НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ ЗАХИСТУ ІНФОРМАЦІЇ

«На правах рукопису»
УДК _____

«До захисту допущено»
В.о. завідувача кафедрою
_____ М.М.Савчук
(підпис)
“___” _____ 2019 р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності: 113 «Прикладна математика»

на тему: «Методи забезпечення приватності транзакцій та користувачів блокчейнів»

Виконала: студентка 6 курсу, групи ФІ-73мн
Соловйова Марина Сергіївна
(підпис)

Керівник: професор, д.т.н., с.н.с. Кудін А. М.
(підпис)

Рецензент: доцент, к.т.н. Проскуровський Р. В.
(підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ – 2019 року

Національний технічний університет України
«Київський політехнічний інститут
імені Ігоря Сікорського»
Фізико-технічний інститут
Кафедра математичних методів захисту інформації

Рівень вищої освіти: другий (магістерський) за освітньо–професійною програмою

Спеціальність: 113 «Прикладна математика»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедрою
_____ М.М.Савчук
(підпис)
«___»_____ 2019 р.

ЗАВДАННЯ
на магістерську дисертацію студентки

Соловйової Марина Сергіївни

1. Тема дисертації: «Методи забезпечення приватності транзакцій та користувачів блокчейнів», науковий керівник дисертації: професор, д.т.н., с.н.с. Кудін А. М.,
затверджені наказом по університету від _____ р. № _____

2. Термін подання студентом дисертації _____

3. Об’єкт дослідження: технологія блокчейн, зокрема декілька його реалізацій - криптовалюти Біткойн, Монеро та технологія Hyperledger Sawtooth.

4. Предмет дослідження: алгоритми та підходи, що використовуються в технології блокчейн.

5. Перелік завдань, які потрібно розробити:

- аналіз існуючих рішень для забезпечення приватності даних користувачів блокчейну та їх складання звіту про них;
- синтез нових моделей компонент блокчейну для підвищення анонімності його клієнтів та приховання їх конфіденційної інформації;

- компонування порівняння характеристик існуючих та побудовани рішень, орієнтованих на збереження приватності даних в децентралізованих публічних мережах.

6. Орієнтовний перелік ілюстративного матеріалу:

- ілюстрації до структур і процесів в технологіях, розглянутих в даній роботі;
- перелік формул, що характеризують роботу проаналізованих протоколів;
- таблиця порівнянь властивостей обраних криптовалют, приватних блокчейнів та нових запропованих підходів.

7. Орієнтовний перелік публікацій відсутній.

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Ознайомлення з літературою на визначену область блокчейн технології, що буде розглянута в дослідницькій роботі	листопад 2017 р.	Виконаний
2	Конкретизація проблемних аспектів обраної області блокчейну	січень 2018 р.	Виконаний
	Визначення, який саме аспект буде досліджуватися, та підготовка звіту на цю тему	квітень 2018 р.	Виконаний
3	Формулювання теми магістерської дисертації	вересень 2018 р.	Виконаний
4	Постановка задач дослідницької роботи та переліку потенційних методів для їх виконання	листопад 2018 р.	Виконаний

5	Аналіз та опис декількох технологій та алгоритмів, що будуть використані для синтезу нових рішень, орієнтованих на удосконалення технології блокчейн. Формування відповідних звітностей	січень 2019 р.	Виконаний
6	Детальна формалізація нових підходів, опис їх загальної структури, потенційних обмежень, сфер використання та задіяних алгоритмів	березень 2019 р.	Виконаний
7	Складання переліку властивостей для порівняння існуючих та нових підходів, описаних в роботі. Компонування відповідних даних у вигляді таблиці	квітень 2019 р.	Виконаний

Студент _____ Соловйова М. С.
(підпис)

Науковий керівник дисертації _____ Кудін А. М.
(підпис)

РЕФЕРАТ

Роботу виконано на 82 аркушах, вона містить 1 додаток та перелік посилань на використані джерела з 43 найменувань. У роботі наведено 7 рисунків та 1 таблиця.

Наразі технологія блокчейн посідає провідне місце серед технічних рішень, що використовуються в найрізноманітніших сферах нашого життя, наприклад, при захисті авторських прав чи в електронних голосуваннях. Цей протокол побудований, щоб вирішувати прикладні задачі по розподіленим та децентралізованим довіреним обчисленням. Тому, метою даної дипломної роботи є аналіз блокчейну та окремих його аспектів – підходів та методів забезпечення приватності даних користувачів.

Об'єктом дослідження є технологія блокчейн, зокрема декілька його реалізацій – криптовалюти Біткойн, Монеро та технології Hyperledger Sawtooth.

Предметом дослідження є алгоритми та підходи, що використовуються в технології блокчейн.

БЛОКЧЕЙН, ПРИВАТНІСТЬ, БІТКОЙН, МОНЕРО, HYPERLEDGER SAWTOOTH

ABSTRACT

The thesis is presented in 82 pages. It contains 1 appendix and bibliography of 43 references. 7 figures and 1 table are given in the thesis.

Currently blockchain technology occupies a leading place among the technical solutions used in a wide range of areas of our lives, for example, in copyright protection or electronic voting. This protocol is designed to solve applied problems for distributed and decentralized trusted calculations. Therefore, the purpose of this thesis is to analyze the blockchain and its individual aspects – approaches and methods for ensuring the privacy of user data.

The object is a blockchain technology, including several of its implementations – Bitcoin, Monero cryptocurrencies and Hyperledger Sawtooth technology.

The subject is set of the algorithms and approaches used in the blockchain technology.

BLOCKCHAIN, PRIVACY, BITCOIN, MONERO, HYPERLEDGER SAWTOOTH

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

I2P — Invisible Internet Project

ECDHM — Elliptic-Curve Diffie–Hellman-Merkle

BTC — Bitcoin

GUI — Graphical User Interface

TEE — Trusted Execution Environment

SGX — Software Guard Extensions

DKSAP — Dual-Key Stealth Address Protocol

SDK — Software Development Kit

P2P — Peer-to-Peer network

ACID — Atomicity, Consistency, Isolation and Durability

ECDSA — Elliptic Curve Digital Signature Algorithm

PoET — Proof of Elapsed Time consensus protocol

PoW — Proof of Work consensus protocol

BFT — Byzantine Fault Tolerance

AML — Anti-Money Laundering

KYC — Know Your Customer concept

ВСТУП

Актуальність дослідження. Сучасна технологія блокчейн розвивається дуже швидко та має чимало модифікацій і версій. Проте в ній наявні деякі суттєві недоліки, що так і не знайшли однозначного рішення, наприклад, стосовно протоколу консенсусу та приватності даних в мережі. Саме тому дана тема є дуже актуальною для вивчення.

Метою дослідження є аналіз блокчейну та окремих його аспектів – підходів і методів забезпечення приватності даних користувачів.

Для досягнення мети необхідно розв'язати **задачу дослідження** синтезу нових моделей компонент блокчейну для підвищення анонімності його користувачів та приховання конфіденційної інформації. Зокрема, потрібно:

– дослідити існуючі рішення з точки зору цього аспекту, переваги та недоліки їх поточної роботи;

– визначити випадки, коли технологія блокчейн наразі не в змозі виконати вимоги користувачів через незадовільну роботу його механізмів забезпечення приватності;

– проаналізувати різні алгоритми та технології, створені захищати конфіденційність даних та їх власників в умовах розподілених децентралізованих систем;

– запропонувати власні підходи забезпечення приватності користувачів та порівняти їх з існуючими реалізаціями в технології блокчейн.

Об'єктом дослідження є протокол блокчейн, зокрема декілька його реалізацій – криптовалюти Біткойн, Монеро та технологія Hyperledger Sawtooth.

Предметом дослідження є алгоритми та підходи, що використовуються в технології блокчейн.

При розв'язанні поставлених завдань використовувались такі *методи*

дослідження, як аналіз, порівняння основних характеристик предметів дослідження та синтез моделей нових підходів, що функціонують на основі блокчейну та створені для покращення цього протоколу.

Наукова новизна отриманих результатів полягає у винайдені нових підходів збереження конфіденційності даних клієнтів в різних реалізаціях технології блокчейн шляхом додавання до базового протоколу компонент та механізмів інших децентралізованих систем, що володіють інакшим набором властивостей.

Практичне значення результатів полягає у покращенні забезпечення приватності даних користувачів та їх анонімності в технологіях на основі блокчейну, підвищення рівня децентралізації та удосконалення інших характеристик даних систем. Як наслідок, збільшиться коло застосування блокчейну та знайдуться нові форми і сфери для його реалізації.

1 ПОНЯТТЯ КОНФІДЕНЦІЙНОСТІ ДАНИХ В ПРОТОКОЛІ БЛОКЧЕЙН

Немає сумніву, що технологія блокчейн має величезний потенціал. Децентралізовані біржі, ринки прогнозування та платформи для управління активами – все це лише частина з існуючих цікавих додатків, які вивчаються блокчейн-розробниками.

Але попри загальновідомі переваги, блокчейн має кілька основних технічних бар'єрів, які роблять його непрактичними для застосування. Перерахуємо основні з них:

– обмежена масштабованість;
– обмежена конфіденційність;
– відсутність формальної перевірки смарт-контрактів;
– обмеження зберігання даних;
– нестійкі механізми консенсусу;
– відсутність засобів управління та стандартів;
– недостатнє оснащення;
– квантова обчислювальна загроза, тощо.

У цьому розділі розглянемо питання приватності в блокчейні більш детально. Це допоможе сформувати перелік випадків, коли дана технологія не може бути використана через недоліки її реалізації щодо захисту даних клієнтів мережі.

1.1 Загальна проблематика та існуючі рішення

Враховуючи, що транзакції в блокчейні безпосередньо не прив'язані до користувача, вони, на перший погляд, можуть здаватися достатньо

приватними. Будь-хто має можливість анонімно створити новий електронний гаманець та з його допомогою здійснювати транзакції. Однак варто приділити цьому процесу та його реалізації трохи більше уваги, щоб мати ширше уявлення про нього.

