціал для застосування NFT у сферах освіти, охорони здоров’я, життя, медицини та права.

Оскільки невзаємозамінні токени все ще користуються популярністю і мають достатні перспективи використання у різних сферах, питання безпеки

даної технології є одною з причин стабільного та безпечного розвитку цього ринку.

Метою даної кваліфікаційної роботи є дослідження та аналіз ризиків технології невзаємозамінних токенів, а також методи їх вирішення. Для досягнення поставленої задачі було здійснено аналіз NFT технології, її ризиків та пов'язаних з ними вразливостей NFT, запропоновано методи вирішення або пом'якшення ризиків, розроблено смарт-контракт для визначення ризику транзакції на основі інформації від оракула, розроблено рекомендації щодо підвищення безпеки під час використання невзаємозамінних токенів.

Об'єктом дослідження є технологія невзаємозамінних токенів. Предметом дослідження виступають ризики безпеки, що виникають під час створення та використання NFT, а також методи їх вирішення для підвищення захищеності даної технології

Наукова новизна роботи полягає у комплексному аналізі ризиків NFT технології та методів їх вирішення на кожному з її рівнів, зокрема подані методи вирішення ризиків смарт-контрактів та блокчейн мереж, запропоновано смарт-контракт для підвищення безпеки під час проведення транзакцій, що дозволяє покращити захищеність NFT технології, а також розроблено рекомендації щодо підвищення рівня безпеки для користувачів даною технолгією.

Практичне значення результатів роботи полягає у підвищенні безпеки невзаємозамінних токенів за допомогою вирішення можливих їх вразливостей та використання смарт-контракту для оцінки ризику транзакцій. Результати роботи можуть бути використанні розробниками під час проектування та створення смарт-контрактів та блокчейн мереж, а також користувачами NFT.

Апробація результатів магістерської роботи. Основні результати проведених досліджень обговорювались на: XII науково-технічна конференції «Інформаційні моделі, системи та технології».

Публікації. Основні результати кваліфікаційної роботи опубліковано у працях конференції (див. Додаток А).

РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Що таке невзаємозамінні токени, їх історія та особливості

NFT або невзаємозамінний токен – це криптографічний сертифікат цифрового об’єкту з можливістю передачі сертифікату через блокчейн. Окремо токен не є підтвердженням права на володіння цифровим активом у контексті законодавства про авторське право, оскільки токен є певним стандартизованим записом і містить лише посилання у метаданих на місце зберігання твору в цифровій формі [1].

Дана технологія базується на використанні блокчейн-системи Ethereum та смарт-контрактів. Смарт-контракт (англ. smart contract) – це код, який детерміновано виконується у контексті мережі блокчейн. Кожен учасник такої мережі перевіряє операції зміни стану, які виконує код смарт-контракту. Смарт контракти є основним засобом, за допомогою якого розробники можуть створювати токени та керувати ними у блокчейні. Такі контракти можуть зберігати невеликі обсяги інформації у загальних структурах даних, що є критично важливим компонентом випадків використання токенізації, які зіставляють ідентифікатори токенів ідентифікаторами власників, щоб відстежувати хто володіє певним токеном.

NFT-токени почали використовуватись у блокчейні для вирішення проблеми створення та управління унікальними цифровими активами. До появи даних токенів проблема підтвердження права власності та визначення цінності цифрових активів, таких як мистецтво, музика чи відео, була актуальною. Невзаємозамінні токени дозволяють створювати унікальні цифрові активи, право власності та автентичність яких можна легко перевірити за допомогою технології блокчейн. Це надає нові можливості для авторів та колекціонерів, дозволяючи їм монетизувати цифрові активи.

Історію появи NFT можна пов’язати зі створенням технології радіочастотної ідентифікації (RFID), розробленої у 70-80-х роках минулого століття для

відстеження запасів та активів за допомогою радіохвиль. Свою популярність RFID отримала наприкінці 1990-х років, коли компанії стали використовувати дану технологію для відстеження своїх продуктів, активів та менеджменту ланцюгів постачання.

Перші дослідження та ідеї невзаємозамінних токенів було висунуто у 2009 разом із запуском мережі Bitcoin, яка використовувала технологію блокчейн для створення децентралізованої цифрової валюти. Дана технологія за допомогою розподіленого реєстру дозволяє безпечно та прозоро записувати та відстежувати транзакції. Проте тодішні ідеї так і залишились ідеями до 2015, коли NFT у мережі Ethereum були вперше створені.

Експерименти з NFT почались у 2012-2014 роках на скриптовій мові блокчейну Bitcoin на проектах Colored Coins та Counterparty. Colored Coins не були токенами зі складною інфраструктурою, натомість це були лише біткоїни або їх частки (satoshi), яким були присвоєні унікальні ідентифікатори.

4 грудня 2012 року криптограф та президент біткоїн-асоціації Мені Розенфельд опублікував статтю «Overview of Colored Coins» у якій представив механізм використання «взаємозамінності» біткоїна для виокремлення монет для певних цілей [2]. Він розглянув інші варіанти розвитку такої технології, додавши інші особливості монети, щоб відокремити їх від решти. У закритих середовищах користувачі могли додати до транакцій додаткові дані, наприклад ідентифікатори або хешовані документи у дерево Меркла. Це могло дозволити налаштовувати токени за допомогою метаданих, які можна було б використати у реальному світі для представлення певного активу, який був оцифрований і накладений на біткоїни у блокчейні.

У 2014 році було створено фінансову платформу Counterparty, яка працювала як розширення протоколу Bitcoin. Платформа дозволяє створювати цифрові активи, випускати токени, виплачувати дивіденди та торгувати бінарними опціонами [3]. Цей проект використовувався як майданчик для обміну невзаємозамінними токенами та внутрішньою валютою відомою як XCP, яка створена шляхом відправлення біткоїнів у «dead-address», що робить їх

непридатними для використання. Після цього користувач отримував суму XCP пропорційну кількості втрачених біткоїнів.

Також, у 2014 році було розроблено криптовалюту Ethereum, яка окрім блокчейну тажко використовувала технологію смарт-контрактів, які виконуються самі та містять певні визначені наперед умови контракту, записані у коді. Це дозволило розробникам створювати децентралізовані додатки (DApps) та зробило можливим розробку нових варіантів використання мережі Ethereum.

У жовтні 2015 року був запущений перший повноцінний NFT-проект Etheria, який був продемонстрований на першій конференції розробників Ethereum – DEVCON1. Тоді було продано 457 шестикутних плиток Etheria не продавались більше наступних 5 років, поки 13 березня 2021 року, коли зацікавлення до NFT знову зросло, не були продані на загальну суму 1.4 мільйони доларів США.

У листопаді 2016 року був запущений перший NFT-проект, який зберігає зображення безпосередньо у блокчейні – PixelMap. Проте через падіння зацікавлення у технології проект був закритий. Але у серпні 2021 року PixelMap був заново відкритий та продав перші 3000 плиток за 3.3 мільйони доларів.

У 2017 на блокчейні Ethereum з’явився новий тип токенів – ERC-721. На відміну від звичайних взаємозамінних криптовалют, дані токени були унікальними та неподільними. Такі токени довзолили створювати та торгувати унікальними цифровими активами, зокрема музикою, відео та іншими витворами мистецтва.

Вперше смарт-контракт NFT з використанням стандарту ERC721 було використано у 2017 році у мережі Ethereum як гра «Crypto Kitties». У ній можна вирощувати та «розводити» криптовалюту у формі котів. Один крипто-кіт – неподільний об’єкт, який має власний унікальний номер, ген із 256-бітної ДНК та 12 атрибутів, які можна передати нащадкам – колір очей, форму голови, тощо. У NFT вшита інформація про творця кота, а також решти його віртуальних котів. У 2018 році найдорожчий криптокіт був проданий за 140 тисяч доларів.

У другій половині 2018 року було представлено новий стандарт токенів – ERC-1155, який усунув обмеження попередніх стандартів токенів ERC-20 та ETC-721. Даний стандарт дозволяв підтримувати кілька типів токенів та кілька активів у рамках одного смарт-контракту, що зменшило складність та витрати, пов'язані з проведенням кількох контрактів.

У лютому 2020 року Decentraland, віртуальний світ, заснований на блокчейн, розробка якого почалась у 2015 році, було відкрито публіці. У перших версіях даного проекту світ являв собою піксельну сітку, яка дозволяла володіти пікселями за допомогою доказу виконання роботи. З часом Decentraland було перетворено у тривимірну платформу. Під час свого бета-тестування у 2017, розробники продавали ділянки за 20 доларів США кожен, проте з ростом популярності ринку NFT, ціни на дані ділянки зросли до 100тис. доларів.

У 2020 році ринок NFT швидко виріс, його вартість збільшилась до 250 мільйонів доларів. За перші три місяці 2021 року на NFT було витрачено більше 200 мільйонів доларів США.

У 2021 році вперше було перетворено матеріальний витвір мистецтва, картину Бенксі «Morons», на цифровий актив – було створено невзаємозамінний токен, а оригінальний матеріальний екземпляр картини було знищено (спалено).

Проте у травні 2022 року ринок NFT почав стрімко падати. Продажі токенів NFT за день знизились на 92% у порівнянні з вереснем 2021 року, а кількість активних гаманців на ринку упало на 88% у порівнянні з періодом найбільшої популярності у листопаді 2021 року. До вересня 2023 року більше 95% колекцій NFT мали нульову оціночну вартість.

Не зважаючи на падіння ринку за минулі роки, на разі NFT все ще користується популярністю, оскільки кількість унікальних гаманців за день сягає близько 300 тисяч. Також періодично здійснюються транзакції з високою вартістю, що знову ж демонструє зацікавленість користувачів у даній технології. На думку аналітиків, через декілька обіг коштів на ринках NFT складатиме більше 200 мільярдів доларів США.

Характеристики NFT відрізняють їх від інших типів цифрових активів, а також забезпечують універсальність для різних цілей. До ключових особливостей невзаємозамінних токенів можна віднести:

- унікальність – кожен NFT є унікальним і не може бути зміненим або відтвореним. Це дозволяє створювати унікальні цифрові активи, які можна перевірити та відстежити;
- підтвердження права власності – NFT використовують технологію блокчейну для запису та відстеження права власності, що дозволяє передавати дане право у безпечний спосіб, який можна перевірити;
- сумісність NFT з іншими технологіями – невзаємозамінні токени створюються за допомогою технології блокчейн, що дозволяє інтегрувати їх з іншими платформами та додатками на основі блокчейну, що робить їх більш універсальними;
- децентралізованість – оскільки NFT-токени створені на основі децентралізованої технології блокчейну, яка гарантує, що вони не контролюються жодним центральним органом, що робить їх більш безпечними, прозорими та стійкими до цензури;
- незмінність – усі транзакції пов’язані з невзаємозамінними токенами, а також інформація про власників зберігається у реєстрі блокчейну, який неможливо змінити;
- програмованість – NFT можна запрограмувати для підтримки правил та обмежень, які регулюють умови використання та передачі активів, надаючи авторам більше контролю над своїми активами;
- можливість перевірки – дані про створення, також відоме як карбування або мінтинг, передача та знищення токена постійно зберігаються у блокчейні, що дозволяє перевірити повну історію NFT-токена;
- гнучкість – за допомогою невзаємозамінного токена можна представити різні цифрові активи, зокрема віртуальна нерухомість, музика, фільми та живопис.

1.2 Процеси створення та використання невзаємозамінного токена

Зазвичай процес створення та використання NFT-токена складається з шести пунктів:

- створення токена – особа або організація створює унікальний цифровий актив, для прикладу витвір мистецтва або предмет колекціонування, додає до нього метадані (опис та унікальний ідентифікатор);
- карбування – автор створює (карбує) NFT на блокчейні, додаючи цифровий актив та його метадані у блокчейн та випускаючи унікальний цифровий токен, який символізує право власності на актив. Процес карбування описаний детальніше нижче;
- торгівля токенами – NFT-токени можна купити, продати або обміняти на ринку, який підтримує дану технологію. Дані ринки зазвичай побудовані на основі блокчейну, наприклад Ethereum;
- передача права власності – при продажі NFT, право власності на даний цифровий актив передається від продавця до покупця у блокчейні;
- перевірка права власності на цифровий актив – дана перевірка може бути легко здійснена, оскільки історія транзакцій зберігається у блокчейні, що дозволяє швидко її перевірити;
- використання – новий власник NFT-токена може використовувати токен як вважає за потрібне, але відповідно до обмежень вказаних автором.

Карбування, також відомий як процес створення або мінтинг (від англ. minting – карбування) – це процес запису цифрового елементу у блокчейн, що дозволяє довести право на його власність та законність

Створення NFT-токена відбувається шляхом виклику відповідного методу у контракті токена, який у більшості випадків відповідає стандартам ERC-721 або ERC-1155. Контракт одного токена може контролювати право власності на кілька NFT.

У процесі створення, кожному токену присвоюється унікальний ідентифікатор відомий як tokenId. У поєднанні з token_contract_address, адресою розташування смарт-контракту, який керує балансами власників токенів, допомагає ідентифікувати конкретний NFT у блокчейні.

Процес карбування токенів може базуватись на основі трьох алгоритмів: використання звичайного контракту, його копії або використання зовнішнього контракту.

Використання звичайного контракту під час процесу карбування означає, що токен створюється як частина попередньо створеного договору про визначений маркер, яким керує ринок. Якщо автори не розгортають спеціальний контракт, такі ринки як OpenSea, Foundation або SuperRare пропонують угоду за замовчуванням для зберігання NFT.

Під час використання алгоритму з копією контракту, біржа ініціює контракт від імені автора колекції до якої належить NFT-токен. Розгорнуті контракти мають ідентичний байтовий код, проте їх можна персоналізувати за допомогою параметрів ініціалізації. Біржами, що використовують даний тип створення токенів є Nifty та Rarible. Оскільки звичайними контрактами та копіями контрактів керують біржі, дані типи належать до контрактів внутрішніх токенів;

У випадку карбування токена за допомогою зовнішнього контракту, автор самостійно розгортає спеціальний контракт для керування колекцією і після чого імпортує його на ринок. Для сумісності з різними NFT-біржами зовнішні контракти повинні відповідати встановленим стандартам токенів, інакше в іншому випадку може знадобитись спеціальна інтеграція.

Кожен процес карбування токенів можна розбити на наступні етапи:

- ініціація процесу мінтингу NFT з клієнтської програми, надаючи запит на створення NFT;
- підтвердження запиту майнером;
- якщо запит узгоджений, служба відповідальна за створення токенів використовує блокчейн для генерації токена NFT;
- блокчейн підтверджує створення токена та надсилає відповідь майнеру;

- генерація метаданих майнером використовуючи відповідні сервіси;
- служба генерації метаданих надсилає відповідь майнеру;
- збереження NFT у блокчейні майнером;
- підтвердження збереження метаданих NFT та отримання майнером відповіді;
- майнер повідомляє клієнтську програму про успішне або невдале карбування невзаємозамінного токена;
- запит на розміщення створеного NFT-токена на ринку;
- підтвердження або відмова у розміщенні токена.

Процес карбування NFT подано на рисунку 1.1.