З одного боку, безумовно вірно, що велика перевага технології блокчейн – анонімність. Транзакції записуються та зберігаються в публічній книзі обліку, а також пов'язані з адресою облікового запису, що складається виключно з цифр і букв. Якщо до цієї адреси не додано жодної ідентичності, визначити відправника транзакції здається неможливим. Проте подібне визначення повної безпеки може бути досить оманливим. Насправді, користувач зберігає свою приватність допоки його псевдонім не пов'язаний з його ідентичністю, але як тільки хтось виявить цей зв'язок – таємниця викривається. Один приклад такого явища був оприлюднений, коли правоохоронні органи визнали, що під час одного з своїх розслідувань[1] були в змозі ідентифікувати конкретних користувачів біткойну, і таким чином "де-анонімізували" їх та порушили загальну передумову транзакційної невидимості. Далі розглянемо, як саме вони це зробили. Веб-трекери та "cookie"-файли на веб-сайтах дозволяють витік інформації про транзакцію в інтернет мережі, де будь-хто, включаючи уряд, правоохоронні органи та зловмисників, з готовністю використовують ці дані.

Існують навіть деякі компанії, що направлені виключно для відстеження та де-анонімізацію публічних блокчейнів. Наприклад, компанія Elliptic[4] пропонує інтерактивний досліджувальний процесор, який схематично зображає потік коштів між біткойн-мережею, платіжними системами, біржами, форумами, фінансовими ринками, благодійними організаціями та іншими установами. На Рисунку A.1 показано графік, де описані деякі транзакції в біткойнах на початку 2010-х років, у тому числі зв'язки між великими угрупуваннями майнерів – біржами Mt. Gox[2] та Silk Road[3].

Крім того, використовуючи такі блокчейн платформами, як

Ethereum[5], користувачі взаємодіють зі смарт-контрактами, які обробляють більше даних, ніж у звичайних грошових переказах. Всі деталі зазначених смарт-контрактів загальнодоступні в блокчейні Ethereum, включаючи відправників, одержувачів, дані про транзакції, виконаний код і поточний стан мережі. Тому завантаження важливих бізнес-даних у блокчейн, де хакери, конкуренти або інші неавторизовані сторони можуть переглядати інформацію, просто не є можливим рішенням для більшості компаній. Розглянемо деякі приклади таких приватних даних:

– *Електронні медичні записи*, які є надзвичайно приватною та конфіденційною інформацією. Неприпустимо, щоб вона коли-небудь була виголошена громадськості в загальнодоступних блокчейнах, тим самим загрожуючи конфіденційності пацієнта.

– *Дані перевірки ідентичності*, такі як номери соціального страхування, не можуть бути відкрито збережені в смарт-контракті.

– *Управління обліковими записами*, де паролі та ключі користувачів не можуть мати місце у відкритому смарт-контракті.

– *Фінансові документи*, такі як таблиці капіталізації або зарплати працівників, ніколи не повинні бути публічно пов'язані з адресами, які можна легко відстежити.

Отож конфіденційність залишається основною перешкодою для окремих осіб, організацій і галузей, які піклуються про приватність і індивідуальний суверенітет. Парадоксально, але ми використовуємо загальнодоступну, легко простежувану книгу обліку, щоб створити систему, де буде досягнута безпека приватних даних. Проте варто прослідкувати історію криптовалют та їх розвитку. Блокчейн у свій час був створений, щоб усунути централізовані інстанції, які концентрують занадто багато влади у своїх руках, та замінити їх на децентралізовані механізми, вбудовані в саму систему та підпорядковані розподіленим вузлам користувачів. Тому забезпечення приватності – це лише наступна сходинка для удосконалення поточного протоколу блокчейну.

Оглянемо кілька прикладів існуючих рішень у цьому питанні, з якими працювали різні команди розробників.

1) *Адреси з еліптичної кривої Діффі-Хелмана-Меркле (ECDHM).*

Щоб освоїти поняття адрес ECDHM, потрібно розуміти алгоритм обміну ключами Діффі-Хелмана. Його ідея полягає в тому, що він встановлює спільний секрет між двома сторонами. Наступний приклад проілюструє, як саме це здійснюється. Припустимо, що Аліса хоче створити спільний ключ з Бобом, але єдиний канал, доступний для них, може бути підслуханий третьою стороною. Спочатку встановлюються параметри еліптичної кривої – "параметри домену" (p, a, b, G, n, h). Крім того, кожен учасник повинен мати пару ключів, придатну для використання з еліптичною кривою, що складається з приватного ключа d (випадково вибране ціле число в інтервалі $[1, n-1]$) і відкритого ключа, що представлений точкою Q (де $Q = dG$), тобто результатом додавання G до самої себе d разів. Нехай пара ключів Аліси є (d_A, Q_A) і пара ключів Боба є (d_B, Q_B). Кожна сторона повинна знати відкритий ключ іншої сторони до початку виконання протоколу. Аліса обчислює точку $(x_k, y_k) = d_A Q_B$, а Боб – точку $(x_k, y_k) = d_B Q_A$. Спільний секрет – x_k (координата точки x). Він однаковий для обох сторін, тому що згідно їх розрахунків: $d_A Q_B = d_A d_B G = d_B d_A G = d_B Q_A$.

Зазначений алгоритм може бути використаний для приватного обміну повідомленнями через загальнодоступну мережу. Відправник і приймач обмінюються один з одним адресами ECDHM, а потім використовують спільний секрет для отримання анонімних адрес в мережі блокчейну. Ці адреси можуть бути розкриті тільки тими, хто володіє секретом. Єдине, що видно публічно, це багаторазові адреси ECDHM. Проте вони приховують лише ідентичність користувача, а не його загальну транзакційну діяльність, де ці адреси фігурують у якості відправника чи отримувача коштів. А як вже зазначалося, поведінка учасників мережі це той фактор, який наразі активно досліджується.

2) *Змішувачі.*

Ідея змішувача полягає в тому, що група користувачів може об'єднати власні виплати в одне угрупування, відстежуючи свої позички в окремій інстанції – приватній книзі, що надається їм як інструмент для виплат. Потім, коли кошти з угрупування витрачаються, джерела кожного платежу "затьмарюються". Будь-який користувач блокчейну, може побачити сплачені суми разом з одержувачами, але теоретично неможливо простежити особу, яка ініціювала платіж. Прикладом сервісу змішування слугує платформа CoinJoin [6].

На жаль, змішувачі виявилися ненадійним рішенням. Дослідження[7] показали, що транзакції CoinJoin можна легко визначити і довели, що, витративши лише 32,000 доларів США, зловмисник спроможний анулювати анонімність транзакції з 90-відсотковим успіхом. Більше того, було доведено, що змішувачі забезпечують лише незначний захист від Сібіл-атаки[8] та атаки відмови в обслуговуванні. Ще більш критичним є той факт, що прихованою приватною книгою обліку змішувача повинен управляти деякий центральний орган, а це означає, що треба довіряти третій стороні для "змішування" транзакцій.

3) *Криптовалюта Monero*[9].

Інший спосіб забезпечення конфіденційності – це створення криптовалюти, яка є приватною за замовчуванням. Прикладом такої технології є Monero ("Монеро"). На відміну від багатьох інших альтернатив, Монеро не є форком від біткойну. Натомість ця криптовалюта базується на альтернативному протоколі CryptoNote[10], що вимагає забезпечення двох властивостей, а саме неспроможності відстежити користувача та пов'язати його особу з певною транзакцією. Тому, щоб виконати ці вимоги, Монеро пропонує використання схеми "кільцевого підпису" та генерування одноразових "невидимих адрес" для кожної транзакції. Кільцеві підписи – це тип групового підпису, де кожен учасник має секретний і відкритий ключ. На відміну від традиційних криптографічних підписів, які доводять, що транзакція була "схвалена" одним підписувачем, використовуючи згаданий приватний ключ, підпис

групи підтверджує, що один підписувач з фіксованої групи схвалив транзакцію, не розкриваючи його ідентичності.

Але й протокол Монеро має певні недоліки реалізації, як, наприклад, можливість побудови кільцевої атаки[11] та аналізу простежуваності транзакційної діяльності користувачів[12].

4) *Доказ з нульовим знанням.*

Це криптографічний протокол, з допомогою якого користувач може переконати будь-кого, що він має певні знання, не розкриваючи їх безпосередньо. Доказ з нульовим знанням забезпечує фундаментальні примітиви, які можна використовувати для побудови механізмів збереження конфіденційності.

5) *Заплутування коду.*

Його мета полягає в тому, щоб знайти спосіб зробити програму P такою, що "обфускатор" (заплутувач коду) міг виробляти іншу програму $O(P) = Q$, таку, що P і Q повертають один і той же результат для однакових даних на вході. При цьому Q не показує інформацію про внутрішній стан P. Саме це дозволяє зберегти приховані конфіденційні дані всередині Q, такі як паролі, номери соціального страхування тощо, але все одно використовувати їх у програмах.

Хоча дослідження[13] показують, що загальне заплутування коду в чорному ящику неможливе, існує слабше поняття, відоме як нерозрізнене заплутування, яке можна реалізувати. Визначення нерозрізнення обфускатора O полягає в тому, що якщо взяти дві еквівалентні програми A і B (тобто для однакових вхідних даних A і B виробляються однакові вихідні дані) та обчислити $O(A) = P$ і $O(B) = Q$, то не є обчислювально можливим для будь-кого, хто не має доступу до A або B, визначити, чи прийшов результат P з A чи B.

Нещодавно дослідники Крейг Джентрі, Аміт Сахаї та інші[14] були в змозі досягти нерозрізненого заплутування коду. Однак отриманий алгоритм має високі накладні витрати. Якщо його можна буде покращити, потенційна користь виявиться дуже значною. Адже найбільш

цікавою можливістю в світі криптовалют є ідея смарт-контракту на блокчейні, що містить приватну інформацію. Наприклад, можна уявити існування контракту в мережі Ethereum, який міститиме пароль користувача до Coinbase. Тоді можливо буде написати програму такою, що при задоволенні певних умов контракту, він, використовуючи деякий проміжний вузол, ініціює HTTPS-сеанс з Coinbase та увійде з паролем користувача й здійснюватиме торгівлю. Оскільки інформація в контракті буде заплутаною, проміжний вузол або будь-який інший користувач блокчейну не зможе змінити запит у транзиті або визначати пароль користувача.

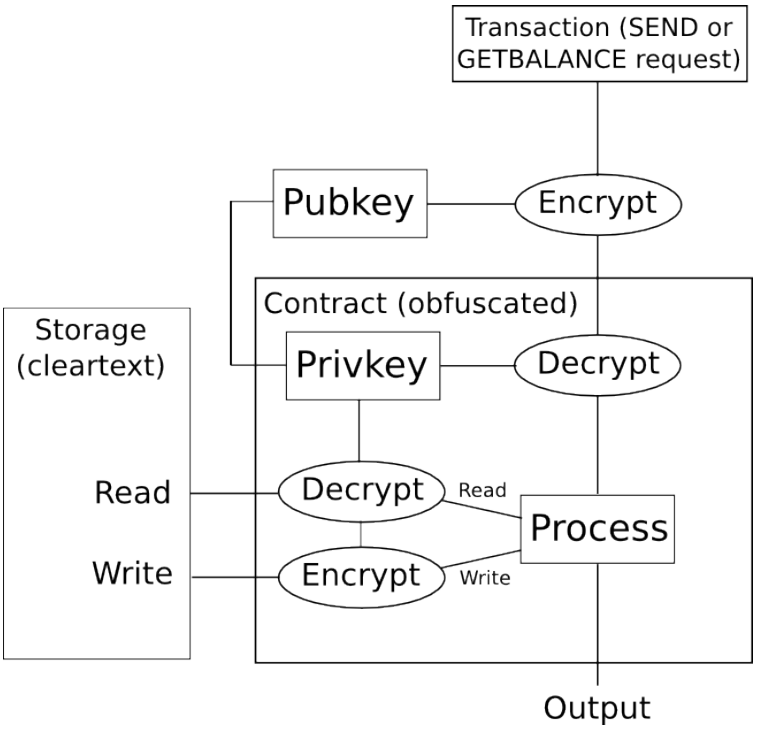

Рисунок 1.1 – Схема роботи смарт-контракту з заплутувачем коду

Щоб мати краще уявлення про використання обфускатора в блокчейні, наведемо ще один простий приклад, зображений на Рисунку 1.1. Ми створюємо смарт-контракт з заплутаним кодом, який містить закритий ключ і приймає інструкції, зашифровані за допомогою відповідного відкритого ключа. Контракт зберігає залишки на рахунках у зашифрованому сховищі. Якщо до нього прийде запит на зчитування

даних зі сховища – він розшифровує їх внутрішньо, а при запиті на збереження даних – зашифрує бажаний результат перед його записом. Якщо користувач хоче отримати стан балансу свого рахунку, то він кодує цей запит як транзакцію і виконує його на своїй машині. Заплутаний код смарт-контракту перевіряє підпис у цій транзакції, щоб визначити, чи має він право перегляду даного балансу. При позитивних результатах перевірки, контракт поверне розшифрований баланс, а в іншому випадку користувач отримає повідомлення помилки без розкриття будь-якої іншої інформації.

6) *Оракули.*

У просторі блокчейну оракул є механізмом, який передає інформацію між смарт-контрактами та зовнішніми джерелами даних. Схематично його робота зображена на Рисунку 1.2. Блокчейн не може отримати доступ до даних за межами своєї мережі, тому були створені оракули, як джерела даних, що надаються третьою стороною та призначені для використання в смарт-контрактах у блокчейні.

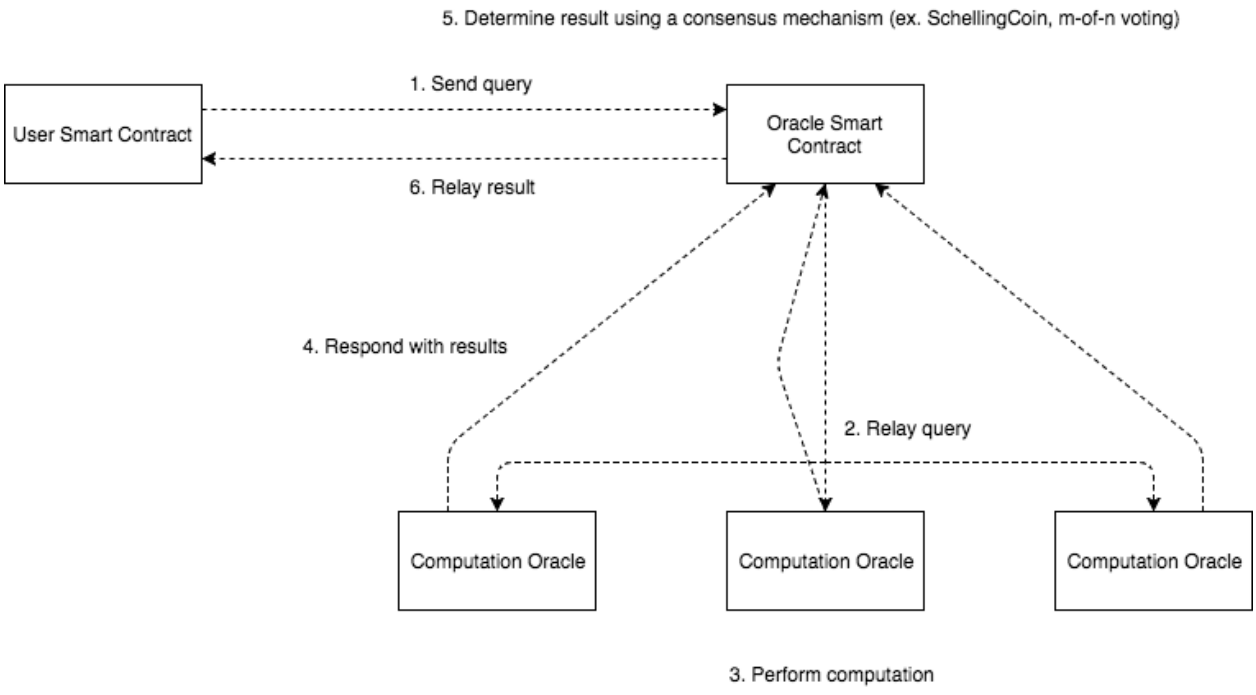

Рисунок 1.2 – Процес обробки транзакції користувача блокчейну з використанням оракула

Оракули надають зовнішні дані і запускають виконання смарт-контрактів, коли зустрічаються заздалегідь визначені умови. Такими умовами можуть бути будь-які показники, такі як температура погоди, успішно проведені платежі, коливання цін тощо. Оракули є частиною контрактів з мульти-підписом, де, наприклад, довірені особи підписують контракт для майбутнього переведення коштів лише за виконання певних умов. До того, як будь-які кошти будуть переведені, оракул також має підписати цей смарт-контракт.

Проте оракули – це послуги стороннього сервісу, який не є частиною загального механізму консенсусу в блокчейні. Тому головною проблемою є те, що користувачі блокчейну змушені довіряти цим джерелам інформації.

7) *Довірені середовища виконання (TEE).*

TEE являє собою захищену зону головного процесора. Вона гарантує, що коли дані завантажені всередину, вони захищені з урахуванням конфіденційності та цілісності. TEE працює паралельно з операційною системою, якою користується користувач, але має на меті бути більш приватною та безпечною, ніж звичайні операційні системи. Продукт Intel SGX[15] є репрезентативною технологією для реалізації TEE. Наприклад, платформа Ekiden[16] – рішення на основі SGX для смарт-контрактів із збереженням конфіденційності даних, де процеси обчислення відокремлені від алгоритму консенсусу. У ньому обчислення виконуються в смарт-контрактах в TEE на вузлах поза ланцюгом блокчейну, а потім використовується протокол дистанційної атестації для перевірки правильності їх результатів вже в ланцюзі. Вузли, що забезпечують консенсус мережі, використовуються для підтримки блокчейну і не вимагають використання надійного обладнання. Але варто зазначити, що TEE лише частково надають захист конфіденційних даних, так як пропонують лише довірене середовище для виконання не всіх, а частини операцій в блокчейні.

Отже, кожен з перерахованих механізмів так чи інакше вирішує задачу забезпечення анонімності користувачів та приватності їх даних

лише частково, тому варто зазначити перелік основних типів вимог до транзакційної системи блокчейну, при дотриманні яких мережа вважалась би захищеною:

– *узгоджений стан мережі поміж всіма вузлами*: учасники блокчейну мають прийти до консенсусу щодо його єдиновірного глобального стану, щоб він був функціональним;

– *цілісність транзакцій*: завжди існує ризик свідомої фальсифікації або підробки сертифікатів транзакцій з боку зловмисників, тому система повинна гарантувати цілісність виконаних операцій та запобігати будь-яким спробам внести до них зміни;

– *доступність даних системи*: доступність в даному контексті відноситься до обох системного та транзакційного рівнів, де з одного боку блокчейн повинен відповідати на запити користувачів у випадку мережевої атаки, а з іншого – дані його транзакцій повинні бути неушкоджені та доступні учасникам, які мають дозвіл на їх перегляд, в будь-який час;

– *запобігання подвійних витрат коштів*: для транзакцій, що виконуються в децентралізованому мережевому середовищі, потрібні надійні механізми безпеки і контрзаходи для запобігання витрат одних і тих самих коштів більше одного разу;

– *конфіденційність транзакцій*: дана характеристика означає, що неавторизовані користувачі не мають доступу до деталей транзакцій без згоди їх власника, а також їх захист та доступність навіть за умов виникнення помилок в системі або при кібер-атаках;

– *анонімність ідентичності користувача*: при тривалій фінансовій діяльності в мережі, збільшуються ризики розкриття особи власника певного облікового запису в системі, тому потрібно вводити додаткові протоколи та алгоритми, щоб їх максимально знизити.

– *незв'язність транзакцій*: користувачі прагнуть, щоб їх транзакції не можна було б пов'язати з їх ідентичностями і тим самим вони підсилюють свою анонімність.