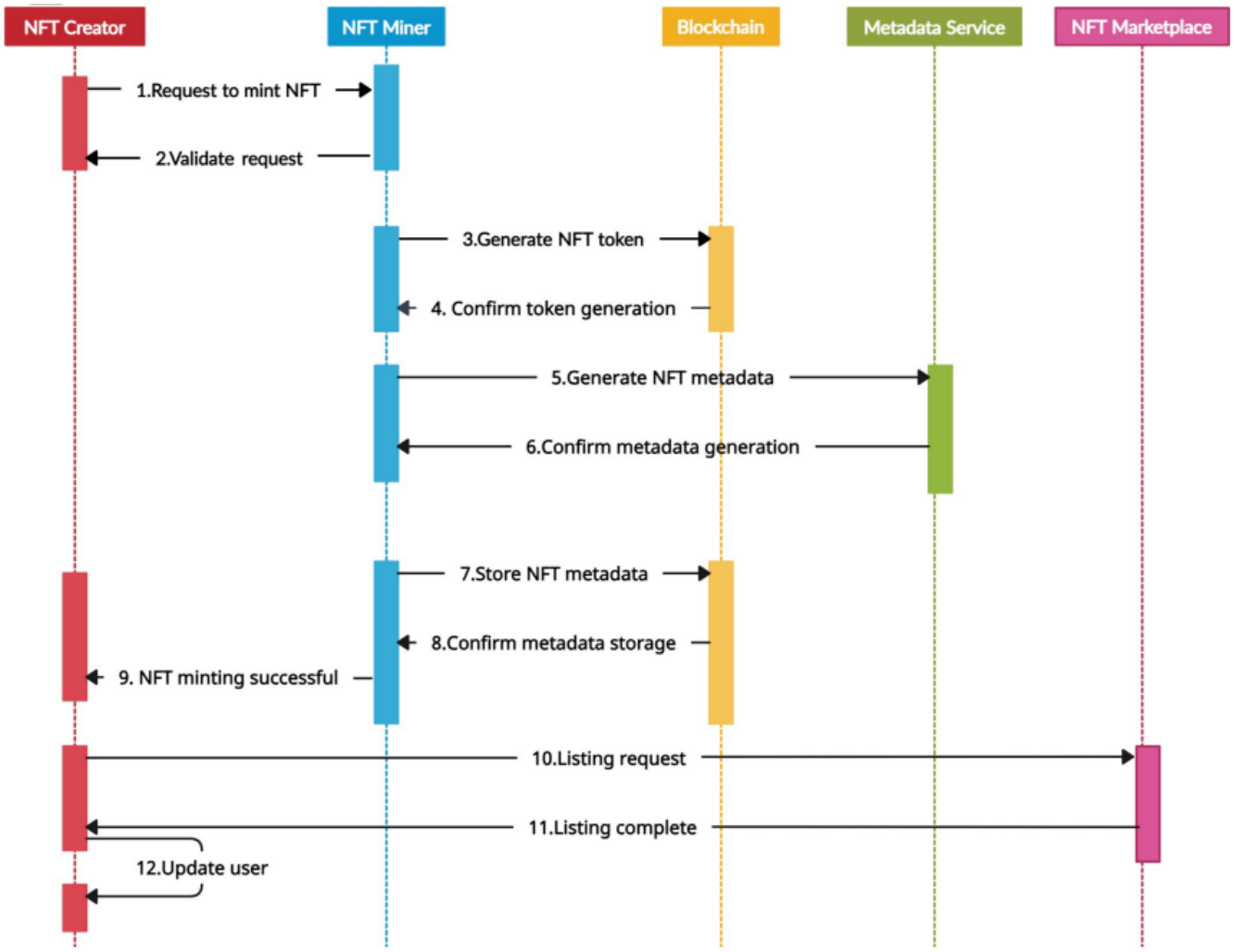

Рисунок 1.1 – Процес створення невзаємозамінного токена

1.3 Сфери використання NFT

NFT є популярним рішенням у сфері цифрового контенту. Зараз цифрове мистецтво є найпоширенішим варіантом використання невзаємозамінних токенів, цінність яких пов’язана з цифровою автентичністю та правом власності, які вони надають. Цифрове відтворення та розповсюдження мистецтва створюють значні перешкоди для успішного оцифрування мистецтва, враховуючи легкість, з якою цифрові файли можна дублювати, ділитися та поширювати. Сучасні формати зображень є повністю взаємозамінними та містять ідентичні метадані, ускладнюючи забезпечення ексклюзивності. Проте NFT дозволяють створювати унікальні цифрові активи, тим самим даючи цінність тому чи іншому активу. Цифрові митці можуть продавати свої твори, користуючись тими ж перевагами, що і творці фізичних витворів мистецтва.

Також NFT використовуються в ігровій індустрії для створення унікальних елементів внутрішньоігрової або віртуальної нерухомості. Гравці можуть купувати або продавати NFT певних предметів, які можна використовувати у грі або торгувати на сторонніх ринках. Використання невзаємозамінних токенів в іграх дозволяє гравцям та інвесторам купувати та продавати ігровий контент, створюючи значні доходи та відсотки. Крім того, NFT можна використовувати для створення унікальних винагород для гравців або груп [4]. Також NFT може використовуватись для створення гнучкої та децентралізованої системи для обміну ігровими активами [5]. Запропоноване рішення гарантує, що активи завжди будуть доступні у блокчейні без жодної точки збою, тому користувачі не ризикують втратити контроль над своїми предметами або цінністю, навіть якщо розробник гри втратить інтерес або оголосить про банкрутство.

На даний момент все висувається все більше ідей з використання невзаємозамінних токенів для ідентифікації особистості. Науковці та розробники пропонують присвоєння кожному користувачу особистого NFT, який може використовуватись для ідентифікації їхньої особистості для різних цілей, зокрема отримання державних послуг або відкриття банківського акаунту [6].

Використання даного методу мають певні переваги:

- посилена криптографічна безпека за рахунок використання технології блокчейн, яка сприяє надійному захисту від зламу та шахрайства, що створює безпечну основу для екосистеми цифрової ідентифікації;
- спрощений метод ідентифікації особи скорочує витрати часу та ресурсів приносить користь кінцевим користувачам, пришвидшуючи їхню перевірку, а також постачальникам послуг, оптимізовуючи їхні операційні ресурси;
- незалежність користувачів від центрального управління за рахунок децентралізації, що надає користувачам більший контроль над своєю ідентичністю, а також зменшує ризики, пов’язані з одноточковими збоями, забезпечуючи цифрову ідентифікацію, орієнтовану на користувача;
- незмінність інформації записаної у блокчейн гарантує постійну цілісність та автентичність цифрових ідентифікаторів, тобто після закріплення у блокчейні дані залишаються захищеними від втручання, забезпечуючи основу довіри та надійності цифрової ідентифікації;
- простий аудит транзакцій цифрової ідентичності, завдяки притаманній блокчейну прозорості, забезпечує довіру між користувачами та зацікавленими сторонами, запевняючи їх у цілісності та підзвітності, що є важливим у системі цифрової ідентифікації;
- здатність до взаємодії з іншими платформами та системами забезпечується дотриманням NFT стандартизованих протоколів.

Ґрунтуючись на ідеї ідентифікації особистості, NFT можна використовувати у безлічі сферах життєдіяльності, зокрема у концепції розумних міст.

Створюючи унікальні NFT для кожного громадянина, системи голосування на основі блокчейну можуть забезпечити більшу безпеку та прозорість процесу голосування. Завдяки розподіленій системі технологія блокчейн може вирішити проблеми пов’язані з традиційними системами електронного голосування. Пропонується використання смарт-контрактів на платформі Ethereum для розробки надійної електонної системи голосування [7]. Традиційні системи

голосування за допомогою бюлетнів у скриньках мають багато проблем, у тому числі недостатню прозорість та потенціал для корупції.

Також NFT може використовуватись для створення унікальної віртуальної нерухомості, такої як віртуальна ділянка або будівля, яка може бути куплена або продана, як фізична нерухомість. Остаточність активу може бути встановлена шляхом токенізації активів нерухомості та шифрування даних, а також створенням цифрового права власності. У даному випадку NFT реєструє статичну та динамічну інформацію про нерухомість для торгівлі на ринку даних між різними споживачами та зацікавленими сторонами, зокрема інвесторами, мешканцями, страховими компаніями та менеджерами нерухомості. Інформація, що записана у NFT, включає акти нерухомості, транзакції та різні змінні параметри [8].

Ще одним варіантом застосування NFT у концепції розумного міста є система децентралізоване розумне місто речей (Decentralized Smart City of Things, DSCoT). Це приватна блокчейн-архітектура, що використовує блокчейн-токенізацію, зокрема NFT, для унікальної ідентифікації та перевірки активів користувачів та інтернету речей (Internet of Things, IoT). Дана архітектура гарантує унікальне представлення активів за допомогою смарт-контрактів та автентифікації користувачів. Ця концепція є ефективною стосовно складності виконання та розмірів комісій за транзакцій, відомих як «gas fee», оскільки набір функцій набір функцій для запиту смарт-контракту щодо статусу активів у реєстрі NFT не стягує жодних комісій за транзакції [9]. Використання технології блокчейну та механізмів шифрування SHA-3 надає безпеку та ефективність для DSCoT.

Ще одною галуззю використання невзаємозамінних токенів може стати медицина. Зокрема, для зміни системи управління даними пацієнтів. На заміну паперовій документації, блокчейн дозволяє створення єдиного безпечного та прозорого запису, який належить самому пацієнту. Це дозволить вирішити проблему втрати та дублювання інформації, а також обмежень у доступі до неї [10].

Використання NFT також може вирішити проблему підроблення медичних пристроїв, оскільки існуючі системи управління використовують централізовану архітектуру, яка схильна до повної відмови або виходу з ладу систем і усуває прозорість та можливість перевірки. Пропонується використовувати NFT для створення цифрового аналогу медичного пристрою, який буде збирати важливу інформацію про медичний пристрій на протязі його роботи, що дозволить відстежувати та керувати правом власності на медичні пристрої та вирішити проблему з цілісністю даних, походженням та відстежуваністю медичних пристроїв. Схоже рішення може бути використане для відстеження медичних препаратів. Також пропонується використання NFT для відслідковування результатів клінічних випробувань, надійно записуючи всі результати випробувань, забезпечуючи їх цілісність та надійність.

Ланцюги постачання та логістика – це сектори, які також можуть отримати вигоду від використання технології NFT. Токени здатні гаранантувати автентичність та прозорість транзакцій. NFT можна використати для створення унікального цифрового відбитку для кожного товару, що дозволяє легко перевірити його справжність, що важливо для брендів з хорошою репутацією та колекціонерів. Також, за допомогою NFT можна спростити відстеження власності на товари протягом усього ланцюга постачання, забезпечуючи прозорість та мінімізуючи ризики підробок. Смарт-контракти на основі NFT автоматизовують багато процесів, зменшуючи потребу в ручній обробці документів.

Застосування NFT у сфері освіти дозволить створювати та розповсюджувати цифрових сертифікатів. Традиційні паперові варіанти можна легко втратити, підробити або пошкодити, що може призвести до потенційних проблем із підтвердженням навчальних досягнень. Університети та школи можуть токенізовувати досягнення студентів та видавати цифрові дипломи, які буде неможливо підробити. Невзаємозамінні токени можуть змінити процес створення, розповсюдження та монетизації освітнього контенту. Педагоги зможуть токенізовувати свої навчальні матеріали, наприклад книги, плани уроків

або онлайн-курси, як NFT. Дані токени можуть бути формою визнання та співпраці, сприяючи культурі надійного та безпечного обміну знаннями та інноваціями в освітніх спільнотах.

1.4 Ринки NFT

Ринок NFT – це онлайн-платформа, де користувачі можуть створювати та торгувати невзаємозамінними токенами. Ці платформи використовують блокчейн як основну технологію для забезпечення прозорості кожної транзакції та запису торгового процесу та токенізації активів. Біржі NFT можуть працювати з будь-яким ринком цифрових активів.

Нове покоління ринків невзаємозамінних токенів спрямоване на вирішення проблем відсутності сумісності NFT, обміну інформацією між різними блокчейнами, а також сумісності з Metaverse.

Торгові майданчики об’єднують покупців та продавців цифрових предметів колекціонування, які зберігаються у блокчейні. Більшість ринків NFT пропонують багато функцій, створених для зручності та ефективності під час торгівлі.

Користувачі можуть здійснювати пошук токенів, перевіряти інформацію про NFT та його походження, а також робити ставки на аукціонах. Після завершення транзакції оплата здійснюється за допомогою смарт-контрактів, які забезпечують надійний переказ коштів між покупцем та продавцем. Крім цього, NFT-ринки запроваджують нормативні рамки для дотримання міжнародних законів та правил, пов’язаних із купівлею криптовалюти.

Найпопулярнішими торговими майданчиками NFT, завдяки своїй прозорості, невисоким комісіям та великому обсягу доступних NFT, є Rarible, OpenSea, SuperRare, Foundation та Mintable.

Платформа Rarible дозволяє користувачам карбувати, купувати та продавати цифрові токени. Інтерфейс є інтуїтивно зрозумілим, що полегшує навігацію. Крім цього, Rarible містить елементи соціальних мереж, наприклад

стеження за авторами NFT та сповіщення про нові NFT-колекції. Платформа також представила власний токен RARI, який використовується як інструмент управління платформою та дозволяє користувачам впливати на розвиток платформи. Також, Rarible використовує блокчейни Polygon, Immutable X, zkSync Era, Base, Astar zkEvm та Kroma [11]. Платформа стягує комісію у розмірі 5%, які розподіляє порівну між покупцем та продавцем, а максимальний розмір створеного токена повинен не перевищувати 30 мегабайтів, що можна вважати мінусом;

Ринок OpenSea, заснований у 2017 році, є одним з перших та найбільших торгових майданчиків для торгівлі невзаємозамінними токенами, який обслуговує власників цифрових активів Polygon, Ethereum та Bitcoin. Платформа пропонує токенізовані витвори мистецтва, музику, ігрові предмети, доменні імена та інші цифрові активи.

На відміну від Rarible, OpenSea не стягує комісій за створення NFT та вимагає меншої комісії при покупці – 2.5% від вартості. Даний майданчик використовує різні блокчейни, зокрема Flow, Tezos, Binance Smart Chain, Klatytn та інші. З мінусів даного ринку можна виділити обмеження на розмір файлів, тут воно складає 100 мегабайт, а також можливість розрахунку лише за допомогою криптовалюти [12];

Ринок SuperRare є першим ринком мистецтва NFT, що сфокусований на ексклюзивних токенах. Майданчик одобрює лише 1% заявок на створення невзаємозамінних токенів, щоб забезпечити цінність витвору мистецтва. Даний ринок заохочує користувачів карбувати оригінальні твори мистецтва та створювати NFT у єдиному екземплярі, щоб підвищити дефіцит та рідкість творів мистецтва. Мінусами даного маркетплейсу є певний період очікування на дозвіл створення NFT, а також високі комісії у розмірі 15% від продажів [13];

Foundation було створено у 2021 році з унікальною можливістю для авторів NFT – вони можуть постійно отримувати 10% роялті від своєї роботи, проте максимальна кількість авторів для одного токена не перевищує чотирьох осіб. Як і SuperRare, Foundation приймає лише найкращі NFT забезпечуючи високу якість

та ексклюзивність кожного невзаємозамінного токена [14]. Усі транзакції проводяться за допомогою технології Ethereum та продала NFT на більш ніж кілька сотень мільйонів доларів США. Серед мінусів даної платформи варто виділити високі комісії, що складають 15% від продажів, відсутність фільтрації невзаємозамінних токенів та максимальну величину файлів, що складають 50 мегабайтів;

Останній ринок NFT, Mintable, дозволяє користувачам створювати та торгувати невзаємозамінними токенами. Він надає користувачам можливість торгувати NFT у категоріях цифрового мистецтва, відео, ігрових предметів, доменних імен, тощо. Також автор має можливість встановити власні роялті на свої токени, які можуть сягати 90%. Даний проект використовує блокчейни Ethereum, Immutable X та Ripple. Комісії складають 2.5% від вартості, проте використання блокчейну Immutable X може пришвидшити виконання транзакцій та знизити комісії до нуля [15].

РОЗДІЛ 2 ТЕХНОЛОГІЯ НЕВЗАЄМОЗАМІННИХ ТОКЕНІВ

2.1 Технології застосовані у невзаємозамінних токенах

В основі невзаємозамінні токенів лежать технології блокчейну та смарт-контрактів. Усунувши посередника, блокчейн сприяє відкритості та надійності як розподіленого так і незмінного запису. Дана технологія може автоматизувати транзакції за допомогою смарт-контрактів, а угоди можуть бути представлені у вигляді самовиконуваного коду. NFT, засновані основі блокчейну, є унікальними цифровими активами з можливістю перевірки прав власності на них. За допомогою смарт контрактів розробники можуть додавати до невзаємозамінних токенів умови роялті, тобто платежі за використання, а також створювати децентралізовані ринки, які поєднують споживачів та виробників.

Поєднання даних технологій розширило можливості авторів, змінивши такі галузі як мистецтво, ігри та нерухомість, а також є потенціал для трансформації інших секторів, зокрема фінансів та постачання. Інтеграція блокчейну, смарт-контрактів та невзаємозамінних токенів може бути використана також у технології розподіленого реєстру.

2.1.1 Блокчейн

Блокчейн – це децентралізований розподілений реєстр, який полегшує процес запису транзакцій та відстеження активів у бізнес-мережі [16]. У блокчейні перед збереженням нової транзакції у новому блоці відбувається перевірка у розподіленій мережі. Вперше дана ідея була висунута Сатоші Накамото у статті «Біткоїн: електронна пірингова система готівки» також відомій як «Біла книга». На даний час блокчейн набирає популярності у різних сферах, зокрема фінансах [17], інтернеті речей [19], безпровідних транспортних мережах VANET та мережах розумних міст [20], а також у сфері БПЛА [21].

Завдяки функціям прозорості, децентралізації, швидкості та безпеки блокчейн користується популярністю у сучасному світі. Сама технологія побудована на наступних принципах: блоках, хешуванні та алгоритмі консенсусу.

Блоки – у них зберігаються записи. Ці блоки з'єднані один за одним, утворюючи ланцюжок, який і є блокчейном. Кожен блок цієї послідовності містить хешований вказівник, який посилається на попередній блок. Після запису даних у блок їх неможливо змінити, оскільки будь-які зміни у будь-яких записах будуть помітні через зміну хешу цих записів. Заголовок блоку містить мітку часу, корінь дерева Меркла, версію блоку та хеш блоку, який йому передує. Оскільки кожен блок у блокчейні пов'язаний із хешем попереднього блоку, хеші з заголовку роблять транзакцію незмінною. Хеш заголовків всіх попередніх блоків буде змінено, якщо будь-який блок у мережі буде змінено або видалено.

Хешування – це процес перетворення вхідних даних будь-якої довжини у рядок попередньо визначеної довжини. Ethereum, а також технологія невзаємозамінних токенів використовують алгоритм SHA-3, також відомий як Keccak256. Для використання у блокчейні, хеш обов'язково має бути криптографічним, тобто задовільняти наступні параметри: детермінованість (той самий вхід дає ідентичний хеш), нерозбірливість (неможливо знайти вхідні дані будь-яким методом, окрім перевірки усіх можливих входів), захист від колізій (неможливість або дуже низька ймовірність отримати однаковий хеш для двох різних входів), ефект лавини (будь-яка зміна вхідних даних кардинально змінює вихід), швидкість генерації хешу.