1.2 Криптовалюта Монеро

Окремо розглянемо криптовалюту Монеро, як зразок однієї з найкращих реалізацій протоколу блокчейн з набором додаткових махізмів захисту приватності даних користувачів, їх ананонімності тощо. Тож, Монеро побудований на основі технології блокчейн, де дані транзакцій доступні для всіх, подібно до реалізованого в біткойні [17], та покладається на протокол PoW[18] для досягнення розподіленого консенсусу. Але, на відміну від біткойну, де кожен здатний відстежувати потік грошей між адресатами, у Монеро користувачі не можуть зробити те ж саме. Технології кільцевого підпису та одноразових відкритих ключів реалізовані як параметри за замовчуванням для поліпшення анонімності даних транзакцій. Справжні відправники "затемнені" шляхом додавання кількох так званих "приманок", де встановлено безліч еквівалентних між собою відправників, які неможливо відрізнити один від одного. Одноразовий відкритий ключ означає, що для кожного виходу буде створена своя унікальна адреса, в той час як реальна адреса отримувача в блокчейні не розкриється ніколи. Без будь-якої додаткової інформації неможливо визначити зв'язок між адресами та їх власниками.

Монеро – це криптовалюта, породжена з вже існуючої криптовалюти під назвою Bytecoin. Обидві монети базуються на протоколі CryptoNote, що був запропонований Ніколасом Ван Саберхагеном[19] у 2013 році. Ідея протоколу полягає у тому, щоб створити криптовалюту, яка б зберігала конфіденційність. У біткойні є проблеми, пов'язані з анонімністю користувачів, оскільки, як вже зазначалося, попередні дослідження змогли ідентифікувати інформацію про користувачів біткойну та їх діяльність у мережі.

Анонімність в технології Монеро розбивається на дві частини: незв'язність і невідстежуваність. Незв'язність визначається як

неможливість для будь-яких двох різних операцій з'ясувати, чи були вони відправлені до однієї і тієї ж особи, в той час як невідстежуваність визначається для набору входів: для них неможливо вирішити, який вхід реально був використаний транзакцією. Ґрунтуючись на даних визначеннях можна наголосити, що незв'язність стосується захисту приймачів коштів, а невідстежуваність полягає в захисті відправника. Обидва незв'язність і невідстежуваність включені в протокол CryptoNote і є тим, на чому головним чином фокусується система. Невідстежуваність досягається за допомогою кільцевого підпису. Незв'язність забезпечується завдяки використанню одноразових відкритих ключів. Ці особливості реалізовані на рівні протоколу, що робить дані процедури обов'язковими для всіх користувачів системи.

Як і в будь-яких інших криптовалютах, існує щонайменше двоє учасників, які утворюють середовище технології: демони та гаманці. Демон Монеро – це сервер, який надає інформацію клієнтам. Він синхронізує дані в блокчейні для своїх користувачів та зберігає повний запис про всі транзакції в локальному сховищі. Гаманець Монеро – це програма, яка допомагає користувачам керувати своїми гаманцями, а також виявляти, чи отримують вони нові платежі, обчислювати баланс і створювати нові транзакції. Гаманець Монеро не зберігає блокчейн в локальному сховищі. Замість цього він створює запити до демону задля здобуття будь-якої інформації, необхідної для оновлення даних.

На ринку є різні продукти гаманця Монеро. Перший і, мабуть, головний – це гаманець, створений основною групою розробників, який отримав назву "monero wallet-cli". Нова його версія обладнана GUI. Другий гаманець – онлайн гаманець під назвою "MyMonero1". Це веб-гаманець, який можна використовувати для укладання угод, здійснення транзакцій, а також сканування блокчейну для обчислення поточного балансу. "OpenMonero2" – це версія "MyMonero" з відкритим вихідним кодом та аналогічним інтерфейсом, але краще сумісним з "monero-wallet-cli". Третій продукт – Android-сумісний гаманець, що

називається ”Monerujo3”, який також є проектом з відкритим вихідним кодом. ”OpenMonero” і ”Monerujo” використовують ту ж кодову базу, що й офіційний ”monero-wallet-cli”, щоб обробляти обчислення для Монеро мережі, але вони використовують свій інтерфейс. Інший гаманець Monero, наданий Freewallet5 є проектом з закритим вихідним кодом і не рекомендований спільнотою Монеро до використання, адже він зберігає в собі приватні ключі своїх користувачів.

При створенні транзакції гаманець Монеро не може працювати сам по собі – він потребує інформації, наданої демоном Монеро. Це тому, що в даній технології кожен реальний вихід, який витрачається на вхідні дані необхідно ”заплутати” кількома іншими виходами (”заплутувачами”). Схему цього процесу можна побачити на Рисунку 1.3. Заплутувачі та реальний вихід використовуються, щоб побудувати кільцевий підпис. Число заплутувачів разом з реальним виходом формують розмір кільця.

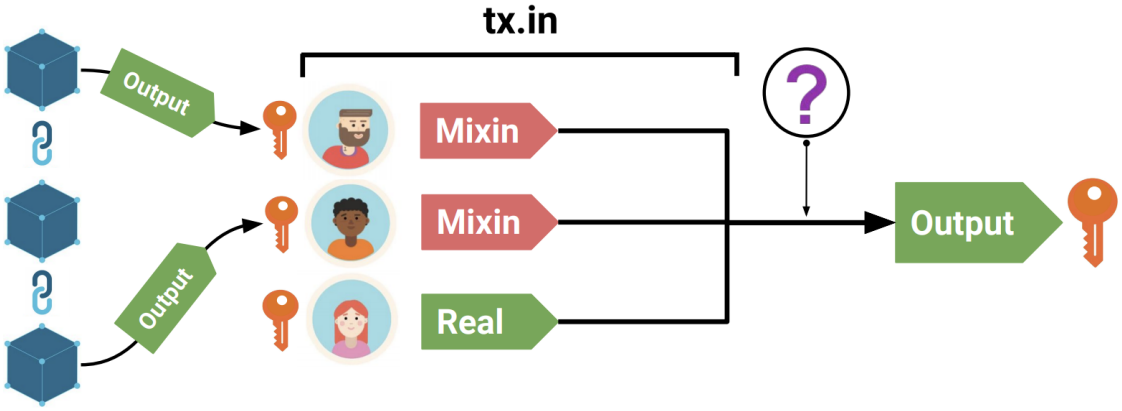

Рисунок 1.3 – Схема процесу формування кільцевого підпису в криптовалюті Монеро

Заплутувачі – це справжні публічні ключі, які вже з’являлися у блокчейні. Іншими словами, вони були виходами з інших операцій. Ці відкриті ключі групуються на основі кількості монет, що містяться в публічних ключах, а потім послідовно індексуються на основі їх появи в блокчейні по часових рядах. Спочатку гаманець Монеро робить запит на отримання ”даних гістограми”. Це інформація максимального індексу для

кожної суми в групі виходів у блокчейні. На її основі обирається декілька індексів, кількість яких перевищує розмір кільця. Для транзакції "RingCT"[22] індекси будуть вибрані з гістограми виходів з сумою 0. Оскільки вся інформація про суми у транзакціях починаючи з транзакції RingCT зашифрована, система не може її прочитати і відмічає суму як 0, хоча вона може бути й ненульова. Щоб забезпечити перевірку балансів на вході і виході транзакції належним чином, використовуються базові верифікації: кожна сума на балансах повинна мати позитивне значення, не виходити за межі певного числового діапазону тощо.

Кількість ключів транзакції на виході безпосередньо пов'язана з кількістю адрес, які отримують монети Монеро (позначаються як "XMR"). Кожна адреса отримувача містить вихідний ключ та відповідну кількість XMR монет. Ці вихідні ключі пізніше будуть обрані системою для заплутування майбутніх транзакцій. Та незважаючи на особливість Монеро у вигляді механізму заплутування, обмеження в його впровадженні перешкоджає системі досягти свого повного потенціалу. Досліди показали, що велику частину транзакцій Монеро все ж можна простежити [20, 21].

Щоб досягти виконання умови незв'язності транзакцій з ідентичностями їх власників, до технології Монеро було додано протокол "невидимих адрес з подвійними ключами" (DKSAP). Він розроблений, щоб приховати особу отримувача коштів. Даний протокол використовує дві пари криптографічних ключів, а саме пару ключів для сканування і пару для витрат. Приклад формування одноразової адреси оплати за транзакцію зазначено нижче:

1) приймач має дві пари приватних та відкритих ключів (s, S) і (b, B), де $S = s \cdot G$ – "відкритий ключ сканування", $B = b \cdot G$ – "відкритий ключ витрат", а G є базовою точкою групи еліптичної кривої;

2) відправник генерує пару ключів (r, R), де $R = r \cdot G$, і передає R з транзакцією;

3) тепер і відправник, і приймач можуть обчислити загальний

секрет c за допомогою формули з алгоритму ECDH: $c = H(r \cdot s \cdot G) = H(r \cdot S) = H(s \cdot R)$, де $H(\cdot)$ є криптографічною хеш-функцією, узгодженою сторонами до початку роботи протоколу;

4) відправник використовує значення $c \cdot G + B$ як адресу призначення для надсилання платежу;

5) приймач активно відстежує блокчейн і перевіряє, чи була передана деяка транзакція до адреси призначення $c \cdot G + B = (c + b) \cdot G$. Якщо знайдена відповідність, оплата здійснюється з використанням відповідного закритого ключа $(c + b)$. Зауважимо, що закритий ключ $c + b$ може бути обчислений лише приймачем.