Алгоритм консенсусу – це процедура досягнення згоди між різними вузлами мережі блокчейну. Дані алгоритми дозволяють досягати довіри та надійності між невідомими партнерами у розподіленому середовищі. Протокол консенсусу блокчейну забезпечує участь кожного з вузлів блокчейну у процесі консенсусу для досягнення згоди. Це гарантує надання рівних прав кожному вузлу мережі та дозволяє знайти спільну згоду між усіма вузлами.

Серед типів блокчейну можна виділити три різні типи: публічний, приватний та консорціумний.

Публічний блокчейн це відкрита мережа, до якої може приєднатися будь-хто. Кожен, хто хоче долучитися до мережі, повинен завантажити відповідне програмне забезпечення та запускати вузол самостійно. Прикладами таких мереж є Bitcoin та Ethereum.

Приватний блокчейн є мережею, у якій усі дозволи зберігаються централізовано. Кожен учасник мережі має можливість перегляду транзакцій у даному блокчейні, але лише певні члени мережі мають право перевіряти та вносити транзакції у блок. Прикладом приватного блокчейну є Hyperledger.

Консорцімний блокчейн це також приватний блокчейн, проте ним керує група осіб або товариство. Лише попередньо визначені вузли мають доступ до запису даних або блоку. Прикладом блокчейну такого типу є Web Foundation.

2.1.2 Смарт-контракти

Смарт-контракт – це комп’ютеризований протокол транзакції, код, який детерміновано виконується у блокчейн-мережі і цим забезпечує виконання контракту. Завдяки автоматизації умов угоди між сторонами, розумні контракти сприяють довірі та прозорості у децентралізованій системі.

Смарт-контракти можуть бути виконані лише при умові виконання певних критеріїв, цим самим прибираючи потребу посередництва та знижуючи ризики шахрайства. Це дозволяє використовувати їх під час управління ланцюгом постання та фінансових послуг.

Використання блокчейн-технологій робить смарт-контракти більш безпечними, оскільки зменшує ймовірність внесення змін в них. На даний час є велика кількість блокчейн-проектів, які підтримують даний тип контрактів, зокрема Ethereum, Binance Smart Chain, Solana, Cardano, Polygon, Avalanche, Polkadot та Cosmos.

Концепція смарт-контрактів була створена американським інформатиком Ніком Сабо у 1994 році. У своїй роботі він представив ідею використання цифрового коду для виконання умов договору, а також запропонував укладати такі контракти щодо синтетичних активів, зокрема деривативів та облігацій. Однак тоді дана ідея не отримала широкої популярності. Проте з появою технології блокчейн у 2008 році смарт-контракти отримали своє поширення. Першою платформою смарт-контрактів на основі блокчейну став Ethereum, запущений у 2015 році. Розумні-контракти на даній платформі були створені на мові програмування Solidity. Після цього, смарт-контракти отримали широке визнання як інструмент автоматизації транзакцій та зменшення потреби у посередниках у різних галузях.

На даний час параметри для виконання смарт-контракту мають бути конкретними та об'єктивними, тобто якщо умова «X» задовільняється – виконується крок «Y». Таким чином, завдання, які виконують смарт-контракти, є доволі елементарними, наприклад, автоматичне переміщення певної кількості криптовалюти з одного гаманця в інший, якщо задовільняються певні вимоги. У міру поширення блокчейну та токенізації все більшої кількості активів, смарт-контракти ставатимуть складнішими та здатними до обробки більш складних транзакцій. Зараз розробники об'єднують декілька кроків транзакцій для створення складнішних контрактів. Проте розумні контракти поки що не здатні визначати більш суб'єктивні юридичні критерії, зокрема чи задовільнила сторона стандарти комерційних відносин, чи має бути ініційований протокол відшкодування.

Перед тим, як скомпільований смарт-контракт може бути виконаним у блокчейнах, потрібно сплатити комісію за контракт, який буде додано до блокчейну та виконано. Говорячи про блокчейн Ethereum, смарт-контракти виконуються на віртуальній машині Ethereum (EVM) і сплата комісії відома як «gas fee». Розмір даної комісії залежить від складності смарт-контрактів, яка залежить від кроків транзакції, які потрібно виконати. Таким чином, цей податок

діє як шлюз для запобігання надто складним або численним смарт-контрактам, які можуть призвести до перевантаження EVM [22].

У сучасному світі смарт-контракти найкраще підходять для автоматичного виконання двох типів транзакцій, які часто зустрічаються у контрактах:

- забезпечення виплати коштів при виконанні початкових умов;
- накладення фінансових санкцій, якщо певні умови не виконуються.

У кожному випадку потреба втручання людини, у тому числі і через довірену особу або судову систем, є непотрібним після того, як смарт-контракт буде розгорнуто, тим самим зменшуючи витрати на виконання контрактактного процесу.

Етапи роботи смарт-контрактів можна розбити на наступні пункти:

- угода – досягнення сторонами консенсусу щодо умов контракту;
- створення – після узгодження умов відбувається їх кодування у смарт-контракт;
- розгортання – додавання створеного контракту у мережу блокчейн, що передбачає сплату комісії за виконання контракту;
- виконання – активний у блокчейні розумний-контракт може виконуватись автоматично при виконанні певних умов, які запрограмовані у контракті та викликаються конкретними подіями чи діями;
- верифікація – під час виконання розумних контрактів кожен крок перевіряється вузлами у мережі блокчейн, що гарантує правильне виконання контракту та дотримання умов;
- завершення – після виконання смарт-контракту контракт виконується автоматично, а його результат, наприклад переказ коштів, реєструється у блокчейні.

Смарт-контракти здатні усунути очікування між придбанням та оплатою. Для прикладу, товар надходить та сканується на складі, розумний-контракт може одразу зініціювати запити на необхідні схвалення та переказувати кошти від

покупця до продавця після отримання необхідних дозволів, що може суттєво спростити фінансові операції для обох сторін. Також смарт-контракт може бути запрограмованим на відключення доступу до підключеного до Інтернету активу, якщо платіж не був отриманим.

2.2 Стандарти токенів на блокчейні Ethereum

Стандарти токенів являють собою набір правил та специфікацій, які визначають як певний тип токена, наприклад невзаємозамінний токен, має створюватися та передаватися у блокчейн-мережі.

У контексті NFT було створено кілька стандартів для забезпечення взаємодії та сумісності між різними платформами і застосунками для невзаємозамінних токенів. Серед найпопулярніших стандартів NFT є ERC-20, ERC-721 та ERC-1155. Дані стандарти забезпечують розробникам та користувачам загальну структуру створення, керування та обміну невзаємозамінними токенами у різних блокчейн-мережах. Дотримання даних стандартів дозволяє легко створювати та передавати NFT без складних налаштувань та модифікацій.

Стандарти ERC-721 та ERC-1155 підтримують кросчейнові транзакції типу Burn-and-Mint. Під час проведення даної транзакції смарт-контракт «спалює» прив’язані активи в одному блокчейні та створювати аналогічну кількість таких активів в іншій мережі, не зачіпаючи активи, які заблоковані у вихідному блокчейні. Крім цього, дані стандарти підтримують функцію перевірки KYC (Know Your Customer), що дозволяє ідентифікувати та перевіряти користувачів і цим самим боротись з відмиванням коштів, фінансового тероризму, а також протидії фінансовим злочинам. Такі перевірки підвищують рівень довіри до програм, проте дана технологія критикується через нівелювання основних переваг криптовалюти – анонімність та децентралізованість.

2.2.1 ERC-20

ERC-20 є одним з найпопулярніших стандартів для токенів у мережі Ethereum. Дана специфікація визначає набір правил, яких повинен дотримуватись будь-який токен, що використовує даний стандарт, щоб забезпечити сумісність з іншими програмами та службами на основі Ethereum [23]. Ключові особливості стандарту токенів ERC-20:

- загальна кількість токенів ERC-20 є фіксованою та зазделегідь визначеною, що жодні токени не можна буде карбувати або знищувати після досягнення даного ліміту;
- токени ERC-20 можна розділити на менші одиниці, які називаються десятковими, що забезпечує гнучкість у вартості токенів та полегшує їх використання у щоденних транзакціях;
- право власності на токени закріплюється за адресами Ethereum, які контролюються приватними ключами;
- передача токенів здійснюється за допомогою відповідної фукнції передачі, яка приймає адресу одержувача, кількість передаваних токенів та перевіряє чи має відправник достатню кількість токенів для переказу і після здійснення транзакції оновлює баланси користувачів;
- витрачання токенів ERC-20 іншою адресою може бути схвалено відповідною функцією, що зазвичай використовується у децентралізованих біржах, де користувачі повинні надати дозвіл на біржі витрачати їхні токени від їх імені;
- токени можуть створювати мітки, коли відбуваються передачі або затвердження, які можуть використовуватись смарт-контрактами для запуску додаткових дій.

Даний стандарт дозволив створити активну екосистему додатків децентралізованого фінансування (DeFi), де користувачі можуть торгувати та позичати токени централізовано та без потреби у довірі. Недоліком стандарту

ERC-20 можна виділити те, що їх можна використовувати лише для представлення взаємозамінних токенів, що не дозволяє розподіляти спеціальні функції власності. Це призвело до розробки іншого стандарту токенів – ERC-721, який в свою чергу є незамінним.

2.2.2 ERC-721

Стандарт ERC-721 створений для представлення незамінних токенів у блокчейні Ethereum, який дозволяє створювати унікальні цифрові активи, які можна легко перевірити. Даний стандарт визначає набір правил та інтерфейсів, яких мають дотримуватись NFT для сумісності з іншими програмами та платформами. Також ERC-721 надає стандартний API для смарт-контрактів NFT. Ключові особливості стандарту:

- передача власності на NFT закріплено за адресом Ethereum та передається за допомогою функції передачі, відмінної від аналогічної функції у ERC-20;
- токени ERC-721 є унікальними та не можуть бути змінені як взаємозамінні токени, що гарантує їх автентичність та дефіцитність;
- кожен NFT містить у собі додаткові метадані, такі як назва, опис, зображення, або інші дані які описують актив;
- NFT не має можливості поділу на менші одиниці;
- токени стандарту ERC-721 можна використовувати у різних додатках та платформах, які також підтримують даний стандарт, що забезпечує просту інтеграцію та можливість передачі активів [24].

Стандарт ERC-721 набув популярності серед різних платформ, включаючи мистецькі біржі, ігрові платформи та майданчики віртуальної нерухомості. Це дозволило створювати та передавати унікальні цифрові активи безпечним, стандартизованим та сумісним методом, відкриваючи нові можливості використання токенів у мережі блокчейн.

2.2.3 ERC-1155

У 2018 році на блокчейні Ethereum було випущено новий мультитокеновий стандарт під назвою ERC-1155. Даний стандарт є розширенням попереднього стандарту ERC-721 та дозволяє створювати як взаємозамінні, так і невзаємозамінні токени в одному контракті.

На відміну від стандарту ERC-721, де кожен токен є унікальним та має власний смарт-контракт, стандарт ERC-1155 дозволяє створити єдиний смарт-контракт, який може містити необмежену кількість токенів, як взаємозамінних, так і унікальних. Це дозволяє створювати та керувати різними токенами більш ефективно та менш витратно.

Стандарт ERC-1155 визначає набір функцій, які повинні бути реалізованими смарт-контрактом для забезпечення сумісності та взаємодії з іншими контрактами та додатками у мережі Ethereum. Ці функції включають:

- balanceOf – дана функція повертає баланс певного токена, що належить певному адресу;
- balanceOfBatch – ця функція схожа за функціоналом до функції balanceOf, проте вона повертає баланс кількох токенів, що належать певному адресу;
- setApprovalForAll – функція дозволяє обліковому запису надати дозвіл іншому обліковому запису на передачу його токенів від його імені;
- isApprovedForAll – дана функція повертає інформацію про те, чи було надано обліковому запису дозвіл на передачу всіх токенів від імені іншого облікового запису;
- safeTransferFrom – ця функція дозволяє безпечно передавати токени з одного облікового запису на інший, запобігаючи втраті або крадіжці токенів;
- safeBatchTransferFrom – аналог функції safeTransferFrom, проте замість обробки передачі одного токена, дана функція дозволяє безпечну передачу одразу кількох токенів.

Можливість стандарту ERC-1155 обробляти пакетні транзакції, тобто передачу кількох токенів одночасно, дозволяє зменшити витрати та підвищити ефективність. Загалом, даний стандарт забезпечує більш гнучкий та ефективний спосіб створення та керування токенами у блокчейні Ethereum, що робить його оптимальним варіантом для розробників та користувачів.

2.2.4 BRC-721E

Стандарт BRC-721E – це новий стандарт токенів, що дозволяє користувачам передавати невзаємозамінні токени, створені на блокчейні Ethereum, на блокчейни мережі Bitcoin. Даний стандарт базується на протоколі порядкових номерів Bitcoin, який можна порівняти зі стандартом BRC-20 (стандарт взаємозамінних токенів на блокчейні Bitcoin), проте він створений для NFT, для яких стандарт BRC-20 не може бути прийнятий. Використовуючи порядкові номери біткоїнів, стандарт BRC-721E діє як міст між блокчейн-мережами Ethereum та Bitcoin.

Даний стандарт все ще перебуває у розробці, проте прогнозується, що його використання зможе підвищити популярність та швидкість впровадження порядкових номерів біткоїнів, що дозволить власникам NFT на основі Ethereum записувати свої токени та вписувати їх у мережу Bitcoin

2.3 Ризики безпеки невзаємозамінних токенів

2.3.1 Вразливості смарт-контрактів

Як зазначалось раніше, смарт-контракт це код, який детерміновано виконується у блокчейн-мережі та забезпечує виконання угоди. З їх допомогою проводяться транзакції з великими об’ємами важливих даних та активів, таких як перекази коштів, доступ до захищеного контенту, тощо, що робить їх ціллю зловмисників.

Різні типи атак на смарт-контракти та їх масштаби, визначають важливість безпеки смарт-контрактів для нових рішень блокчейну та Web3. Найважливішим є те, що вразливості смарт-контрактів не можуть бути вирішені після публікації смарт-контракту у мережі блокчейн, оскільки вносити зміни у контракт заборонено. Нижче подано опис найпопулярніших вразливостей смарт-контрактів.

Одною з найбільш відомих вразливостей смарт-контрактів можна вважати маніпуляції з оракулом. Оракул це постачальник інформації, який за запитом контракту надає з реального світу достовірні дані, потрібні для коректної роботи контракту. Основними вимогами до оракула є достовірність та повнота даних. Однак зловмисники можуть маніпулювати оракулами для досягнення особистих інтересів. Підроблені або неточні дані від оракула можуть дозволити виконання смарт-контракту. Найпоширенішим прикладом використання таких вразливостей є атаки на флеш-кредити. Флеш-позики дозволяють користувачам позичати будь-яку суму криптовалюти без обмежень при умові що позику повертають в одній транзакції. Хакери можуть використовувати такі кредити для спотворення цін на активи та отримання прибутку без шкоди для принципів технології блокчейн.

Наступною вразливістю смарт-контрактів варто визначити атаку повторного входу. Дана атаки стає можливою через особливості виконання у мові Solidity: кожен рядок коду має виконатися до того, як почнеться наступний. Таким чином, виконання контракту, який звертається до зовнішнього смарт-контракту, можна призупинити до повернення виклику, що дозволить викликаному контракту отримати контроль над подальшими діями. Зокрема, зловмисник може здійснити рекурсивний виклик до оригінального контракту, щоб вилучити ресурси, не чекаючи завершення першого виклику, тому оригінальному контракту не дозволяється оновлювати свій баланс до завершення функції. Існують різні форми атак повторного входу, включаючи однофункціональні, міжфункціональні, міжконтрактні атаки та атаки повторного входу лише для читання.

Смарт-контракти стають загальнодоступними одразу після їх публікування у мережі блокчейн, що дозволяє хакерам виконувати атаки на порядок виконання транзакцій, тобто обирати транзакції з найвищими комісіями, адже величина комісії впливає на шанс того, що транзакція буде схвалена майнером перед іншими. Дана особливість дозволяє зловмисникам підбирати коли вони можуть проводити прибуткові угоди подаючи ідентичний контракт, проте зі збільшеною комісією, щоб їх транзакція обробилась першою. Оскільки даний тип атак вимагає швидкості виконання, вони зазвичай виконуються ботами.

Атаки типу «примусового підживлення» використовують неможливість запобігання отриманню смарт-контрактами криптовалюти Ether. Дана особливість дозволяє переносити криптовалюту на будь-який контракт, примусово підживлюючи його для зміни утримуваного балансу, щоб маніпулювати логікою, яка покладається виключно на очікуваний баланс. Для прикладу, виплата винагороди, якщо баланс досягає певного рівня.

Наступною вразливістю є залежність від міток часу. Дані мітки генеруються вузлом, який виконує смарт-контракт, але через розподіленість мережі Ethereum, практично неможливо гарантувати правильну синхронізацію часу на кожному вузлі. Проте вузол може маніпулювати значеннями цих міток, які можна використати для створення логічних атак проти смарт-котракту, який покладається на мітку часу для виконання критичних за часом операцій

Вразливості цілочисельного недоповнення та переповнення виникають, коли результат арифметичної операції виходить за межі можливого діапазону значень – від 0 до 255 у випадку цілочисельного типу uint8. Значення, вищі за 255 скидаються до 0, а нижчі за 0 піднімаються до 255. Дана поведінка спричиняє несподівані зміни стану та логіки контракту, що може призвести до неправильних операцій у смарт-контракті.

Ще одною вразливістю смарт-контрактів є винекнення проблем з комісіями. Як було сказано раніше, для виконання смарт-контракту необхідна певна комісія, відома як «gas», яка визначається в залежності від попиту, пропозиції та потужності мережі на момент укладання угоди. Проблеми можуть виникнути

якщо користувач надішле необхідну комісію для виконання певного смарт-контракту, але недостатню для виконання субконтрактів. Також існують обмеження комісій у блоці. Якщо для виконання певної транзакції вимагається більше, ніж встановлений у блоці ліміт, транзакція не буде виконана. Даний тип вразливостей поширений у контрактах, які містять цикли, кількість ітерацій яких є нефіксованою та може зростати.

2.3.2 Вразливості блокчейн-мереж

Безпека блокчейн-мереж вимагає все більшої уваги, оскільки дана технологія набуває все більшої популярності. За оцінками аналітиків, у 2025 році блокчейн-ринки зростуть до 40 мільярдів доларів, а випадки використання будуть збільшуватись. Проте, як і будь-яка технологія, блокчейн має певні вразливості, вирішення яких є ключовим у контексті використання даної технології у цілях безпеки транзакцій, обміну інформацією та іншого.

Першим типом атак, які можуть бути здійснені на мережу блокчейн є атаки на маршрутизацію. Дані атаки використовують вразливості мережевих протоколів та механізмів маршрутизації для впливу на функціонування блокчейн-мережі. Для здійснення даної атаки зловмисник перехоплює консенсусні запити між вузлами та ізолює частину вузлів від решти мережі, що дозволяє йому маніпулювати транзакціями та записами у блокчейні. Такі атаки можуть спричини втрату децентралізації, конфіденційних даних та затримати транзакції. До атак на маршрутизацію належать атаки затемнення, атаки на сегментацію мережі, атаки на протокол BGP та атаки для зміни часу на вузлі [29].

Також варто виокремити атаку Sybil, при якій зловмисник створює велику кількість підроблених вузлів з метою непропорційного впливу на мережу. Це дозволяє зловмиснику маніпулювати консенсусом, ізольовувати частини мережі або виконувати атаки 51%, якщо підконтрольні вузли складатимуть мінімум 51% від усіх вузлів мережі. При цій атаці хакер матиме можливість передавати

криптовалюту іншим користувачам, після чого оголошувати дані транзакції недійсними та повертати собі кошти за послуги або товари.

Окрім атак на мережу, зловмисники можуть використовувати вразливості блокчейн-протоколів, щоб отримати контроль над проведенням транзакцій. Найбільш серйозні атаки та методи їх виконання наведені нижче.

Одною з найсерйозніших атак є атаки типу «Long-range», під час якої зловмисник створює ще одне розгалуження блокчейну і коли воно стає довшим за справжній блокчейн, хакер отримує можливість додавати власні транзакції.

Наступною серйозною вразливістю є «Race attack». Дана атака базується на принципі подвійної витрати. Зловмисник надсилає жертві непідтверджену транзакцію, також паралельно надсилає ще одну транзакцію з аналогічною кількістю коштів. Жертва отримує повідомлення про першу транзакцію, то ж з високою ймовірністю вважатиме що платіж завершено. Проте, оскільки решта блокчейн-мережі побачила подвійну витрату на початку, то існує висока ймовірність , що транзакція буде анульована.

Атака типу «vector76», названа на честь користувача, який вперше описав її принцип, є ще одною варіацією атак подвійної витрати. Під час даної атаки, зловмисник генерує блок з транзакцією, яка витрачає його монети на його адрес, але не транслює його у мережу. Паралельно з цим, зловмисник створює ще одну транзакцію, витрачаючи ті ж самі монети, і транслює її у мережу для оплати товару або послуги. Після підтвердження другої транзакції продавцем, хакер транслює попередньо створений блок, який замінює транзакцію продавця, оскільки він має вищий пріоритет через величину PoW. У результаті транзакція продавця анулюється, а зловмисник зберігає свої монети та отримує товар.

Ще одною критичною вразливістю блокчейн-мережі є цензурованість. Зараз цензура застосовується у відношенні до певних адрес, пов’язаних із організаціями або користувачами з метою дотримання законодавства. Проте зловмисники можуть отримати контроль над валідаторами блокчейну, що дозволить їм не додавати певні транзакції в блок через особисті або несправедливі причини, що може затримати або зупинити роботу мережі.

Також серйозною є атака попереднього обчислення, також відома як «grinding attack», яка впливає на системи PoS. Використовуючи відсутність випадковості у процесі вибору лідера слота, лідер може маніпулювати частотою їх обрання у наступних блоках.

Цілісність блокчейну також може бути порушена через криптографічні атаки. До даних атак можна віднести атаки грубої сили, частотного аналізу, атаки з відомим відкритим текстом, колізійні атаки. Також варто відмітити атаки типу передбачення приватного ключа, які проводяться методом впливу на генератори випадкових чисел, що дозволяє хакеру зламати шифр.

2.3.3 Фішингові атаки

Фішинг є поширеною стратегією, яку використовують зловмисники у цифровому середовищі. При даних атаках часто використовується автоматизація, націлена на велику кількість користувачів, що дозволяє досягти результатів з мінімальними зусиллями. Даний підхід також працює у відношенні до власників невзаємозамінних токенів. Зловмисники можуть поширювати шкідливі посилання, що ведуть на шкідливі сайти. Методами соціальної інженерії жертва може бути змушена завантажити шкідливе програмне забезпечення, яке дозволить хакеру отримати контроль над машиною користувача [31].

Яскравими прикладом успішної фішингової атаки є інцидент з колекціонером Тоддом Крамером, який ставши жертвою фішингової атаки втратив 15 NFT з колекцій «Bored Ape Yacht Club» та «Mutant Ape Yacht Club», які оцінювались в більш ніж 2 мільйони доларів. Користувач звернувся до маркетплейсу OpenSea з даною проблемою, що дозволило йому зберегти його токени. Важливу роль у поверненні токенів відіграла їх вартість та поширення у новинах, що не дозволило біржі проігнорувати його запит.

Також хакери можуть створювати точні копії відомих маркетплейсів, поширюючи посилання на них за допомогою електронної пошти або соціальних

мереж, щоб отримати повний доступ до акаунтів користувачів. Прикладом такого вебсайту є підроблений сайт проект «Pixelmon NFT». Оригінальний проект зібрав велику кількість користувачів завдяки багатообіцяючій стратегії розвитку, яка включає розробку метавсесвіту з можливістю утримання віртуальних домашніх тварин Pixelmon. Підробний веб-сайт заманював користувачів можливістю отримати доступ до ранньої версії гри, проте замість цього користувачі скачували архів, який містив PowerShell скрипт, який дозволяв розгортати стілер паролів Vidar на комп'ютері жертви.

2.3.4 Вразливості NFT-маркетплейсів

Вразливості NFT-маркетплейсів швидко виправляються, проте користувач не може бути точно впевненим, що біржа є абсолютно надійною.

Для прикладу, у 2022 році було виявлено вразливість маркетплейсу Rarible, яка дозволяла зловмиснику виводити усі активи користувачів. Для цього хакер розміщував NFT зі шкідливим кодом, надсилаючи жертві посилання на нього (також жертвою міг стати будь-який користувач, що випадково натиснув на токен при перегляді торгового майданчика). Шкідливий скрипт містив надсилання запиту setApprovalForAll, який у випадку підтвердження корисутвачем, надавав зловмиснику повний доступ до своїх активів. Виявлення даної вразливості трапилось після проведення атаки з викраденням NFT «Bored Ape» сумою на 500 тисяч доларів. Ще одна вразливість платформи Rarible була пов'язана з API OpenSea, яка дозволяла користувачам купляти невзаємозамінні токени за зниженою ціною, після маніпуляцій з неповним знищенням токенів на торговому майданчику.

Біржа Binance, створивши власний NFT-маркет, була атакована з використанням вразливості двофакторної автентифікації. Для реалізації даної атаки, зловмисник використав методи соціальної інженерії для зараження комп'ютера жертви вірусом REDLINE, який дозволив хакеру викрасти сесію авторизації користувача та використовуючи вразливість обходу двофакторної

авторизації, отримав доступ до акаунту користувача. Після цього з даного акаунту зловмисник купив власний NFT за суму коштів, яка була доступна у жертви, та вивів їх з біржі.

2.3.5 Вразливості гаманців

Криптогаманці вразливі до хакерських атак через баги у додатках та помилок корисутвачів. Перш за все, зловмисники можуть отримати ключі використовуючи уже згадані методи фішингу. Крім цього, користувачі можуть використовувати ненадійні генератори онлайн-гаманців, які можуть видавати однакові ключі різним користувачам або містити вразливості для перехоплення даних ключів [32].

Через помилки у генераторі випадкових чисел гаманець може бути створений лише з кількома випадковими байтами, для прикладу 0000000000000000000000000000000000000000aaf2bfaaa0000000000000000.
Хакери моніторять такі гаманці, перевіряючи їх баланси, тому як тільки користувач перерахує валюту на такий гаманець – зловмисники матимуть можливість її викрасти.

Використання методу braiwallet, який генерує приватний ключ на основі певної фрази, а не числа, дозволяє не запам’ятовувати послідовність випадкових символів, а запам’ятати ключову фразу гаманця. Проте користувачі не завжди використовують складні та унікальні фрази. Зазвичай ключі генеруються з одного слова, телефонних номерів або цитат, що дозволяє зловмисникам створювати ключі з популярних фраз, слів або ж злитих паролів.

Також існує можливість знайти ключ до гаманця у публічному доступі, зокрема на GitHub або GitLab. Розробники зазвичай тестують свої розробки з власними гаманцями і можуть випадково залишити їх у коді під час публікації. Хакери автоматично моніторять відкриті репозиторії у пошуках рядків які місять слова з сід-словника або співпадають з довжиною закритого ключа.

2.4 Висновки другого розділу

У другому розділі кваліфікаційної роботи було розглянуто технології, використані у невзаємозамінних токенах, зокрема технологія смарт-контрактів та блокчейн-мереж. Також були розглянуті стандарти блокчейну Ethereum ERC-20, ERC-721 та ERC-1155, які дозволяють створювати та керувати невзаємозамінними токенами, а також стандарт BRC-721E для взаємодії токенів з блокчейном Bitcoin.

Також було проаналізовано ризики та пов’язані з ними вразливості невзаємозамінних токенів. Ключові ризики NFT технології пов’язані з вразливостями смарт-контрактів, зокрема маніпуляції з оракулами, атаки повторного входу, порушення порядку виконання транзакцій, залежність від часових міток, арифметичні переповнення та обмеження комісій. Ще одним ключовим фактором є вразливості блокчейн-мереж, зокрема атаки на маршрутизацію, маніпуляція консенсусом, атаки з подвійною витратою та неприйняття певних транзакцій через особисті переконання, тобто цензурованість. Експлуатація даних вразливоствостей може дозволити зловмиснику викрадати кошти жертви, впливати на транзакції для отримання власної вигоди, зокрема блокуючи її, що є неприйнятним у концепції децентралізованих систем, зокрема NFT-технології. Не менш важливими є вразливості NFT-маркетплейсів, гаманців та фішингові атаки, що також може призвести до втрати коштів користувача.

Вирішення або пом’якшення даних ризиків є важливим етапом на шляху до створення надійного та безпечного середовища для використання NFT технології.

РОЗДІЛ 3 МЕТОДИ ВИРІШЕННЯ РИЗИКІВ БЕЗПЕКИ NFT

3.1 Методи вирішення ризиків смарт-контрактів

Більшість ризиків пов’язаних із вразливостями смарт-контрактів можуть бути виявлені під час розробки контракту завдяки якісному аудиту, що дозволяє точно визначити проблеми та виправити їх перед розгортанням смарт-контракту у блокчейні. Аудит може бути проведений внутрішньо, а також делегований стороннім компаніям, для прикладу Trail of Bits та Zeppelin Solutions, що обслуговують проект смарт-контрактів MakerDAO.

Протокол аудиту смарт-контрактів є важливою складовою у процесі пошуку вразливостей. Даний процес повинен включати наступні кроки:

- первинна оцінка – що включає визначення функціональних вимог до смарт-контракту;
- перевірка коду – дозволяє виявити помилки та невідповідності ще до початку аудиту, що забезпечить покращення коду та дозволить аудитору сконцетруватись на помилках, які не були виявлені під час даного аналізу;
- автоматизований аудит – перевірка за допомогою інструментів автоматизованої перевірки коду, що дозволяє виявити певні значні вразливості та заощадити час;
- ручне тестування – обмеження на глибину аудиту автоматизованих інструментів, зокрема неможливість врахування людського фактору та можливої нечіткої логіки використання смарт-контрактів, зумовлює потребу у ручній перевірці після автоматизованого аудиту;
- аналіз використання комісій (gas) – використання «газу» є ключовим для виконання смарт-контракту, тому перевірка його використання є важливим для визначення його вартості, ефективності та безпечності;
- перевірка відповідності до норм – важливим кроком є визначення того, що логіка смарт-контракту не порушує жодних існуючих норм та діє у рамках існуючої правової бази та її юрисдикції;

- огляд документації – технічна документація дозволяє аудитору оцінити чи виконує смарт-контракт поставлену мету та його роботу згідно запланованої логіки;

- створення звіту – після повної перевірки, порівняння результатів автоматизованої та ручної перевірок, аудитори повинні сформувати підсумковий звіт з переліком усіх проблем, вразливостей та проведених тестів [33];

Після внесення змін згідно рекомендацій аудитора, смарт-контракт може бути випущений.

Для автоматизованої перевірки використовуються спеціальні сканери вразливостей смарт-контрактів, наприклад Mythril або Oyente. Дані рішення схожі за функціоналом та дозволяють виявити найпоширеніші помилки у смарт-контрактах, зокрема залежність від міток часу, вразливості порядку виконання транзакцій та цілочисельного недоповнення та переповнення, можливість атак повторного входу. Дані програми перетворюють смарт-контракт у байт-код, використовуючи методи статичного аналізу та перевірки за допомогою випадкових даних (фаззинг). Недоліком даних інструментів тестування є неповна реалізація смарт-контрактів, що не дозволяє повністю перевірити усі можливі випадки виконання контракту.

Окрім статичного аналізу, смарт-контракти повинні бути перевіреними з використанням формальних методів перевірки, щоб надати докази того, що за допомогою прикладної математики логіка контракту є правильною. Це можна здійснити використовуючи засіб аналізу «Cetora Prover».

Інструмент формального аналізу смарт-контрактів «Certora Prover» містить засоби виявлення вразливостей та надає гарантії, що основні властивості системи завжди зберігаються. Дане рішення використовує схожу до Solidity мову специфікацій CVL, інструмент мутаційного тестування Gambit, а також засіб перевірки еквівалентності двох функцій. Для встановлення потрібних умов, специфікацій, використовується мову CVL, а для пошуку випадків порушення даних специфікацій використовуються алгоритми SMT.

Схема роботи засобу «Certora Prover»:

- Інструмент перетворює смарт-контракт у байт-код EVM;
- Декомпілює його у внутрішнє представлення TAC (three-address code);
- Алгоритм статичного аналізу виводить вірні інваріанти коду, спрощуючи перевірки;
- Після цього перетворює його у математичні формули та генерує умови перевірки;
- Інструмент шукає усі можливі способи поведінки, які порушують специфікацію;
- Генерує математичний доказ правильності роботи смарт-контракту.

Cetora Prover показує кращі результати у порівнянні з попередніми інструментами тестування за рахунок використання потужних специфікацій CVL та використання розв’язувачів SMT. Даний інструмент дозволяє виявити потенційно рідкісні шляхи виконання смарт-контрактів, що може допомогти ідентифікувати раніше невиявлені вразливості.

Проте окрім автоматизованої перевірки можна звернутись до ручної перевірки, під час якої можна виявити деталі, які були упущені під час автоматичної перевірки.

Виявлені під час аудиту вразливості смарт-контрактів вимагають правильного вирішення або ж мінімізації ризиків експлуатації даних вразливостей.