Варто підкреслити, що кожен обмін коштами має плату, що компенсує роботу майнерів, які виконують різного роду обчислення для підтвердження транзакцій. Крім того, ця плата збільшується в залежності від розміру транзакцій, розрахованого в байтах. Кількість вхідних даних, заплутувачів та виходів мають безпосередній вплив на розмір транзакції. До того ж, плата змінюється разом з поточною винагородою за генерацію блоку та з врахуванням інших еталонних значень, визначених протоколом. Сума виплат нараховується в XMR за кожний кілобайт даних в транзакції й визначається наступною формулою[23]:

$$Fee\ per\ kB = (R/R_0) \times (M_0/M) \times F_0 \times (60/300) \times 4$$

де R – базова винагорода за генерацію блоку; R_0 – орієнтовна базова винагорода (10 XMR); M – обмеження розміру блоку для майнера, щоб уникнути випадків з надмірним розміром блоку; M_0 – фіксований максимальний розмір блоку (300 Кілобайт); F_0 = 0,002 XMR; 60/300 – коефіцієнт коригування для врахування збільшення максимального розміру блоку з обмеженням від 60 до 300 Кілобайт; 4 – коефіцієнт коригування множника плати. Операції за замовчуванням мають азначення 4, а мінімальна плата має множник 1.

Станом на кінець 2017 року плата за транзакції досягла пікового середнього значення в 20 доларів США[24], головним чином через бум на ринку криптовалют. Цей приклад показує, що атака переповнення ("SYN flood attack"[25]) може стати досить дорогою для зловмисників. Вважається, що це була одна з причин, чому даний тип атаки не був досліджений розробниками Монеро.

Ще одною цікавою компонентою даної технології, над якою почала роботу спільнота Монеро, став протокол Kovri[26]. Це реалізація маршрутизатора I2P[27], що написаний на мові C++. Коли користувач створює транзакцію, він повідомляєте всю мережу про те, що хочете, щоб вона була включена в наступний блок. І, як наслідок, його IP-адреса разом з іншими метаданими стають доступні іншим учасникам системи. Так, IP-адреса користувача не зберігається в блокчейн назавжди, проте зловмисники можуть спробувати з'ясувати цю IP-адресу, якщо вони активно стежать за мережею. Цей факт підкреслює важливість проекту Kovri.

Висновки до розділу 1

Ідентичності користувачів в блокчейн мережах приховані за допомогою використання різних криптографічних примітивів. Це робить, на перший погляд, неможливим визначити особу власника рахунку, що є однією з головних причин великої поширеності цієї технології. Однак слід зазначити, що криптовалюти на основі блокчейну не є повністю приватними. Оскільки всі дані транзакцій фіксуються, включаючи відправника, одержувача та суму обміну, й, крім того, загальнодоступні – їх легко зчитати та дослідити.

Наразі існує велика кількість компаній, що аналізують різні аспекти

діяльності клієнтів криптовалют, і на основі своїх результатів, в окремих випадках, можуть встановити особу користувача. Тож, як наслідок, з плином часу застосовується все більше й більше підходів та криптографічних методів для забезпечення приватності користувачів. У даному розділі було розглянуто основні з них. Особливу увагу приділено задачам, що вони виконують, та які недоліки мають у контексті технології блокчейн. Даний аналіз допоміг сформувати головні вимоги до захисту анонімності користувачів блокчейну та збереження конфіденційності даних їх транзакцій.

Серед усіх інших новітніх технологій, особливо виділяється криптовалюта Монеро, що завдяки використанню в своєму протоколі кільцевих підписів та одноразових публічних ключів, дає своїм клієнтам достатньо високий рівень анонімності, попри деякі зазначені в розділі уразливості з боку аналізу транзакційної діяльності користувачів.

2 АНАЛІЗ БЛОКЧЕЙНУ З ОБМЕЖЕНИМ ДОСТУПОМ

У даному розділі розглянуто протокол блокчейну з обмеженим доступом на прикладі фреймворку Hyperledger Sawtooth[28], а також проаналізовано основні його характеристики у порівняння з іншими приватними та публічними блокчейнами.

2.1 Огляд технології Hyperledger Sawtooth

Hyperledger Sawtooth – це бізнес-проект з відкритим кодом, побудований на блокчейні. Він був створений організацією Intel Corporation в об'єднанні з R3cev (творцями Corda[29]). Це платформа для розподілених рішень з модульною архітектурою, що забезпечує високу ступінь конфіденційності, надійності, гнучкості та масштабованості. Він був розроблений для підтримки різних реалізацій його компонент і враховує тонкощі та складності, які існують в економічній екосистемі.

Головні характеристики протоколу Hyperledger Sawtooth:

– дозволяє додати різні типи консенсусу в одну й ту саму блокчейн мережу;

– за умовчанням наявний такий механізм консенсусу як "Доказ вичерпаного часу" (PoET);

– смарт-контракти можна писати практично на будь-якій мові програмування (Python, JavaScript, Go, C ++, Java і Rust);

– включає підтримку смарт-контрактів Ethereum через інтеграцію з Hyperledger Burrow [31];

– може реалізувати мережу як з обмеженим, так і необмеженим доступом з боку користувачів;

– має модульну архітектуру;

– високо масштабований;

– транзакції можуть за необхідності виконуються паралельно, даючи таким чином більшу загальну продуктивність.

Основна мета дизайну Sawtooth спрямована на збереження розподілених записів користувачів та безпечне використання смарт-контрактів, особливо для великих компаній. Sawtooth спрощує розробку додатків на основі блокчейну, відокремлюючи основну систему від домену проекту. Розробники додатків можуть вказувати бізнес-правила, які підходять саме у їх випадку, використовуючи мову програмування на їх розсуд, без необхідності знання базового дизайну основної системи.

Слід також зазначити, що Sawtooth дуже модульний продукт. Це дозволяє підприємствам і консорціумам приймати нагальні рішення, які підходять їм найкраще. Конструкція ядра Sawtooth дає змогу обирати правила обробки транзакцій, дозволів та алгоритми консенсусу, які підтримують їх унікальні бізнес-потреби.

Проект клієнта Sawtooth може базуватися на вбудованій бізнес-логіці чи на віртуальній машині зі смарт-контрактами. Насправді, обидва типи додатків можуть співіснувати в одному й тому ж блокчейні. Sawtooth дає змогу реалізувати ці архітектурні рішення на етапі обробки транзакцій. Отож, всі вимоги клієнта визначаються у спеціальному процесорі транзакцій для їх виконання.

До того ж, Sawtooth надає декілька кастомізованих сімей транзакцій, які служать в якості моделей для функцій низького рівня (наприклад, підтримка загальносистемних налаштувань і зберігання дозволів у блокчейні), а також для конкретних додатків, таких як аналіз продуктивності та зберігання інформації блоків.

Sawtooth побудований для вирішення завдань приватних мереж з обмеженим доступом. Кластери його вузлів легко розгортаються з окремим переліком дозволів. У ньому немає централізованого сервісу,

який міг би потенційно викривати транзакційну діяльність користувачів або іншу конфіденційну інформацію. Блокчейн зберігає в собі параметри, які визначають політику дозволів, такі як ролі та ідентифікатори, щоб усі учасники мережі могли отримати доступ до цієї інформації.

Більшість існуючих блокчейнів вимагають послідовне виконання транзакцій, щоб гарантувати правильне та єдине їх упорядкування в кожному вузлі мережі. Sawtooth включає в себе розширений планувальник паралельних процесів, який розбиває обробку транзакцій на паралельні потоки. В залежності від загального стану системи, яку оновлює операція, Sawtooth ізолює виконання транзакцій одне від одного, зберігаючи контекстні зміни. Тому, коли це можливо, транзакції виконуються паралельно, не допускаючи атаки, пов'язаної з подвійними витратами, навіть при декількох модифікаціях одного й того ж стану. Паралельне планування додає системі значного потенціалу у збільшенні продуктивності в порівнянні з послідовним виконанням.

Крім того, Hyperledger Sawtooth підтримує створення і трансляцію подій в мережу. Це дозволяє програмам:

1) підписатися на події, які пов'язані з блокчейном, наприклад, отримувати інформацію про генерацію нового блока або перехід на нову версію;

2) підписатися на конкретні події програми, визначені сімейством транзакцій;

3) передавати інформацію про виконання транзакції назад клієнтам, не зберігаючи ці дані в загальному стані.

У блокчейні консенсус – це процес узгодження групи учасників, які не довіряють один одному. Алгоритми досягнення консенсусу з невеликим коефіцієнтом випадкових помилок зазвичай вимагають проведення голосування серед відомого набору учасників. Загальні підходи включають в себе консенсус у стилі Накамото[32], який обирає лідера за допомогою певної форми лотереї, і варіанти традиційних алгоритмів візантійської відмовостійкості (BFT)[33], що використовують кілька

раундів голосувань для досягнення консенсусу.

Серед особливостей Hyperledger Sawtooth найбільш вражаючим є його алгоритм консенсусу, який називається "Доказ вичерпаного часу"'(PoET). Він передбачає вибір лідера серед вузлів та реалізований за використання процесорів Intel, що включають в себе SGX. Sawtooth абстрагує основні концепції консенсусу та ізолює його від семантики транзакцій. Інтерфейс підтримує інтеграцію з різними реалізаціями консенсусу. Що ще більш важливо, Sawtooth дозволяє різні типи консенсусу на одному й тому ж блокчейні. Один алгоритм узгодження налаштовується під час початкової конфігурації мережі та може бути змінений на запущеному блокчейні з використанням специфічної транзакції, роблячи дану компоненту динамічною.

У Sawtooth модель даних і мова транзакцій реалізовані в так званих "сім'ях транзакцій". Хоча очікується, що користувачі створюватимуть власні такі структури даних, які відображають унікальні вимоги до їхніх систем, технологія надає декілька основних вбудованих сімей транзакцій:

– "IntegerKey": використовується для тестування розгорнутих блокчейнів;

– "Settings": надає реалізацію для зберігання налаштувань конфігурації мережі;

– "Identity": обробляє дозволи, визначені в ланцюгу блокчейну для транзакційних і верифікаційних ключів, щоб спростити управління процесам ідентифікації, що використовують списки відкритих ключів.

Додаткові сім'ї транзакцій надають моделі для конкретних областей:

– "Smallbank": збирає та обробляє результати аналізу продуктивності мережі, що потрібні для порівняння з іншими блокчейнами;

– "BlockInfo": надає методологію для зберігання інформації про конфігураційну кількість історичних блоків.