Однією з найбільш важливих вразливостей є можливі маніпуляції з оракулами, адже багато смарт-контрактів покладаються на надану інформацію для здійснення транзакцій, тому зміна даних від оракула несе загрозу фінансових втрат. Одним з найлегших методів вирішення даної проблеми є використання децентралізованих оракулів, наприклад Chainlink, API3 та Tellor, які надають дані з декількох незалежних джерел, що дозволяє знизити ризики маніпуляцій з даними. Також, інформація повернена оракулом повинна бути підписана, що гарантуватиме її достовірність та надійність джерела. Використання

усередненого значення від кількох оракулів забезпечує підвищену безпеку, оскільки це ускладнює атаку через складність та ціну одночасних атак на різні оракули. Це також гарантує отримання необхідних даних, навіть якщо одне зі звернень до оракула не вдалось. Додатковим рівнем безпеки може стати встановлення певних меж, які дозволяють обмежити допустимий діапазон зміни даних, що дозволяє виявити аномальні значення, які можуть сигналізувати про певні маніпуляції з даними. Також, ефективним механізмом захисту є використання механізму запізнення, наприклад TWAP – Time-Weight Average Price, що зменшує можливість короткочасних маніпуляцій. У складніших системах корисне використання механізмів голосування серед спільноти для визначення достовірності даних, що дозволить спільно приймати рішення у спірних ситуаціях.

Для захисту від атак повторного входу (reentrancy attack) потрібно використовувати шаблон «Перевірка-Ефект-Взаємодія» (Checks-Effects-Interactions), який рекомендує виконувати перевірки умов та оновлення стану перед викликом зовнішнього коду. Ще одним кроком до захисту від атак даного типу є використання модифікатора «nonReentrant» з бібліотеки OpenZeppelin, який блокує повторний вхід у функцію. Ще одним можливим рішенням може бути використання спеціальних змінних для позначення станів виконання функції, що дозволить блокувати їх виконання, якщо воно непотрібне, а також введення обмеження на кількість викликів для певної операції. Іншим варіантом захисту може стати використання функції «transfer» або «send» замість «call». Перевагою даних методів є зменшення ризику виконання зовнішнього коду, а також однакова ціна «газу», проте даний метод може не працювати в умовах, коли складніші контракти вимагаються більше «газу». Дієвою практикою може стати розподілення логіки контрактів на декілька різних контрактів. Наприклад, один контракт оброблятиме виконання транзакцій, в той час як інший керуватиме балансами. Це ускладнить атаку, оскільки хакеру потрібно знайти декілька вразливих точок.

Наступною можливою вразливістю атаки на порядок транзакцій, які дозволяють зловмисникам впливати на пріоритет виконання потрібних транзакцій для отримання фінансової вигоди. Одним з можливих методів захисту є механізм Commit-Reveal, який передбачає надсилання зашифрованого або хешованого запиту, проте сама інформація розкривається після прийому усіх змін, що знижує можливість зловмисників дізнатись зміст запиту зазделегідь. Ще одним варіантом захисту є використання протоколу Flashbots, який дозволяє надсилати транзакції безпосередньо до майнерів, оминаючи публічний мемпул. Завдяки цьому транзакції є недоступними для аналізу сторонніми учасниками мережі. Наступним способом є використання приватних мемпулів та валідаторів, які знижують ризики доступу інших учасників мережі до транзакцій. Ще одним можливим рішенням є використання MEV-протекційних платформ, наприклад CowSwap та Balancer, які забезпечують мінімальний вплив Maximum Executable Value цим самим оптимізуючи порядок виконання операцій.

Атаки примусового підживлення, як згадувалось раніше, можливі при маніпуляціях з балансами користувачів. Для уникнення такої атаки не потрібно використовувати баланс контракту для перевірки, оскільки фактичний баланс ефіру може бути вищим за очікуваний внутрішнім кодом контракту. Для слідкування за балансом контракту краще використовувати властивості самого смарт-контракту, записуючи значення балансів в окремих змінних.

Вразливість залежності від часових міток у смарт-контрактах може дозволити зловмисникам отримати короткочасний контроль над виконанням контракту. Існує два ключових методи захисту від даної вразливості. Перший це уникнення використання мітки часу для критичних функцій. Як варіант, часові мітки можна замінити на непрямий метод оцінки часу, наприклад методом використання номеру блоку та його порівняння з середньою кількістю блоків за добу. Проте для некритичних функцій, таких як записи подій, використання часової мітки є прийнятним. Другим методом захисту від даної вразливості є встановлення дозволеного діапазону похибок, наприклад 900 секунд, що дозволяє зменшити вплив коливань мітки часу на логіку смарт-контракту.

Наступною розглянутою вразливістю була можливість цілочисельного переповнення та недоповнення. Найбільш поширеною практикою вирішення даної проблеми є використання бібліотеки SafeMath [34], яка правильно обробляє арифметичні операції та дозволяє уникнути атак пов’язаних з цілочисельним переповнення та недоповненням. Дана бібліотека пропонується OpenZeppelin у своєму репозиторії [35]. З виходом нової версії мови Solidity – 8.0 – розробники додали вбудовані засоби перевірки арифметичних помилок. Проте якщо смарт-контракт розроблений на старішій версії Solidity, він вимагатиме використання бібліотеки SafeMath, що робить це рішення більш гнучким, оскільки підтримується різними версіями компілятора. Ще однією рекомендацією для боротьби з атаками переповнення та недоповнення є комплексна перевірка та валідація даних. Це вимагає перевірки змінних та вхідних даних, які пов’язані з арифметичними операціями. Також важливо переконатись у дійсності введених значень і відповідності вимогам контракту.

Останньою розглянутою вразливістю були проблеми з комісіями. Першим кроком для захисту від даного типу загроз є проведення аналізу кількості комісії. Точна перевірка потреб газу для кожної функції контракту, яку можна здійснити за допомогою спеціальних інструментів, може запобігти недооцінці або переоцінці. Перевірка успішності додаткових викликів та зовнішніх комунікацій гарантуватиме правильну роботу контракту. У випадку помилок потрібно застосовувати механізми їх обробок, щоб скасувати транзакції, якщо це потрібно. Також, одним з методів вирішення даної вразливості є використання засобів захисту від повторного входу, які були згадані раніше, щоб забезпечити правильний порядок операцій у смарт-контракті.

3.2 Методи вирішення вразливостей у блокчейн-мережах

Вразливості блокчейн-мережі можуть призвести до серйозних наслідків, зокрема втрату децентралізації, наслідком якої може бути втрата конфіденційних

даних, довіри до мережі та фінансові втрати через втрату цінності мережі, а також через викрадення коштів зловмисниками.

Вразливості блокчейну на рівні маршрутизації є доволі небезпечними, адже вони можуть призвести до більш серйозної атаки 51%, при якій мережу можна вважати повністю скомпрометованою, що несе за собою фінансові втрати. Першою серйозною атакою на захист від якої потрібно звернути увагу є атака затемнення (eclipse attack), під час якої зловмисник отримує контроль над одним з вузлів у мережі. Збільшення кількості вузлів з'єднаних з іншими вузлами ускладнює ізоляцію окремого хоста у мережі, тим самим зменшуючи можливість проведення даної атаки. Ще одним підходом до захисту є можливість блокчейн-мережі випадково вибирати вузли для синхронізації з мережею, а також використання чорних списків для шкідливих вузлів з можливістю їх динамічної зміни.

Наступною вразливістю на мережевому рівні блокчейн-систем є зміна маршрутів BGP, яка дозволяє зловмиснику перенаправляти трафік на підконтрольні вузли. Впровадження RPKI, resource public key infrastructure, може захистити від даного типу атак, оскільки технологія дозволяє перевіряти істинність маршрутів. Також для підвищення рівня безпеки можна використовувати статичні маршрути, що запобігає можливим змінам шляхів трафіку. Не менш важливим є моніторинг маршрутизації у реальному часі за допомогою спеціальних сервісів, зокрема BGPMon.

Серед загальних рекомендацій щодо захисту від мережевих атак на блокчейн-системи варто виділити наступні пункти: використання протоколів безпечної передачі даних, фаєрволів та проектування мережі з можливістю резервування.

Використання протоколів безпечної передачі даних, наприклад SSL/TLS, для шифрування даних, які передаються, що ускладнює перехоплення інформації, яка може бути використана для початку мережевої атаки;

Використання фаєрволів для фільтрації вхідного та вихідного трафіку між вузлами. Моніторинг трафіку допомагає запобігти надсиланню шкідливого

трафіку на цільовий вузол, а також допомагає запобігти поширенню шкідливого трафіку від зараженого вузла по мережі.

Проектування мережі з урахуванням можливості резервування, тобто надання однакової послуги різними вузлами, здатне покращити роботу мережі та уникнути відмов у випадку, якщо один з вузлів було скомпрометовано.

Окремим випадком атак на мережу блокчейн є атака Сивілли під час якої зловмисник отримує контроль не над одним хостом у мережі, а може контролювати велику їх кількість. Якщо дана кількість більша за половину кількості вузлів, атака Сивілли може перерости в атаку типу 51% при якій зловмисник може змінювати консенсус та транзакції. Захист від даної атаки включає у себе комплекс заходів, серед яких використання технології KYC (know your customer), яка дозволяє перевіряти ідентичність учасників мережі, що особливо важливо у приватних та консорціумних блокчейнах. Також важливим є впровадження обмежень на кількість підключень з однієї IP-адреси, а також системи репутації вузлів на основі попередньої роботи, що дозволить зменшити вплив новостворених вузлів на мережу до досягнення певного рівня репутації.

Зміна алгоритму консенсусу зі стандартного Proof of Work (PoW) на альтернативні та більш безпечні Proof of Stake (PoS) та Delegated Proof of Stake (DPoS) також можуть пом’якшити загрози від атак Сивілли, а також 51%. Алгоритм PoS, через особливості своєї роботи, робить економічно недоцільним контроль більшої частини мережі. Його покращена версія, Delegated Proof of Stake (DPoS), додає ще один рівень децентралізації безпеки, дозволяючи зацікавленим сторонам самостійно обирати делегатів для перевірки транзакцій та захисту блокчейну. Це значно ускладнює атаки, оскільки у мережі задіяно багато сторін. Також, даний алгоритм можна поєднувати з можливістю вибору творців блоків для більш безпечної та ефективної обробки транзакцій, як це реалізовано у блокчейн-мережі EOS.

Наступним типом атак є атаки на протокол блокчейну. Одним з підвидів даних атак є атаки подвійної витрати. Однією з найважливіших атак даного виду є атака перегонів (race attack), яка використовує затримку підтвердження

транзакцій для витрат тієї самої криптовалюти двічі, створюючи ідентичну транзакцію, яка згодом скасує попередню. Для уникнення даної атаки продавці повинні вимагати кількох підтверджень перед остаточним прийняттям транзакції, щоб ускладнити її скасування. Також важливим є безперевний моніторинг мережі на предмет спроб подвійного витрачання та впровадження розширених протоколів безпеки та механізмів виявлення шахрайства для виявлення та запобігання даним атакам, наприклад використання додаткових кроків перевірки.

Наступною атакою подвійної витрати є vector76, яка може бути застосована не лише до блокчейну Ethereum, але і Bitcoin та інших блокчейн-мереж. Дана атака користується вразливістю в протоколі консенсусу, створюючи дві з однаковими витратами транзакції, проте на відміну від проведення їх паралельно, як у випадку з атакою перегонів, зловмисник відправляє одну транзакцію в локальну мережу, а другу в офіційний блокчейн, після чого ініціалізує злиття мереж і викликає конфлікт подвійної витрати. Для захисту від даної атаки пропонується використання аналізу блоків та транзакцій на предмет конфліктів, підозрілої поведінки та аномалій за допомогою автоматизованого програмного забезпечення, а також з використанням машинного навчання. Використання систем які не приймають транзакції з одноразовим підтвердженням є дієвим методом у захисті від даної атаки. Натомість найменша кількість очікуваних підтверджень варіюється від 2 до 6. Ще одним кроком є моніторинг або заборона вхідних та вихідних з'єднань вузла до невідомих хостів, що запобігає впровадженню зловмисником невірної інформації про блокчейн або отримання інформації про стан блокчейну, який обробляється.

Окремим типом атак є грінд-атаки (grinding attack) під час яких зловмисник впливає на генерацію наступних блоків послідовності, створюючи найбільш вигідний для себе блок. Використання протоколу «ouroboros» може пом'якшити дану атаку завдяки реалізації безпечного протоколу підкидання монети, який генерує випадкове значення для вибору лідерів слотів, що не дає можливості жодному учаснику вплинути на результат. Крім цього, «ouroboros» використовує

функцію затримки, відому як «follow-the-satoshi», у якій результат виборів автора наступного блоку є невідомим до кінця поточного слоту. Це зменшує можливість зловмисників обчислити або передбачити, хто отримає наступний слот, тим самим запобігаючи грінд-атакам.

Не менш важливою проблемою блокчейн-мережі є можливість встановлення цензури на прийняття транзакцій. Дана проблема виникає у блокчейн-мережах де права учасників не розподілені рівномірно і можна ідентифікувати учасників транзакцій. Для унеможливлення такої поведінки валідаторів пропонується використання міксерів транзакцій, які унеможливлюють визначення учасників, впровадження винагород за включення усіх транзакцій та покарань у випадку блокування певних транзакцій. Також можливим методом вирішення даної проблеми є покращення децентралізації для уникнення концентрації влади окремих груп валідаторів, а також моніторинг мережі з метою виявлення вузлів та валідаторів, які цензурують транзакції.

Покращити безпеку блокчейн-мережі можна також за рахунок впровадження сучасних методів криптографії. Використання протоколу доказу з нульовим розголошенням дає можливість одній стороні довести іншій факт, що дана інформація істина, не передаючи жодної інформації , окрім факту, що інформація справді правильна.

У майбутньому проблемою стане використання квантової криптографії, оскільки квантові комп’ютери здатні зламувати шифри RSA, а також теоретично шифри еліптичної криптографії, які захищають інформацію у блокчейні. Розробка квантово-стійких алгоритмів захисту повинна бути досліджена у майбутньому, оскільки загроза квантових розрахунків зростає з кожним роком. На даний момент дослідники уже працюють над розробкою постквантових криптографічних алгоритмів, а також блокчейн-мережею Quantum Resistant Ledger, яка є невразливою до атак з використанням квантових комп’ютерів. Ще одним можливим майбутнім покращенням безпеки блокчейн-мереж та смарт-контрактів є використання алгоритму багатостороннього обчислення (Multi-

Party Computation), який дозволяє різним сторонам досягти згоди щодо стандартного виведення, який узгоджується з приватними вхідними даними.

3.3 Підвищення безпеки транзакцій

Для підвищення безпеки транзакції та зменшення проведення транзакцій у взаємодії зі шкідливими адресами можилве використання смарт-контрактів зі зверненням до зовнішніх джерел інформації, оракулів, для отримання оцінки рівня ризику транзакції. Для впровадження даної концепції використовується децентралізований оракул Chainlink, який дозволяє уникнути ризиків пов'язаних з маніпуляціями оракулами. В свою чергу, оракул створює запити до сервісу «MistTrack» для отримання інформації про рівень ризику транзакції, а також рівень ризику адреси, залученої у транзакції.

Початкова ініціалізація змінних та конструктор контракту поданий у лістингу 3.1. У ньому встановлюється токен _link, адреса оракула (_oracle), ідентифікатор завдання (_jobId) і комісію (fee) для запитів до оракула Chainlink.

Лістинг 3.1 – Ініціалізація смарт-контракту

```
pragma solidity ^0.8.0;
import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
import "@openzeppelin/contracts/access/Ownable.sol";
import "@chainlink/contracts/src/v0.8/ChainlinkClient.sol";
contract SecureNFT is ERC721, Ownable, ChainlinkClient {
    using Chainlink for Chainlink.Request;
    uint256 public tokenCounter;
    // Mapping to store risk levels of addresses
    mapping(address => string) public addressRiskLevel;
    // Oracle-related state variables
    address public oracle;
    bytes32 public jobId;
    uint256 public fee;

    event OracleRequest(bytes32 indexed requestId, address indexed
queryAddress);
    event OracleResponse(bytes32 indexed requestId, address indexed
queryAddress, string riskLevel);

    constructor(address _link, address _oracle, bytes32 _jobId,
uint256 _fee) ERC721("SecureNFT", "SNFT") {
```

Продовження лістингу 3.1

```
        tokenCounter = 0;
        _setChainlinkToken(_link);
        oracle = _oracle;
        jobId = _jobId;
        fee = _fee;
    }
```

Наступною функцією є «requestRiskLevel», яка надсилає запит до оракула Chainlink з параметрами «coin» та «address», які визначають токен та адресу перевірки відповідно, для оцінки ризику певної адреси (queryAddress). Передача запиту до оракула відбувається за допомогою імпортованої функції «sendChainlinkRequestTo». Результатом даної функції є створення події «OracleRequest» з ідентифікатором запиту «requestId» та адресою для перевірки.

Лістинг 3.2 – Реалізація функції requestRiskLevel

```
function requestRiskLevel(address queryAddress) external onlyOwner
returns (bytes32) {
        require(oracle != address(0), "Oracle address not set");

        Chainlink.Request memory request = _buildChainlinkRequest(
            jobId,
            address(this),
            this.fulfillRiskLevel.selector
        );

        request.add("coin", "ETH");
        request.add("address", toAsciiString(queryAddress));

        bytes32    requestId    =    _sendChainlinkRequestTo(oracle,
request, fee);
        emit OracleRequest(requestId, queryAddress);
        return requestId;
    }
```

Для обробки відповіді від оракула використовується функція «fulfillRiskLevel». Дана функція викликається оракулом для передачі результату запиту, який у цьому випадку представлений рівнем ризику «riskLevel». Після отримання відповіді, отримане значення встановлюється для відповідного адресу та генерується подія «OracleResponse». Реалізація даної функції подана у лістингу 3.3.

Лістинг 3.3 – Реалізація функції fulfillRiskLevel

```
function fulfillRiskLevel(bytes32 _requestId, string memory
_riskLevel) public recordChainlinkFulfillment(_requestId) {
        address queryAddress =
parseAddressFromRequestId(_requestId);
        addressRiskLevel[queryAddress] = _riskLevel;
        emit OracleResponse(_requestId, queryAddress, _riskLevel);
    }
```

Згідно отриманих значень, які можуть бути наступними – Low, Moderate, High та Severe – смарт-контракт може дозволяти або відхиляти транзакцію токена в залежності від визначених умов. У функції «safeTransfer» відбувається перевірка значення рівня ризику і у випадку, якщо це значення є «Low» і відправник є власником токена та має дозвіл на його передачу, транзакція дозволяється та викликається відповідна функція «_transfer» для передачі токена. Реалізація функції подана на лістингу 3.4.

Лістинг 3.4 – Реалізація функції SafeTransfer

```
function safeTransfer(address from, address to, uint256 tokenId)
external {
        require(keccak256(abi.encodePacked(addressRiskLevel[to]))
== keccak256(abi.encodePacked("Low")), "Recipient's address risk
level not Low");
        require(_isApprovedOrOwner(msg.sender, tokenId), "Caller is
not token owner nor approved");
        _transfer(from, to, tokenId);
    }
```

Повний лістинг смарт-контракту поданий у додатку Б.

Також було проведено аналіз смарт-контракту на наявність вразливостей, що розглядались раніше, зокрема атаки повторного входу, атаки на порядок виконання транзакцій, атаки примусового підживлення, залежність від міток часу, цілочисельні недоповнення та переповнення, проблеми з комісіями. Розглянувши подані вразливості було виявлено що у контракті відсутні вразливості до атак повторного входу, оскільки у ньому не використовуються функції для взаємодії з іншими контрактами через передачу криптовалюти або

виклик зовнішніх контрактів після зміни стану. Також відсутня загроза атаки на порядок виконання транзакцій, оскільки контракт не впливає на порядок здійснення транзакцій, а лише перевіряє їх ризикованість. Залежність від мітки часу та можливості цілочисельних переповнень або недоповнень також неможливі через відсутність відповідних функцій, які виконувались би у певний час або з використанням функції «block.timestamp» та використання нову версію Solidity 0.8, яка захищена від переповнень, а також відсутні помилки в арифметичних перетвореннях, які могли б спричинити їх.

Даний смарт-контракт може бути вразливим до атак примусового підживлення, зокрема маніпуляцій з оракулом, а також мати проблеми з комісіями. Перші два варіанти можливі лише за умови, коли зловмисник повністю керує оракулом або може перехоплювати та змінювати дані від оракула, проте децентралізований оракул Chainlink є безпечним, що зменшує можливість такої атаки. Проблеми з комісіями можуть виникнути у випадку великої кількості запитів до оракула для перевірки ризикованості адреси, які вимагатимуть непередбачуваних комісій.

Після компіляції даного контракту необхідно розгорнути його у мережі. Одним з найпопулярніших інструментів для цього є фреймворк Hardhat. Для його встановлення та початкового налаштування необхідно виконати наступні команди, подані у лістингу 3.6.

Лістинг 3.6 – Встановлення та налаштування Hardhat

```
mkdir my-project-folder
cd my-project-folder
npm init -y
npm install --save-dev hardhat
npm install @openzeppelin/contracts
```

Після цього необхідно запустити проект Hardhat за використовуючи команду «npx hardhat» для створення базового проекту. Власний контракт з розишренням .sol необхідно зберегти у папці /contracts попередньо видаливши звідти файли-зразки.

Наступним кроком є зміна файлу hardhat.config.js для з’єднання з тестовою мережею. Для цього необхідно вказати власний приватний ключ та бажану мережу, наприклад Mumbai. Наступним кроком є створення сценарію для розгортання контракту «deploy.js» у папці /scripts, код якого поданий у лістингу 3.7. Останнім кроком з розгортання є запуск команди для розгортання смарт-контракту «npx hardhat run scripts/deploy.js –network mumbai». Після запуску даної команди важливо зберегти адресу розгорненого смарт-контракту для подальшої роботи з ним та інтеграції у різні інструменти.

Лістинг 3.7 – Скрипт для розгортання смарт-контракту

```
const main = async () => {
    const ContractFactory = await
hre.ethers.getContractFactory('SecureNFT');
    const linkAddress = "0x...";
    const oracleAddress = "0x...";
    const jobId = hre.ethers.utils.formatBytes32String("job_id");
    const fee = hre.ethers.utils.parseEther("0.1");
    const Contract = await ContractFactory.deploy(linkAddress,
oracleAddress, jobId, fee);
    await Contract.deployed();
    console.log("Contract deployed to:", Contract.address);
};

const runMain = async () => {
    try {
        await main();
        process.exit(0);
    } catch (error) {
        console.log("Error deploying contract:", error);
        process.exit(1);
    }
};
runMain();
```

Для повноцінної роботи смарт-контракту необхідно поєднати його безпосередньо з Chainlink. Для цього потрібно налаштувати зовнішній адаптер (External Adapter), який надсилатиме запити до MistTrack API та оброблятиме відповіді. Реалізація скрипта для обробки запиту подана у лістингу 3.8. Він використовує бібліотеку requests для надсилання HTTP-запитів, а також методи

request та jsonify з фреймворку Flask для отримання даних з вхідного POST-запиту та перетворення їх у JSON-об’єкт.

Лістинг 3.8 – Реалізація функції handle_request

```
app = Flask(__name__)
@app.route('/', methods=['POST'])
def handle_request():
    data = request.get_json()

    address = data.get("data", {}).get("address")
    if not address:
        return jsonify({
            "jobRunID": data.get("id"),
            "status": "errored",
            "error": "Address is missing"
        }), 400

api_url =
f"https://openapi.misttrack.io/v1/risk_score?coin=ETH&address={add
ress}"
    headers = {"Authorization": f"Bearer {MISTTRACK_API_KEY}"}

    try:
        response = requests.get(api_url, headers=headers)
        response_data = response.json()

        if response.status_code == 200 and
response_data.get("success"):
            risk_level = response_data["data"].get("risk_level",
"Unknown")
            return jsonify({
                "jobRunID": data.get("id"),
                "data": {"risk_level": risk_level},
                "result": risk_level,
                "statusCode": 200
            })
        else:
            return jsonify({
                "jobRunID": data.get("id"),
                "status": "errored",
                "error": response_data.get("msg", "Error from
MistTrack API")
            }), 500

    except Exception as e:
        return jsonify({
            "jobRunID": data.get("id"),
            "status": "errored",
            "error": str(e)
        }), 500
```

Перш за все потрібно запустити сервер локально або у хмарному середовищі AWS, Heroku, тощо. Наступним кроком потрібно додати External Adapter як зовнішній API ендпоінт та вказати URL для доступу до даного адаптера. Вузол Chainlink надсилатиме запит у форматі JSON, приклад якого подано у лістингу 3.6, передаючи необхідні для запиту дані (coin, address, txid), та отримує у відповідь значення risk_score відносно певної адреси. Після чого вузок Chainlink повертає відповідь смарт-контракту зі значенням ризику і дозволяє виконати функцію fulfillRiskLevel.

Лістинг 3.6 – Приклад запиту Chainlink до адаптера

```
{
    "id": "job_id",
    "data": {
        "coin": "ETH",
        "address": "0x123...",
        "txid": "0xabc..."
    }}
```

3.4 Підвищення безпеки під час користування NFT

Безпека гаманців є важливою складовою у створенні безпечної екосистеми з використання невзаємозамінних токенів. Підвищення безпеки NFT-гаманців може запобігти втратам активів користувачів через шахрайство або кібератаки.

Для забезпечення найвищого рівня безпеки рекомендується використання апаратних гаманців, наприклад Ledger або Tenzor, оскільки у даному випадку приватні ключі зберігаються на фізичному пристрої, що ускладнює можливість хакерів отримати до них доступ. Апаратні гаманці працюють в офлайн-режимі, що забезпечує захист від більшості типів атак, зокрема фішингу.

Якщо використання апаратних гаманців неможливе, тобто використовується програмний гаманець, можна покращити безпеку використовуючи мультипідпис для підтвердження транзакцій, який вимагає ухвалення рішення від кількох учасників, що зменшує ризики несанкціонованого доступу до активів. Ще одним варіантом підвищення рівня захисту програмних

гаманців від несанкціонованих дій є впровадження технології двофакторної автентифікації. Навіть якщо пароль буде скомпрометовано, зловмиснику все одно буде необхідний ще один фактор підтвердження, наприклад код з мобільного додатку, на кшталт Google Authenticator або Cisco Duo, або підтвердження через електронну пошту. Також важливим є регулярне оновлення програмного забезпечення гаманців для виправлення проблем безпеки та підтримки нових функцій безпеки.

Розподілення активів по кількох гаманацях може вберегти їх від викрадення у випадку якщо зловмисник отримає доступ до гаманця. Для прикладу, можна використовувати апаратний гаманець для зберігання більш цінних токенів, тоді як на програмному гаманці можна зберігати менш цінні активи для щоденних операцій. Електронні гаманці найкраще підходять для проведення транзакцій з вищим ризиком, тоді як апаратні гаманці краще використовувати для довгострокового зберігання цінних активів завдяки ізоляції від Інтернету. Обов'язковим є створення резервних копій ключів і фраз відновлення та збереження у безпечному місці. Для цього можна використати різні методи зберігання резервних копій, наприклад збереження інформації у мережі, на карті пам'яті або іншому фізичному носії, наприклад паперовому записнику або металевій пластині. Важливо періодично перевіряти доступність та роботу резервних копій, щоб бути впевненим у можливості відновлення доступу.

Під час вибору NFT-біржі варто звертати увагу не лише на комісії та функції платформи, а також розглянути питання безпеки. Торгові майданчики невзаємозамінних токенів розділяються на два типи – централізовані та децентралізовані. З точки зору безпеки, централізовані маркетплейси, такі як Binance NFT та Nifty Gateway, є мають вищий рівень захисту із жорсткими заходами з боку централізованого управління, в той час як безпека децентралізованих ринків, наприклад Rarible та Foundation, є змінною, оскільки залежить не лише від дій платформи, але і від рішень самих користувачів. При виборі NFT-біржі варто проаналізувати відгуки про обраний майданчик та його історію на випадки зламів або вразливостей. Наступним кроком може стати

аналіз безпекових функцій маркетплейсу, таких як захищеність сховища та механізм ідентифікації особистості.

Захист від фішингу є важливою складовою у захисті не лише своїх NFT активів, а також іншої особистої інформації. Для захисту від даного типу атак достатньо не переходити за підозрілими посиланнями або відкривати електронні листи від невідомих відправників. Додатковим рівнем захисту буде використання антивірусного програмного забезпечення з функцією веб-фільтрації та захисту пошти, яке заблокує або попередить користувача перед переходом на шкідливий сайт або відкриттям фішингового листа. Також важливим є обережність під час використання відкритих Wi-Fi мереж для проведення дій пов’язаних з гаманцями, адже зловмисники можуть перехопити конфіденційну інформацію користувача і використати її для маніпуляцій з активами.

3.5 Висновки до третього розділу

У третьому розділі кваліфікаційної роботи був розглянутий алгоритм аудиту смарт-контрактів, функціонал інструментів для автоматизованої статичної перевірки коду смарт-контракту, а також формальної перевірки за допомогою математичного моделювання. Запропоновані методи вирішення або зменшення ризиків та пов’язаних з ними вразливостей смарт-контрактів. Зокрема, маніпуляції з оракулом, атаки повторного входу, примусового підживлення, порушення порядку транзакцій, залежність від часових міток та цілочисельне переповнення та недоповнення. Також, було розглянуто методи підвищення безпеки блокчейн-мереж, запропоновані варіанти покращення захисту від атак на маршрутизацію, протокол консенсусу та блокчейну, зокрема атак подвійної витрати та їх підвидів, цензурованості блокчейн-мережі. Проаналізовано майбутні проблеми використання блокчейну у зв’язку зі зростанням загрози квантової криптографії.

Для підвищення безпеки транзакцій було запропоновано смарт-контракт мовою Solidity, який звертається до стороннього джерела інформації, оракула Chainlink, для отримання інформації про ризики транзакції. Оцінка ризику відбувається за допомогою звернення оракула до MistTrack API. Також представлений код реалізації надсилання та обробки запитів до API мовою Python.

Розроблено рекомендації щодо підвищення загальної безпеки під час використання невзаємозамінних токенів, зокрема вимоги до вибору NFT-гаманців, торгового майданчику та захисту від фішингових атак.

РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорона праці

Робота з невзаємозамінними токенами, їх створення, розробка смарт-контрактів, підтримка блокчейн-мереж вимагає роботи з комп'ютерною технікою, що може бути небезпечним у випадку невідповідного користування або нехтуванням правилами та вимогами техніки безпеки. В Україні дані питання регулюються відповідними нормативними актами, які регламентують вимоги до захисту та охорони здоров'я працівників під час роботи з комп'ютерами.

Згідно з наказом Міністерства соціальної політики України від 14 лютого 2018 року №207 "Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями" встановлені мінімальні вимоги безпеки під час роботи з комп'ютерною технікою. Дані вимоги розроблено на основі директиви 90/270/ЄЕС від 29 травня 1990 року про мінімальні вимоги безпеки та здоров'я при роботі з екранними пристроями [41].

Згідно даного наказу робочі місця працівників повинні забезпечувати достатній простір для зміни робочого положення та рухів під час роботи. Випромінювання від екранної техніки повинне бути зведене до гранично допустимого рівня для збереження здоров'я працівників. Крім цього, повинен бути мінімізований вплив на людину факторів довкілля - шуму, вібрації, забруднювачів, температури та інших чинників. Робоче місце працівника повинне відповідати ергономічним, антропологічним та психофізіологічним вимогам, а також характеру виконуваних робіт.

Згідно наказу, освітлення робочого місця працівника з екранною технікою має створювати відповідний контраст між екраном та навколишнім середовищем та відповідати вимогам ДСанПіН 3.3.2.007-98 [42]. У даному документі визначені гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин. Згідно нього, штучне

освітлення має здійснюватись системою загального рівномірного освітлення з яскравістю не більше 200 кд/м2. Освітленість робочого місця повинна бути 300-500 люксів та , недостача освітлення загальною системою може бути компенсована використанням місцевих світильників при умові що вони не створюють бліків на екрані. Такі світильники повинні мати відбивач із захистим кутом більше 40°. Відбита яскравість на екрані має бути ≤40 кд/м², а яскравість стелі – ≤200 кд/м². Для дотримання норм освітленості необхідно двічі на рік чистити світильники та змінювати перегорілі лампи.

Мікроклімат приміщень, де розміщені робочі місця працівників з екранними пристроями має підтримуватись на постійному рівні та відповідати вимогам Санітарних норм мікроклімату виробничих приміщень ДСН 3.3.6.042-99, затверджених постановою Головного державного санітарного лікаря України від 01 грудня 1999 року № 42 (далі - ДСН 3.3.6.042-99). Згідно цього документу, оптимальні умови мікроклімату є наступними:

- показники температури повітря у робочій зоні по висоті та по горизонталі, а також протягом робочої зміни не повинні виходити за межі нормованих температур для визначеної категорії робіт;

- температура внутрішніх поверхонь робочої зони (стіни, стеля та підлога), технологічного обладнання, наприклад екрани, зовнішніх поверхонь технологічного устаткування, огороджуючих конструкцій не повинна виходити більш ніж на 2°С за межі оптимальних величин температури;

- для робіт пов’язаних з нервово-емоційним напруженням в кабінетах, пультах і постах керування технологічними процесами, у залах обчислювальної техніки та інших приміщеннях повинні дотримуватись оптимальні умови мікроклімату (температура повітря повинна становити 22-24°С, відносна вологість повітря повинна становити 60-40%, швидкість руху повітря не перевищувати 0.1 м/с) [43].

Робоче місце також повинне бути облаштоване згідно вимог щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями. Так,

робоча поверхня повинна бути достатнього розміру та мати поверхню з низькою відбивною здатністю, допускаючи гнучкість під час розмішення екрану, клавіатури та іншого периферійного обладнання. Робоче крісло повинне бути стійким та дозволяти працівнику легко рухатись та змінювати робоче положення. Сидіння має регулюватись по висоті, спинка сидіння повинна регулюватись по висоті та по нахилу, також слід передбачати використання підніжки для працівників, яким вона необхідна для зручності.