Сім'ї транзакцій кодують у собі бізнес-правила, які використовуються для зміни стану, тоді як клієнтські програми зазвичай передають операції до виконання та перегляду стану. Як вже зазначалося, можна створювати

власні сім'ї транзакцій, які будуть відображати певні унікальні вимоги, беручи вбудовані сім'ї транзакцій за основу.

2.2 Архітектура приватного блокчейну

Одна з цілей технології Sawtooth полягає в розподілі спільних записів про транзакції між учасниками системи. Можливість забезпечити консистентне копіювання даних на всіх вузлах у візантійському консенсусі є однією з основних переваг технології блокчейн.

Sawtooth представляє собою стан для всіх сімей транзакцій в одному екземплярі дерева Merkle Radix[34], вбудованого в кожний спеціальний "вузол-валідатор". Процес перевірки блоку у валідаторі гарантує, що однакові транзакції призводять до однакових переходів стану і що отримані дані однакові для всіх учасників мережі.

Стан розбивається на простори імен, що дозволяє творцям сімей транзакцій гнучко визначати, обмінюватися та повторно використовувати дані глобального стану між процесорами транзакцій. Зміни до стану додаються шляхом створення та виконання транзакцій. Клієнт створює транзакцію і передає її валідатору, а він, у свою чергу, застосовує транзакцію, яка призводить до зміни стану.

Транзакції завжди організовані всередині деякого пакету, групи. В його межах вони або виконуються всі, або взагалі не застосовуються у системі. Таким чином, дані групи є одиницею зміни стану. Це значно спрощує управління залежностями з точки зору клієнта, оскільки транзакціям в межах пакету не потрібні явно оголошені залежності. В результаті, користь таких залежностей обмежена лише тими випадками, коли транзакції не можуть бути розміщені в одному пакеті через їх занадто велику кількість.

Загальна структура пакетів і транзакцій, зображених на Рисунку 2.1 включає в себе поля *Batch* (”пакет”), *BatchHeader* (”заголовок пакету”), *Transaction* (”транзакція”), and *TransactionHeader*(”заголовок транзакції”).

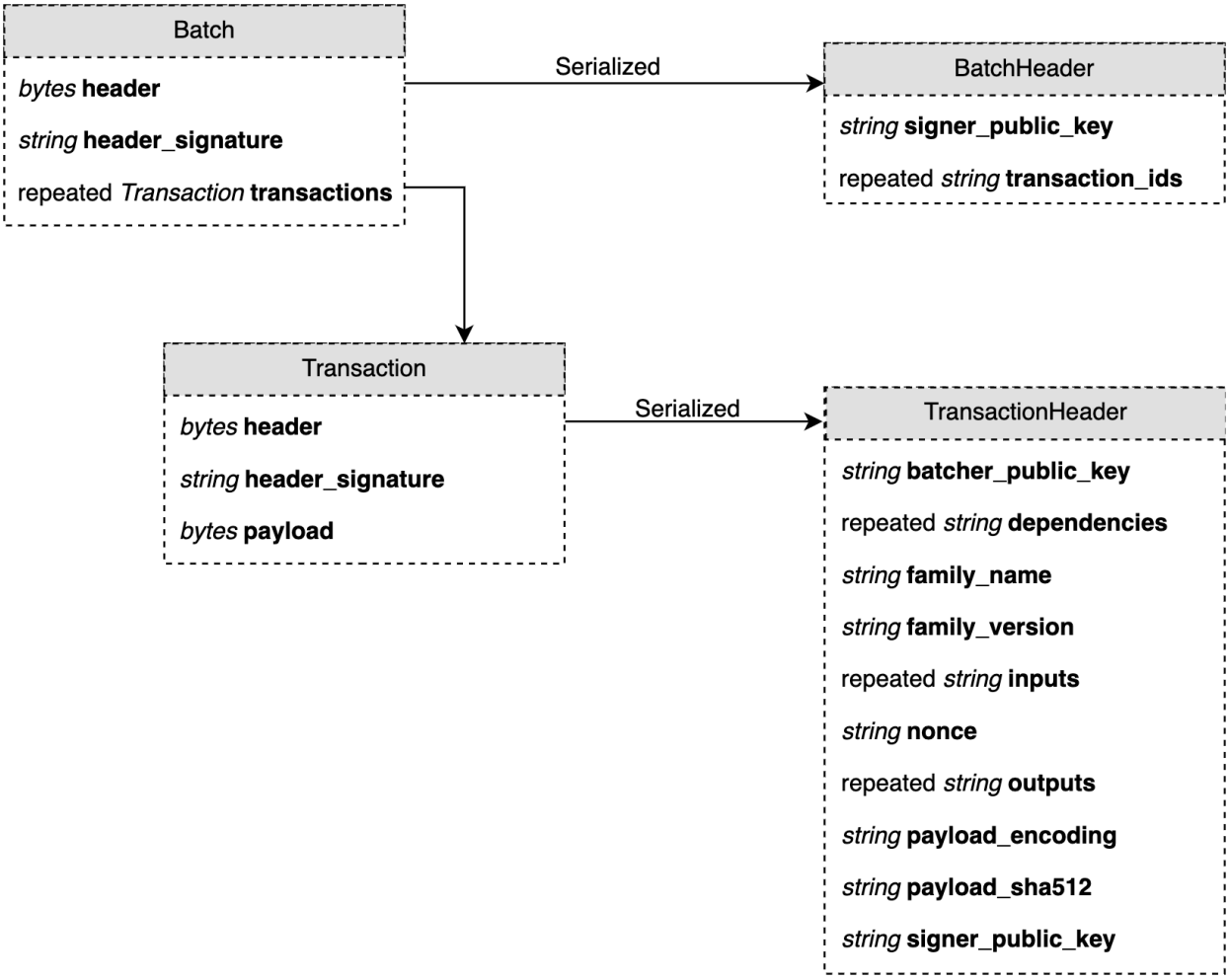

Рисунок 2.1 – Структура пакетів і транзакцій платформи Sawtooth

Поле *header* транзакції є серіалізованою версією структури заголовка транзакції. Він підписується закритим ключем відправника (не надсилається разом із транзакцією), а отриманий підпис зберігається в атрибуті *header_sign*. Заголовок присутній у серіалізованій формі, тому при отриманні транзакції, дані байти можуть бути звірені з підписом. Під час цього процесу перевіряється, що ключ в атрибуті *signer_public_key* підписав байт заголовка і утворив заголовок *header_signature*.

Поле *batcher_public_key* має відповідати відкритому ключу, який

використовується для підписання пакету, в якому міститься ця транзакція. Отриманий серіалізований документ підписується з приватним ключем ECDSA, використовуючи криву secp256k1[35].

Валідатор очікує 64-байтовий "компактний" підпис. Деякі бібліотеки включають додатковий байт заголовка, поле відновлення ID або надають підписи, кодовані в DER форматі. Sawtooth відхилить підпис, якщо він має розмір не 64 байта.

Поле *payload_sha512* містить SHA-512 хеш даних транзакції. Як і частина заголовка, воно підписується й пізніше перевіряється, в той час як поле *payload* – ні. Для верифікації того, що поле даних відповідає заголовку, обчислюється SHA-512 даних та порівнюється з *payload_sha512*.

А от поле *nonce* містить згенерований клієнтом випадковий рядок. У випадку, якщо дві транзакції містять однакові атрибути, *nonce* гарантує, що вони будуть генерувати різні підписи заголовків.

Sawtooth підтримує як послідовне, так і паралельне планування транзакцій. Обидва з планувальників призводять до однакових детермінованих результатів і повністю взаємозамінні. Паралельна обробка транзакцій забезпечує поліпшення продуктивності навіть при великих робочих навантаженнях транзакцій за рахунок зменшення загального часу затримки, що акумулюється при виконанні транзакцій послідовно.

Наступним розглянемо поняття журналу. Це група компонентів валідатора, які працюють разом для обробки пакетів і пропонованих блоків. Ці компоненти відповідають за заповнення опублікованих блоків, додавання пакетів для розширення ланцюга і перевірку блоків для визначення того, чи слід їх розглядати для оновлення стану блокчейну. Блоки та партії надходять або через gossip протокол[36], або внаслідок запитів клієнта. Обробка цих блоків і пакетів здійснюється в декількох потоках. Розглянемо все детальніше згідно Рисунку 2.2:

1) Спочатку *Completer* отримує блоки і пакети. Це гарантує, що всі залежності між ними задовільні.

2) Підготовлені пакети переходять до *BlockPublisher* для їх верифікації та включення в блок.

3) Завершені блоки переходять до *ChainController*, щоб їх перевірити. *BlockCache* і *BlockStore* забезпечують зберігання пакетів і блоків, які обробляються.