Не менш важливим аспектом безпечної роботи є режим праці та відпочинку під час використання екранної техніки. Згідно ДСанПІН 3.3.2.007-98 вимагається впровадження перерв для відпочинку під час роботи з комп'ютерами. Важливо робити короткі перерви ще до появи ознак втоми або зниження продуктивності для збереження здоров'я працівника та його ефективності. Для інженерів програмного забезпечення, чия робота пов'язана з інтенсивною розумовою працею з високим рівнем напруження зору та нервово-емоційним навантаженням вимагається встановлювати 15-хвилинні перерви кожної години. Під час даних перерв рекомендується здійснювати вправи для зняття напруженості, поліпшення кровообігу та працездатності. У разі скарг на зорове перевантаження може бути обмежено час роботи з комп'ютерною технікою або змінено характер праці. Максимально допустима тривалість безперервної роботи повинна не перевищувати 4 години при умові неможливості дотримання регламентованих перерв. Для зменшення монотонності робочого процесу радиться чергувати введення даних та редагування текстів, а також при можливості переключатись на інші види діяльності. У разі високого рівня напруженості робіт рекомендується використання спеціальних кімнат для психологічного відпочинку. Також рекомендується виконання гімнастичних вправ для розслаблення, зняття втоми очей та покращення самопочуття у післяробочий час.

Працівники повинні проходити обов'язкові медичні огляди раз на два роки комісією у складі терапевта, невропатолога та офтальмолога, а також інших спеціалістів за необхідності. Основними показниками при роботі з

комп’ютерною технікою є показники органів зору, а також стан організму в цілому.

Дотримання даних вимог є важливим для безпеки та захисту здоров’я працівників, оскільки це сприяє підвищенню продуктивності та зменшенню ризику травматизації чи отримання професійних захворювань. Під час виконання даної кваліфікаційної роботи були враховані усі відповідні норм охорони праці під час роботи з екранними пристроями, а також вимоги до санітарних норм. Робоче місце було облаштовано з розрахунку на ергономіку та освітлення, згідно ДСанПіН 3.3.2.007-98. Мікроклімат у робочому приміщенні відповідав ДСН 3.3.6.042-99, що гарантувало оптимальні умови праці. Також, були дотримані вимоги до перерв під час роботи з комп’ютерною технікою.

4.2 Безпека в надзвичайних ситуаціях

Планування заходів цивільного захисту на об’єкті у випадку надзвичайних ситуацій – це сукупність документів у яких визначені сили і засоби, порядок і послідовність дій з метою забезпечення захисту населення, виробництва, а також виконання завдань вищих органів, пов'язаних із поданням допомоги населенню інших об'єктів і міст. Дані документи розроблені з урахуванням реальних можливостей та умов об’єкта, є настановою для організованих дій як з метою підготовки захисту об’єкта в надзвичайних умовах, так і з метою ліквідації наслідків надзвичайних ситуацій, зокрема стихійних лих, виробничих аварій, воєнних конфліктів.

Для об’єкту повинно бути розроблено два плани – на мирний та воєнний час. План цивільного захисту на мирний час визначає організацію та порядок виконання заходів цивільного захисту з метою запобігання або зменшення можливих втрат від важких виробничих аварій, катастроф та стихійних лих, а також ведення рятувальних та інших невідкладних робіт при їх виникненні. План цивільного захисту на воєнний час визначає організацію та порядок переведення

об’єкту з мирного на воєнний час та ведення цивільного захисту у початковий період війни.

Безпосередня робота щодо планування заходів цивільного захисту визначається розпорядчим документом (директива, наказ, розпорядження, організаційні вказівки), у якому має бути визначено основних розробників плану (посадові особи), посадовців, що готують пропозиції до проекту плану, основні організаційні питання з підготовки пропозицій до проекту плану, а також термін подання проекту плану на затвердження. Реальність розроблених планів ЦЗ буде залежати від повноти вихідних даних, наявності сил і засобів, правильного обліку всіх можливостей об'єкта. Плани ЦЗ об'єкта розробляють його керівники, спеціалісти і орган управління ЦЗ. Розробляючи заходи служб (формувань) ЦЗ об'єкта, ряд питань необхідно узгоджувати з відповідними районними службами ЦЗ, районним відділом з питань НС та цивільного захисту населення.

Пропозиції в проект плану повинні містити конкретні заходи, що мають забезпечити досягнення цілей, виконання завдань та ефективне використання ресурсів. Терміни розроблення, формування і подання пропозицій визначаються з урахуванням часу необхідного для їх розгляду та узгодження [44].

Планування реагування на надзвичайні ситуації (далі – НС) є однією з найважливіших функцій управління у сфері цивільного захисту, яке здійснюється на основі певних принципів.

До загальних принципів планування реагування на НС відносять:

- цільову направленість;
- системність;
- безперервність;
- збалансованість;
- оптимальність використання ресурсів;
- адекватність рівня загрози та заходів реагування.

Як вихідні документи, що будуть використані при розробці документів плану цивільного захисту об'єкта, необхідні: директивні документи Президента,

Верховної Ради, Уряду України та МНС; витяг із рішення керівника цивільного захисту району про організацію і ведення цивільного захисту на території району, дані про кількість формувань, їх особовий склад, які потрібно створити на даному об'єкті; витяг із плану прийому і розміщення евакуйованого населення; витяг із наряду райвійськкомату на постачання техніки у збройні сили у зв'язку з мобілізацією; окремі розпорядження керівника цивільного захисту району (наряд для виконання спеціальних завдань та ін.); документи, які характеризують господарство і населений пункт.

Розробка плану відбувається у три етапи в певній послідовності. Перший етап – підготовчий, протягом якого визначається склад виконавців і затвердження їх, підготовка виконавців до роботи, доведення до них директив, рекомендацій та інших документів, узагальнення й аналіз вихідних даних, необхідних для розробки плану ЦЗ, визначення обсягу робіт і розподіл обов'язків між виконавцями та закріплення відповідальних за розділами плану.

Для планування, підготовки і проведення заходів евакуації має бути інформація, щоб забезпечити відповіді на такі запитання: чисельність працюючих відвідувачів, обслуговуючого персоналу на даному об'єкті, всього населення в населеному пункті; час доби, коли буває найбільше скупчення людей у приміщеннях; розміщення людей у приміщеннях; стан входів, аварійних виходів; наявність і стан входів для пожежників, міліції, поліції, внесення технічних засобів; труднощі, які треба враховувати під час евакуації людей (вузькі проходи, сходи, непрацюючі ліфти та ін.); забезпеченість будівельними матеріалами, матеріалами для огороджування небезпечних місць, захищення аварійної або цінної апаратури; забезпеченість тимчасовими робочими місцями та ін.; устаткування, прилади, апаратура, документи, які необхідно евакуювати і перелік тих, що можна залишити; можливість переведення виробництва, переведення установ на скорочений режим роботи; вирішення питань зупинення виробництва, установ на скорочений режим роботи; вирішення питань зупинення роботи технологічних ліній, припинення чи скорочення виробництва продукції; забезпечення засобами індивідуального захисту, оповіщення і зв'язку;

підготовленість пунктів збору, транспорту для перевезення людей і цінностей, наявність поблизу загрозливих об'єктів (пожежо- і вибухонебезпечні будівлі й матеріали, столярні цехи, приміщення складів, комор, трансформаторні приміщення, хімічні підприємства чи склади).

Планування евакуації має передбачати виникнення найбільш несприятливих ситуацій під час підготовки і проведення евакуації: відсутність відповідних керівників, транспорту, електрозабезпечення, погані погодні умови, аварія на дорозі, паніка серед людей та ін.

Другий етап — практична розробка, оформлення документів. Заходи, які плануються в документах плану, мають бути спрямовані на виконання завдань ЦЗ в надзвичайних ситуаціях.

У документах плану визначають заходи, які потрібно виконати в мирний час, при загрозі виникнення надзвичайних ситуацій, несподіваному нападі противника, стихійних лихах, виробничих аваріях, катастрофах і при ліквідації наслідків надзвичайних ситуацій, проведенні рятувальних та інших невідкладних робіт, а також характер і порядок дій формувань, зміст і обсяг робіт, строки виконання заходів з урахуванням конкретних умов і можливостей даного об'єкта.

Заходи, які потребують капітальних затрат і матеріально-технічних засобів, також мають бути висвітлені в цих планах.

До них належать: будівництво протирадіаційних укриттів, пункту управління, забійних площадок і пунктів, площадок ветобробки сільськогосподарських тварин; придбання засобів для герметизації тваринницьких ферм, складських приміщень і колодязів; систем зв'язку і оповіщення; придбання майна для формувань, спеціальної техніки, необхідної формуванням для проведення рятувальних та інших невідкладних робіт, автономних джерел електроенергії.

Оскільки заходи потребують матеріальних затрат, вони повинні здійснюватися у комплексі з іншими економічними заходами, через що їх

необхідно включити в поточний і перспективний план об'єкта, де вони будуть забезпечені коштами.

За даними оцінки можливої обстановки, що може скластися на об'єкті, керівники об'єкта планують заходи підвищення стійкості роботи об'єкта. Всі пропозиції, пов'язані із затратами, необхідно документально обґрунтувати з поданням відповідних заявок із кошторисами в місцеві, районні, обласні органи управління ЦЗ, а якщо необхідно то у відповідні міністерства, відомства.

Фінансування капітальних вкладень на будівництво захисних споруд, складів, пункту управління та інших об'єктів ЦЗ відбувається за рахунок об'єкта з коштів, які виділяються міністерствами в межах загальних обсягів капітальних вкладень.

Планування таких заходів, як підготовка і забезпечення майном формувань, навчання керівного особового складу формувань, працюючих, організація зв'язку і оповіщення, створення навчально-матеріальної бази та ін., проводиться за рахунок коштів об'єкта.

Планування забезпечення особового складу формувань ЦЗ засобами індивідуального захисту та іншими матеріально-технічними засобами провадиться за нормами, затвердженими Начальником ЦЗ України, узгодженими з Кабінетом Міністрів, відповідним міністерством, відомством.

Третій етап — узгодження розроблених планів із відділом ЦЗ району, з районним агропромисловим управлінням, адміністрацією населеного пункту, службами ЦЗ району, після цього затвердження документів плану ЦЗ. Документи плану ЦЗ підписує керівник — ЦЗ об'єкта, деякі (план евакуації, прийому і розміщення евакуйованого) підписує і начальник ЦЗ голова адміністрації населеного пункту. Зміст плану ЦЗ об'єкту узгоджується з вимогами плану ЦЗ району, що підтверджує начальник відділу з питань цивільного захисту населення району, після чого план ЦЗ затверджує керівник ЦЗ об'єкта.

Після затвердження плану об'єкта організується вивчення документів усім керівним складом об'єкта. У зв'язку зі зміною вихідних даних, покладених в

основу розробки плану об'єкта, таких як: розвиток господарства, технічне забезпечення, кількість і структура населення, рівень розвитку ЦЗ, установлених вимог і завдань та ін., необхідно періодично уточнювати і переробляти розроблені раніше документи плану даного об'єкта. Уточнення і коригування документів плану проводять на тренуваннях і комплексних об'єктових навчаннях ЦЗ. Зміни і доповнення, які не мають принципових змін, вносять у документи після узгодження з керівництвом об'єкта. Доповнення і зміни принципового характеру узгоджують із відділом цивільного захисту населення району. Усі зміни записуються та засвідчуються підписом керівника ЦЗ та підтверджуються печаткою об’єкта.

ВИСНОВКИ

Робота присвячена аналізу ризиків NFT-технології та розробці методів з їх вирішення або пом’якшення, а також запропоновано смарт-контракт з функцією оцінки ризикованості проведення транзакції на основі даних від оракула, розроблено рекомендації щодо підвищення рівня безпеки під час користування невзаємозамінними токенами.

Під час її виконання було проведено аналіз технології невзаємозамінних токенів, зокрема розглянуто їх історію, процес створення та використання, сфери у яких використовується NFT, а також розглянуті найбільш популярні торгові майданчики для продажу та покупки невзаємозамінних токенів.

У другому розділі було розглянуто технології, використані у невзаємозамінних токенах – смарт-контракти та блокчейн мережі, стандарти NFT ERC-20, ERC-721, ERC-1155 та новий стандарт, який на даний час перебуває у розробці BRC-721E. Крім цього, було проведено аналіз ризиків невзаємозамінних токенів та пов’язаних з ними вразливостей. Зокрема розглянуті вразливості смарт-контрактів, блокчейн-мереж, фішингові атаки, вразливості маркетплейсів та вразливості гаманців.

Третій розділ присвячений пошуку методів вирішення або пом’якшення розглянутих раніше ризиків. Розглянутий алгоритм аудиту смарт-контрактів, інструменти для його проведення, а також запропоновані методи вирішення вразливостей смарт-контрактів. Крім цього висунуті методи покращення захисту блокчейн-мереж від поширених атак, зокрема атак на маршрутизацію, протокол консенсусу, використання подвійної витрати та цензурованості. Було розглянуто проблеми криптографічних алгоритмів та можливий подальший розвиток та використання новітніх шифрів, стійких до квантової криптографії.

У даному розділі також представлено реалізацію смарт-контракту для визначення ризикованості транзакції за допомогою інформації отриманої від оракула Chainlink та MistTrack API. Крім цього створені рекомендації щодо

підвищення безпеки для використання NFT, зокрема вибір безпечних гаманців, бірж, а також захист від фішингу.

Представлені рішення можуть бути використані користувачами невзаємозамінних токенів для підвищення власної безпеки у процесі їх використання. Крім цього розглянуті методи вирішення ризиків смарт-контрактів та блокчейн-мереж, а також впровадження смарт-контракту для оцінки ризику транзакції можуть бути використанні для підвищення безпеки технології невзаємозамінних токенів з технічної точки зору.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Зеров, К. (2023, 24 серпня). Цифрові речі, цифровий контент, NFT та авторське право: Співвідношення в межах розширеного кола об’єктів цивільних прав. Вища школа адвокатури НААУ. https://www.hsa.org.ua/blog/cifrovi-reci-cifrovii-kontent-nft-ta-avtorske-pravo-spivvidnosennia-v-mezax-rozsirenogo-kola-objektiv-civilnix-prav

2. Rosenfeld, M. (2012, 4 грудня). Overview of colored coins. Allquantor. https://allquantor.at/blockchainbib/pdf/rosenfeld2012overview.pdf

3. What is counterparty? | counterparty. (б. д.). Counterparty. https://docs.counterparty.io/docs/basics/what-is-counterparty/

4. Lypa, B., Horyn, I., Zagorodna, N., Tymoshchuk, D., Lechachenko T., (2024). Comparison of feature extraction tools for network traffic data. CEUR Workshop Proceedings, 3896, pp. 1-11.

5. Tymoshchuk, D., Yasniy, O., Mytnyk, M., Zagorodna, N., Tymoshchuk, V., (2024). Detection and classification of DDoS flooding attacks by machine learning methods. CEUR Workshop Proceedings, 3842, pp. 184 - 195.

6. Eltuhami, M., Abdullah, M., & Talip, B. A. (2022, листопад). Identity verification and document traceability in digital identity systems using non-transferable non-fungible tokens. IEEE Xplore. https://ieeexplore.ieee.org/document/10033362

7. Tymoshchuk, V., Mykhailovskyi, O., Dolinskyi, A., Orlovska, A., & Tymoshchuk, D. (2024). OPTIMISING IPS RULES FOR EFFECTIVE DETECTION OF MULTI-VECTOR DDOS ATTACKS. Матеріали конференцій МЦНД, (22.11. 2024; Біла Церква, Україна), 295-300.

8. Serrano, W. (2022, березень). Real estate tokenisation via non fungible tokens. ResearchGate. https://www.researchgate.net/publication/361827384_Real_Estate_Tokenisation_via_Non_Fungible_Tokens

9. Tymoshchuk, D., & Yatskiv, V. (2024). Slowloris ddos detection and prevention in real-time. Collection of scientific papers «ΛΌГОΣ», (August 16, 2024; Oxford, UK), 171-176.

10. Blockchain in healthcare: Transforming patient data management. (2024, 13 лютого). Antematter. https://antematter.io/blogs/blockchain-healthcare-patient-data-management

11. Bănulescu, E. (2023, 7 липня). Rarible review: What to know about the pros, cons, & features. MilkRoad. https://milkroad.com/reviews/rarible/

12. Tymoshchuk, V., Vorona, M., Dolinskyi, A., Shymanska, V., & Tymoshchuk, D. (2024). SECURITY ONION PLATFORM AS A TOOL FOR DETECTING AND ANALYSING CYBER THREATS. Collection of scientific papers «ΛΌГОΣ», (December 13, 2024; Zurich, Switzerland), 232-237.

13. Miller, D. (2023, 5 липня). SuperRare review 2024: What to know about the pros, cons, & features. MilkRoad. https://milkroad.com/reviews/superrare

14. ТИМОЩУК, Д., & ЯЦКІВ, В. (2024). USING HYPERVISORS TO CREATE A CYBER POLYGON. MEASURING AND COMPUTING DEVICES IN TECHNOLOGICAL PROCESSES, (3), 52-56.

15. Mintable: Overview, reviews, related apps & faqs. (2023, 18 липня). Ethereum Ecosystem - Discover Web3. https://www.ethereum-ecosystem.com/apps/mintable

16. Tymoshchuk, V., Vantsa, V., Karnaukhov, A., Orlovska, A., & Tymoshchuk, D. (2024). COMPARATIVE ANALYSIS OF INTRUSION DETECTION APPROACHES BASED ON SIGNATURES AND ANOMALIES. Матеріали конференцій МЦНД, (29.11. 2024; Житомир, Україна), 328-332.

17. Накамото, С. (2012). Біткоїн: Електронна пірингова система готівки. Bitcoin - Open source P2P money. https://bitcoin.org/files/bitcoin-paper/bitcoin_uk.pdf

18. Економічна правда. (2021, 14 квітня). Блокчейн і fintech: Як змінюється сфера фінансів. https://www.epravda.com.ua/columns/2021/04/14/672973/

19. Tymoshchuk, V., Pakhoda, V., Dolinskyi, A., Karnaukhov, A., & Tymoshchuk, D. (2024). MODELLING CYBER THREATS AND EVALUATING THE PERFORMANCE OF INTRUSION DETECTION SYSTEMS. Grail of Science, (46), 636–641. https://doi.org/10.36074/grail-of-science.29.11.2024.081

20. Li, X. J., Ma, M., & Yong, Y. X. (2021). A blockchain-based security scheme for vehicular ad hoc networks in smart cities. IEEE Xplore. https://ieeexplore.ieee.org/document/9707356

21. ТИМОЩУК, Д., ЯЦКІВ, В., ТИМОЩУК, В., & ЯЦКІВ, Н. (2024). INTERACTIVE CYBERSECURITY TRAINING SYSTEM BASED ON SIMULATION ENVIRONMENTS. MEASURING AND COMPUTING DEVICES IN TECHNOLOGICAL PROCESSES, (4), 215-220.

22. Garnett, A. G. (б. д.). Ethereum gas fees: The cost of doing (crypto) business. Britannica Money. https://www.britannica.com/money/ethereum-gas-fees-eth

23. ERC-20 token standard | ethereum.org. (2024, 11 липня). ethereum.org. https://ethereum.org/en/developers/docs/standards/tokens/erc-20/

24. ERC-721 non-fungible token standard | ethereum.org. (2023, 19 листопада). ethereum.org. https://ethereum.org/en/developers/docs/standards/tokens/erc-721/

25. ERC-1155 multi-token standard | ethereum.org. (2023, 15 серпня). ethereum.org. https://ethereum.org/en/developers/docs/standards/tokens/erc-1155/

26. Stanko, A., Wieczorek, W., Mykytyshyn, A., Holotenko, O., & Lechachenko, T. (2024). Realtime air quality management: Integrating IoT and Fog computing for effective urban monitoring. CITI, 2024, 2nd.

27. CoinMarketCap. (2021, 21 серпня). What is smart contract risk? | coinmarketcap. CoinMarketCap Academy. https://coinmarketcap.com/academy/article/what-is-smart-contract-risk

28. OWASP smart contract top 10 | OWASP foundation. (б. д.). OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation. https://owasp.org/www-project-smart-contract-top-10/

29. Yesin, V., Karpinski, M., Yesina, M., Vilihura, V., Kozak, R., & Shevchuk, R. (2023). Technique for Searching Data in a Cryptographically Protected SQL Database. Applied Sciences, 13(20), 11525.

30. Barragan, J., Jefferies, J., & Jevans, D. (б. д.). Top 10 blockchainattacks, vulnerabilities &weaknesses. Cognizium. https://cognizium.io/uploads/resources/Claud%20Security%20Alliance%20-%20Top%2010%20Blockchain%20Attacks%20Vulnerabilities%20and%20Weaknesses%20-%202021.pdf

31. Kshetri, N. (2022, 11 квітня). Scams, frauds, and crimes in the nonfungible token market. IEEE Xplore. https://ieeexplore.ieee.org/document/9755212

32. Orobchuk, O. (2019). Methodology of development and architecture of ontooriented system of electronic learning of Chinese image medicine on the basis of training management system. Вісник Тернопільського національного технічного університету, 92(4), 83-90.

33. Malanii, O., & Parisi, C. (2023, 22 травня). How to audit A smart contract: Hacken's process - hacken. Hacken. https://hacken.io/discover/smart-contract-audit-process/

34. Math - OpenZeppelin Docs. (б. д.). Documentation - OpenZeppelin Docs. https://docs.openzeppelin.com/contracts/2.x/api/math

35. Openzeppelin-solidity/contracts/math/SafeMath.sol at master · ConsenSysMesh/openzeppelin-solidity. (б. д.). GitHub. https://github.com/ConsenSysMesh/openzeppelin-solidity/blob/master/contracts/math/SafeMath.sol

36. T. Lechachenko, R. Kozak, Y. Skorenkyy, O. Kramar, O. Karelina. Cybersecurity Aspects of Smart Manufacturing Transition to Industry 5.0 Model. CEUR Workshop Proceedings, 2023, 3628, pp. 325–329

37. Kuznetsov, A., Karpinski, M., Ziubina, R., Kandiy, S., Frontoni E., Veselska, O., & Kozak, R. (2023). Generation of nonlinear substitutions by simulated annealing algorithm. Information, 14(5), art. no. 259, 1-16. doi: 10.3390/info14050259.

38. Nataliya Zagorodna, Iryna Kramar (2020). Economics, Business and Security: Review of Relations. Business Risk in Changing Dynamics of Global Village BRCDGV-2020: Monograph / Edited by Pradeep Kumar, Mahammad Sharif. India, Patna: Novelty & Co., Ashok Rajpath,. 446 p., pp.25-39. (дата звернення: 17.11.2024).

39. Kuznetsov, O., Poluyanenko, N., Frontoni, E., Kandiy, S., Karpinski, M., & Shevchuk, R. (2024). Enhancing Cryptographic Primitives through Dynamic Cost Function Optimization in Heuristic Search. Electronics, 13(10), 1-52. doi: 10.3390/electronics13101825 (дата звернення: 17.11.2024).

40. Boltov, Y., Skarga-Bandurova, I., & Derkach, M. (2023, September). A Comparative Analysis of Deep Learning-Based Object Detectors for Embedded Systems. In 20 23 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS) (Vol. 1, pp. 1156-1160). IEEE (дата звернення: 18.11.2024).

41. Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями, Наказ Міністерства соціальної політики України № 207 (2018) (Україна). https://zakon.rada.gov.ua/laws/show/z0508-18#Text (дата звернення: 10.12.2024).

42. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, Постанова № 7 (1998) (Україна). https://zakon.rada.gov.ua/rada/show/v0007282-98#Text (дата звернення: 10.12.2024).

43. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99, Постанова № 42 (1999) (Україна). https://zakon.rada.gov.ua/rada/show/va042282-99#Text (дата звернення: 10.12.2024).

44. Крюковська, О. (2018). Конспект лекцій з дисципліни “ Охорона праці в галузі та цивільний захист ”для здобувачів другого (магістерського) рівня спеціальностей 101 «Екологія»очної та заочної форм навчання. Дніпровський державний технічний університет. https://www.dstu.dp.ua/Portal/Data/5/10/5-10-kl3.pdf (дата звернення: 11.12.2024).

45. Методичні рекомендації щодо порядку складання планів реагування у разі загрози та виникнення надзвичайних ситуацій на підприємствах, установах та організаціях. (2015, 20 березня). https://www.schastye-rada.gov.ua/metodichn-rekomendats-shchodo-poryadku-skladannya-plan-v-reaguvannya-u-raz-zagrozi-ta-viniknennya-na (дата звернення: 11.12.2024).

Додаток А Публікація

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
імені ІВАНА ПУЛЮЯ
НАУКОВЕ ТОВАРИСТВО ім. ШЕВЧЕНКА

М А Т Е Р І А Л И

XII НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

18-19 грудня 2024 року

ТЕРНОПІЛЬ

2024

УДК 001
М34

ПРОГРАМНИЙ КОМІТЕТ

Голова: Приймак Микола – професор кафедри комп'ютерних систем та мереж, д.т.н., професо**р.**
Співголови: Марущак Павло – проректор з наукової роботи, докт. техн. наук, професор.
Баран Ігор – канд. техн. наук, доцент, декан факультету ФІС.
Науковий секретар: Надія Крива– старший викладач.

Члени: Василь Кривень - завідувач кафедри математичних методів в інженерії д.ф.-м.н., професор; Галина Осухівська – завідувач кафедри комп'ютерних систем та мереж, к.т.н., доцент; Микола Карпінський - професор кафедри кібербезпеки, д.т.н., професор; Жанна Баб'як - завідувач кафедри української та іноземних мов, к.пед. н., доцент; Ярослав Литвиненко – професор кафедри комп'ютерних наук, д.т.н., професор; Михайло Петрик - завідувач кафедри програмної іженерії, д.ф.-м.н., професор; Наталія Загородна – завідувач кафедри кібербезпеки, к.т.н., доцент.

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

Голова: Скоренький Юрій Любомирович – канд. техн. наук, доцент кафедри фізики.
Члени: доцент кафедри комп'ютерних наук, к.т.н. В. Никитюк; доцент кафедри програмної інженерії, к.т.н. Д. Михалик; доцент кафедри кібербезпеки, к.т.н. М. Стадник; доцент кафедри комп'ютерних систем та мереж, к.т.н. Є. Тиш; ст. викладач Л. Джиджора.

М34 Матеріали XII науково-технічної конфції «Інформаційні моделі, системи та технології» Тернопільського національного технічного університету імені Івана Пулюя, (Тернопіль, 18–19 грудня 2024 р.). – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2024.

Адреса оргкомітету: ТНТУ ім. І. Пулюя, м. Тернопіль, вул. Руська, 56, 46001,
тел. (0352) 52-41-33, факс (0352) 25-49-83.
E-mail: conffis2024@gmail.com
Редагування, оформлення та верстка: Надія Крива

СЕКЦІЇ КОНФЕРЕНЦІЇ, ЯКІ ПРЕДСТВЛЕНІ В ЗБІРНИКУ

- Математичне моделювання
- Інформаційні системи та технології, кібербезпека
- Комп'ютерні системи та мережі
- Програмна інженерія та моделювання складних розподілених систем
- Новітні фізико-технічні та освітні технології

В збірнику надруковано тези доповідей XII науково-технічної конференції «Інформаційні моделі, системи та технології» (Тернопіль, 18–19 грудня 2024 р.) за такими науковими напрямками: математичне моделювання; інформаційні системи та технології, кібербезпека; комп'ютерні системи та мережі; програмна інженерія та моделювання складних розподілених систем; новітні фізико-технічні та освітні технології.

Розрахований на науковців, викладачів та студентів вузів.

За зміст тез та дотримання норм академічної доброчесності відповідальність несе автор.

УДК 004.056:004.9
П.В. Пилипів
Тернопільський національний технічний університет імені Івана Пулюя, Україна

Ризики безпеки NFT-технології та методи їх вирішення

P.V.Pylypiv

Security Risks of NFT Technology and Methods for Addressing Them

NFT (невзаємозамінні токени) стрімко завоювали популярність у різних галузях — від цифрового мистецтва та ігор до фінансів і управління інтелектуальною власністю. Ключовими причинами цього є їх можливість надання підтвердження права власності та походження активу. NFT досі розглядаються як шлях до створення прозорої та підзвітної цифрової екосистеми. Проте разом із поширенням даної технології збільшилась кількість випадків шахрайства та фішингу, зломів гаманців, використання вразливостей смарт-контрактів та атак на мережу блокчейн. Згідно зі звітами, щорічні втрати через кіберзагрози у блокчейн-екосистемах сягають мільйонів доларів.

Проте, попри зростання популярності NFT, наразі немає усталених підходів для мінімізації ризиків. Це робить дослідження у цій сфері необхідними для подальшого розвитку екосистеми.

Аналіз ризиків технології NFT включає аналіз вразливостей смарт-контрактів, пов’язаних з ними атаки, атаки на мережу блокчейн, зокрема атаки маршрутизації, атаки 51%, атаки на протоколи блокчейну[1] та криптографічні атаки.

Мінімізація можливих ризиків невзаємозамінних токенів досягається підвищенням безпеки смарт-контрактів, зокрема їх аудитом, вирішенням вразливостей згідно OWASP Smart Contract Top 10 [2], а також впровадженням використання звернень до стороннього джерела інформації, відомого як оракул, для отримання інформації про транзакції. Дана реалізація здійснена за допомогою Chainlink API, яке дозволяє отримати рівень довіри до адреси та історію його попередніх транзакцій для подальшої оцінки ризиків транзакції.

Важливим аспектом пом’якшення ризиків є методи боротьби з поширеними атаками на блокчейн-мережі, зокрема атакою затемнення, атаками 51% та атаками пов’язаними з подвійною витратою. Також одною з основних сторін захисту є використання децентралізованих та мультифакторних рішень для підвищення безпеки гаманців та орієнтованих на безпеку платформ для створення та торгівлі NFT.

У підсумку, рішення повинні бути спрямовані на комплексний підхід до ідентифікації, аналізу та вирішення ризиків, забезпечуючи більш надійну та безпечну інфраструктуру для використання невзаємозамінних токенів, що є ключовим в умовах зростання популярності NFT у різних сферах життєдіяльності.

Література

1. Framework for Determining the Suitability of Blockchain: Criteria and Issues to Consider. URL: https://www.researchgate.net/publication/353073634_Framework_for_Determining_the_Suitability_of_Blockchain_Criteria_and_Issues_to_Consider

2. OWASP Smart Contract Top 10. URL: https://owasp.org/www-project-smart-contract-top-10/

Додаток Б Лістинг смарт-контракту для звернення до оракула

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
import "@openzeppelin/contracts/access/Ownable.sol";
import "@chainlink/contracts/src/v0.8/ChainlinkClient.sol";

contract SecureNFT is ERC721, Ownable, ChainlinkClient {
    using Chainlink for Chainlink.Request;

    uint256 public tokenCounter;

    // Mapping to store risk levels of addresses
    mapping(address => string) public addressRiskLevel;

    // Oracle-related state variables
    address public oracle;
    bytes32 public jobId;
    uint256 public fee;

    event OracleRequest(bytes32 indexed requestId, address indexed
queryAddress);
    event OracleResponse(bytes32 indexed requestId, address
indexed queryAddress, string riskLevel);

    constructor(address _link, address _oracle, bytes32 _jobId,
uint256 _fee) ERC721("SecureNFT", "SNFT") {
        tokenCounter = 0;
        _setChainlinkToken(_link);
        oracle = _oracle;
        jobId = _jobId;
        fee = _fee;
    }

    /**
     * @dev Request risk level for a specific address from the
oracle.
     */
    function requestRiskLevel(address queryAddress) external
onlyOwner returns (bytes32) {
        require(oracle != address(0), "Oracle address not set");

        Chainlink.Request memory request = _buildChainlinkRequest(
            jobId,
            address(this),
            this.fulfillRiskLevel.selector
        );

        request.add("coin", "ETH");
        request.add("address", toAsciiString(queryAddress));
```

```
        bytes32 requestId = _sendChainlinkRequestTo(oracle,
request, fee);
        emit OracleRequest(requestId, queryAddress);
        return requestId;
    }

    /**
     * @dev Callback function for the oracle to provide risk
level.
     */
    function fulfillRiskLevel(bytes32 _requestId, string memory
_riskLevel) public recordChainlinkFulfillment(_requestId) {
        address queryAddress =
parseAddressFromRequestId(_requestId);
        addressRiskLevel[queryAddress] = _riskLevel;

        emit OracleResponse(_requestId, queryAddress, _riskLevel);
    }

    /**
     * @dev Mint a new NFT, only for addresses with Low risk.
     */
    function safeMint(address to) external onlyOwner {
        require(keccak256(abi.encodePacked(addressRiskLevel[to]))
== keccak256(abi.encodePacked("Low")), "Address risk level not
Low");
        _safeMint(to, tokenCounter);
        tokenCounter++;
    }

    /**
     * @dev Transfer NFT only if the recipient's risk level is
Low.
     */
    function safeTransfer(address from, address to, uint256
tokenId) external {
        require(keccak256(abi.encodePacked(addressRiskLevel[to]))
== keccak256(abi.encodePacked("Low")), "Recipient's address risk
level not Low");
        require(_isApprovedOrOwner(msg.sender, tokenId), "Caller
is not token owner nor approved");
        _transfer(from, to, tokenId);
    }

    /**
     * @dev Burn an NFT (for cleanup or regulatory purposes).
     */
    function burn(uint256 tokenId) external onlyOwner {
        _burn(tokenId);
    }

    /**
```

```
     * @dev Helper function to convert address to string.
     */
    function toAsciiString(address x) internal pure returns
(string memory) {
        bytes memory s = new bytes(40);
        for (uint256 i = 0; i < 20; i++) {
            bytes1 b = bytes1(uint8(uint256(uint160(x)) / (2**(8 *
(19 - i)))));
            bytes1 hi = bytes1(uint8(b) / 16);
            bytes1 lo = bytes1(uint8(b) - 16 * uint8(hi));
            s[2 * i] = char(hi);
            s[2 * i + 1] = char(lo);
        }
        return string(s);
    }

    /**
     * @dev Helper function to parse address from request ID (mock
implementation).
     */
    function parseAddressFromRequestId(bytes32 _requestId)
internal pure returns (address) {
        // In a real implementation, store and retrieve the query
address associated with the request ID.
        return address(0);
    }

    /**
     * @dev Helper function to convert byte to ASCII char.
     */
    function char(bytes1 b) internal pure returns (bytes1 c) {
        if (uint8(b) < 10) return bytes1(uint8(b) + 0x30);
        else return bytes1(uint8(b) + 0x57);
    }
}
```