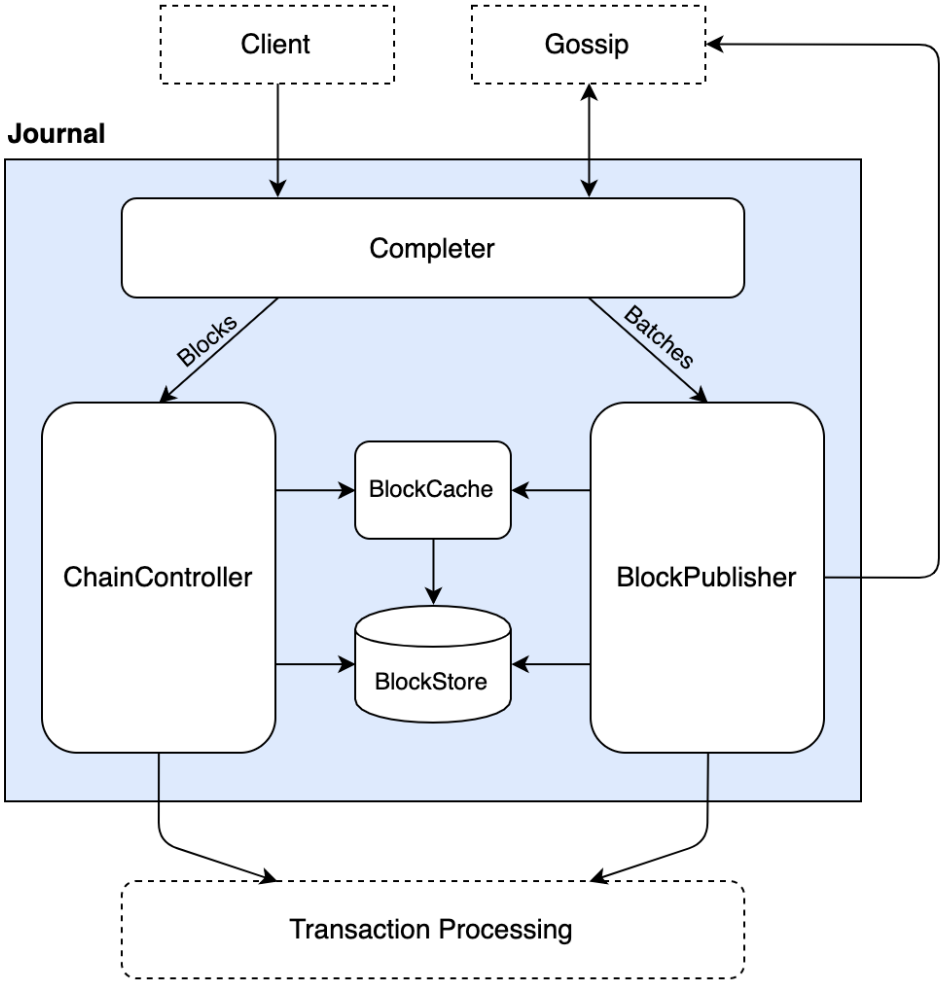

Рисунок 2.2 – Структура даних в журналі Sawtooth

Пакетна і блокова обробка розроблена бути асинхронною, що дозволяє *ChainController* опрацьовувати вхідні блоки паралельно, а *BlockPublisher* – приступити до генерації блоків навіть при високому навантаженні. Цей підхід є досить гнучким для роботи з різними алгоритмами консенсусу. Sawtooth включає консенсусний інтерфейс, який пов'язаний з компонентами в журналі.

Тепер перейдемо до механізму консенсусу. Алгоритм PoET запускається в межах довірених середовищ виконання (TEE), доступних на процесорах Intel з SGX. SGX, в свою чергу, - це набір процесорних інструкцій, які дозволяють виконувати код в межах захищеної області пам'яті. Ця частина особливо важлива, тому що це єдина річ, яка дозволяє PoET правильно функціонувати як консенсусний алгоритм і перевіряти його результати зовнішніми сутностями.

Алгоритм імітує протокол "Доказу виконаної роботи" (PoW), але замість того, щоб конкурувати за вирішення криптографічної задачі і генерації наступного блоку, кожен перевіряючий вузол отримує випадковим чином тайм-аут, і той, в кого він найкоротший стає лідером даного блоку й відповідає за створення і перевірку нового блоку та його додавання в блокчейн. Ця дія засвідчує, що верифікація транзакцій була виконана правильно. Алгоритм консенсусу PoET є гібридом випадкової лотереї і порядку прибуття. Як і PoW, цей алгоритм не має дуже гарної швидкодії, оскільки після додавання блоку клієнту доведеться почекати кілька хвилин, щоб переконатися, що в ланцюжку немає жодної гілки, яка довша від тієї, яка містить його транзакції.

Надзвичайно, але Sawtooth розроблений таким чином, що консенсусний механізм може бути змінений на льоту: у ньому користувачі спроможні висунути пропозицію нового консенсусу як спеціальну операцію до виконання, а потім узгодити політику в межах своєї мережі з іншими вузлами, щоб прийняти цей новий консенсусний алгоритм. Тож ця важлива компонента дуже гнучка та може з часом модифікуватися в залежності від загальних потреб.

І навіть якщо у певній мережі відсутні процесори Intel з SGX, фреймворк Hyperldeger Sawtooth все ж можна реалізувати, оскільки він автоматично пропонує симулятор PoET, який забезпечує консенсус у стилі PoET на будь-якому типі апаратного забезпечення, включаючи віртуальне хмарне середовище. Для розробки та видозміни технології блокчейн на основі Hyperledger Sawtooth доступний особливий режим

спрощеного алгоритму обрання випадкових лідерів.

2.3 Порівняння властивостей публічних та приватних блокчейнів

Окреслимо відмінності між Ethereum[5], Hyperledger Fabric[30] та Hyperledger Sawtooth, щоб більш ефективно виділити особливості кожного з них.

Ethereum – багатоманітний термін, адже він може означати клас технологій, що побудовані на базі віртуальної машини Ethereum (”Ethereum Virtual Machine”, EVM), або мережу блокчейн, керовану проектом Ethereum. Дана мережа призначена бути відкритою та загальнодоступною. Це означає, що кожен вузол Ethereum відомий будь-якому іншому вузлу мережі. Таким чином, усі вузли мають копію однакових впорядкованих даних. Відносини між вузлами повністю децентралізовані, тобто немає головного учасника системи, який керує іншими – кожен вузол просто інформує спільноту мережі про типи своїх транзакцій та порядок їх виконання.

Кожен користувач мережі Ethereum взаємодіє з іншими через смарт-контракти, які повинні бути в нього ”вбудовані”. Процес інсталяції смарт-контракту подібний до того, як учасники мережі Ethereum ініціюють транзакції. Наприклад, автор контракту криптографічно підписує та передає його через свій вузол всім іншим учасникам. Контракт зберігається в деяких адресних сховищах даних мережі. Слід наголосити, що блоки в блокчейні не є сховищем даних – вони містять тільки опис функцій, які будуть викликатися в смарт-контрактах.

Наведемо приклад обробки транзакції в такій мережі. Уявімо укладання угоди між двома сторонами Джоном і Паулом, де Джон обіцяє

заплатити Паулу якусь суму грошей через мережу блокчейн. У цьому випадку Джон безпосередньо не надсилає Паулу повідомлення. Замість цього, він підписує і посилає запит на вузол, яким він володіє, інформуючи його про необхідність виклику деякої функції ”pay” в смарт-контракті. При цьому є вимога наявності в транзакції криптографічного підпису, що зазвичай не потрібен в традиційних системах обробки транзакцій. Потім вузол розміщує це повідомлення в пулі транзакцій. У мережі деякі вузли, відомі як майнери, добровільно додають повідомлення Джона до нового блоку, а потім генерують блок в блокчейні за рахунок своїх обчислювальних можливостей. Оскільки в мережі Ethereum існує більше одного майнера, вони конкурують між собою у вирішенні випадково згенерованої головоломки, при цьому витрачаючи певну кількість енергетичних ресурсів. Цей механізм відомий як PoW. Майнер отримує винагороду за той обсяг роботи (в Ethereum відомий як ”газ”), що був необхідний для перевірки валідності транзакції та генерації відповідного нового блоку.

Усі вузли в мережі Ethereum покладаються на головний ланцюг записів, щоб переконатися, що зміна стану в їхньому обліковому записі узгоджена з іншими вузлами. Майнери, в свою чергу, відтворюють інструкції, записані в блокчейні, і порівнюють стан до і після досягнення консенсусу. Коли майнер успішно вкладає транзакції в блок і додає його в блокчейн, новий стан ланцюгу транслюється всім вузлам мережі. Потім вони самі несуть відповідальність за оновлення власних сховищ даних або рахунків. Вузли, що не належать до майнерів, також забезпечують консенсус шляхом відтворення транзакцій, інкапсульованих в блокчейні.

Продовжимо огляд з визначення технології Hyperledger Fabric. Вона належить до класу блокчейнів з обмеженим доступом. У мережі Fabric вузли не спілкуються один з одним відкрито. Натомість, учасники системи, як правило, деякі бізнес-організації, щоб обмінюватися повідомленнями, повинні узгодити політики видачі дозволів своїм вузлам. Краще розглядати транзакції між сторонами як операції між

бізнес-організаціями. Кожна організація може володіти одним або декількома вузлами. За допомогою Fabric вони налаштовуються для виконання конкретних завдань, таких як створення клієнтського вузла для взаємодії з кінцевими користувачами тощо.

Починаючи з версії Hyperledger Fabric 1.0, формування дозволів обміну повідомленнями між вузлами ґрунтується на використанні сертифікатів X.509[37]. У цій моделі всі організації-учасники погоджуються використовувати загальний кореневий сертифікат, такий, який видається центром сертифікації Verisign[38], щоб перевірити, чи дозволено сторонам здійснювати операції один з одним.

Можуть використовуватися й інші форми механізму видачі дозволів. Кожна організація управляє власними користувачами самостійно. Немає необхідності в централізованому уповноваженому управлінні всіма користувачами мережі Fabric, звісно якщо вимоги учасників не протилежні. У будь-якому випадку, технологія може бути налаштована тим чи іншим чином в обох напрямках.

Коли користувач з однієї організації виявляє бажання співпрацювати з користувачем іншої організації, відбувається подібний до Ethereum процес, за винятком того, що існує ще три етапи – схвалення, впорядкування та відправлення.

Покроково прослідкуємо весь процес обробки транзакції, зображений на Рисунку 2.3:

1) Користувач передає транзакцію кільком або всім вузлам в мережі Fabric, відомим як вузли схвалення.

2) Такі вузли належать різним організаціям. Їх завдання перевірити підпис транзакції та симулювати її виконання.

3) Коли всі вузли схвалення досягають узгодженого стану, вони повертаються до вузла клієнта, що ініціював транзакцію, та передають йому результати.

4) Потім клієнтський вузол посилає ці дані вузлу впорядкування, завдання якого полягає в тому, щоб обгорнути транзакцію в блок.

5) Вузол упорядкування додає транзакцію за принципом структури даних черги, тобто той елемент, який з'явився перший – і буде обслуговуватися в першу чергу. Альтернативним підходом до упорядкування блоків є використання технології обміну повідомленнями Kafka[39] для додавання транзакцій до блоку.

6) Коли всі транзакції додані до блоку, вузол упорядкування повідомляє про новий стан всі вузли мережі, кожен з яких відповідає запит на оновлення свого сховища даних.

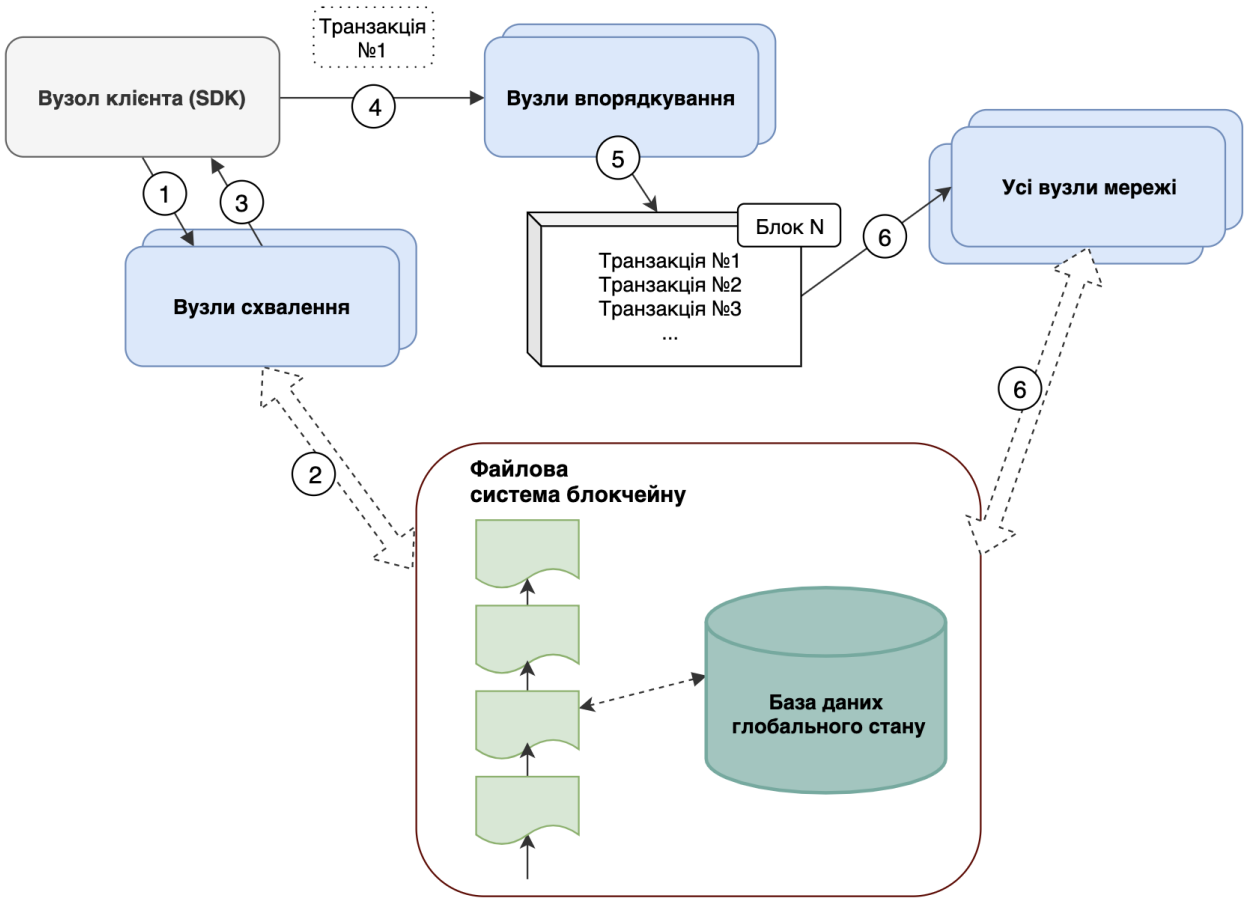

Рисунок 2.3 – Загальний механізм обробки транзакції в мережі Hyperledger Fabric

Додамо, що вузли впорядкування зазвичай розміщуються іншою організацією як нейтральний елемент мережі Fabric. При встановленні смарт-контрактів, вони повинні бути відправлені кожній організації, з якою взаємодіє даний клієнт. Учасники, в свою чергу, відповідають за

встановлення смарт-контрактів у власних вузлах.

Унікальним аспектом мережі Fabric є можливість розмежування дозволів для груп вузлів у окремі ”канали”. Вузли, що належать одному каналу, не можуть здійснювати операції в іншому каналі. Його часто розглядають як певний прошарок для приватних комунікацій між кінцевими користувачами. Проте це не так. Більш точно канал можна окреслити як сукупність вузлів або підмереж, які мають спільний екземпляр блокчейну, а вузол упорядкування може створювати блоки лише для тих вузлів, у яких є дозвіл брати участь в даному каналі.

Завершимо додатковим оглядом технології Hyperledger Sawtooth. Це унікальний блокчейн. Він може бути налаштований так, що мережа буде працювати як публічна або з обмеженим доступом. Тим не менш, він більш пристосований до роботи з формуванням та маніпулюванням політиками дозволів користувачів. Смарт-контракти Ethereum та Fabric - це ті програми, що представляють собою вид контрактних угод між двома учасниками операції. У випадку Sawtooth, концептуальний еквівалент цьому отримав назву процесора транзакцій. Його механізм побудови та встановлення ні чим не відрізняється від звичайного додатку.

Можна розглядати процесор транзакцій як форму обчислювальної платформи, що дозволяє виконувати смарт-контракти. Sawtooth має свій процесор транзакцій – Seth, заснований на віртуальній машині Hyperledger Burrow EVM. Він дає змогу виконувати смарт-контракти Ethereum, написані на мові програмування Solidity. В Sawtooth є еквівалент майнера або вузла замовлення – ”валідатор”. Його завдання полягає в тому, щоб перевірити та переконатися, що транзакція (або повідомлення, що викликає виконання транзакції) має дійсний підпис. Потім він направляє її до відповідного процесора транзакцій, де реалізована бізнес-логіка, що відповідає за правильність обробки транзакції. Коли процесор транзакції завершує перевірку своєї бізнес-логіки, повідомлення надсилається назад валідатору, який потім відповідає за упаковку транзакції в блок і додавання його до блокчейну, а

також за його передачу іншим валідаторам, щоб переконатися, що консенсус досягнутий. На додачу до групування транзакцій в блоки, він також відповідальний за оновлення глобального стану мережі, тобто сховища даних на основі дерева Merkle.

У мережі Sawtooth може існувати не один, а багато валідаторів. Так само, як і майнер в Ethereum, тільки один валідатор може оновлювати блок. Проте замість PoW, Sawtooth робить це за допомогою алгоритму PoET.

Важливою особливістю Sawtooth є його здатність групувати транзакції і формувати так звані пакети, які згодом можуть бути додані до блоку. Таким чином, Sawtooth спроможний обробляти складні угоди. Уявімо ситуацію, де Джон продає свою машину Паулу, що може включати ряд пов'язаних між собою операцій: обмін грошима, передача власності та страхового полісу тощо. Щоб спростити обробку цих даних, можна налаштувати процесори транзакцій спеціальним чином. Проте всі ці операції мають бути або успішними або невдалими у сукупності. Іншими словами, об'єднані транзакції повинні відповідати принципу атомарності, консистентності, ізоляції та тривалої дії (ACID). Також можна налаштувати операції, пов'язані з передачею права власності та страхового полісу, щоб вони виконувалися паралельно з операцією, пов'язаною з обміном грошима, яка повинна виконуватися тільки після того, як перші дві були успішно завершені. А коли обмін грошей здійснюється, весь пакет транзакцій задовольняє принципам ACID.

Нарешті, щоб налаштувати Sawtooth як публічну мережу або чи мати обмеження в плані доступу, потрібно маніпулювати процесорами транзакцій, а саме налаштувати маршрутизацію трафіку від клієнтів відповідно їх вимог: приймати транзакції від будь-якого клієнта в мережі (випадок з загальнодоступним блокчейном) або приймати повідомлення лише від тих клієнтів, яким це дозволено (тобто блокчейн з обмеженим доступом).

Висновки до розділу 3

Hyperledger Sawtooth – один з багатьох прикладів реалізацій блокчейнів з обмеженим доступом, що надають своїм користувачам набір послуг та можливостей, які недоступні в публічних блокчейн мережах. Ця технологія вирішує задачі закритих організацій, які через свої бізнес-вимоги не в змозі використовувати відкриті децентралізовані системи у тому стані, в якому вони знаходяться зараз. Структура Sawtooth з його простою модульною архітектурою не тільки забезпечує його швидке освоєння, але й дозволяє легко налаштовувати мережу та розгортати спеціальні програми, наприклад, сім'ю транзакцій і запускати їх на масштабованій мережі за лічені хвилини. Не менш важливою його компонентою є протокол консенсусу PoET, що представлений як альтернатива традиційному алгоритму PoW.

Підсумуймо основні характеристики трьох водночас схожих і таких різних технологій, що були порівняні у цьому розділі:

– Ethereum: це загальнодоступний блокчейн, де всі вузли повністю децентралізовані, а транзакції впорядковані майнерами. Кожен окремий учасник мережі відповідальний за оновлення своїх даних, щоб синхронізуватися з загальним станом. Майнери отримують право оновлювати блокчейн за допомогою процесу, відомого як PoW.

– Hyperledger Fabric: блокчейн з обмеженим доступом, де вузли відомі іншим вузлам тільки тоді, коли їх власники дозволяють це. Дана політика базується на процесі сертифікації X.509. Транзакції виконуються послідовно, проходячи через етапи їх схвалення, впорядкування та виконання. Блоки впорядковуються за принципом черги або на основі механізму Kafka. Отже, вузли впорядкування фактично не повністю децентралізовані. У випадку конфігурації черги, існує тільки один вузол замовлення для мережі, а при налаштуванні технології Kafka такі вузли

можуть бути розподілені, але для виконання своєї роботи вони все ще залежить від централізованого механізму. Використання кореневого сертифіката також вводить єдине джерело відмовостійкості.

– Hyperledger Sawtooth: може працювати або в режимі публічного блокчейну чи як блокчейн з обмеженим доступом, залежно від його конфігурацій. Усі вузли в будь-якому режимі повністю децентралізовані. За замовчуванням він використовує алгоритм консенсусу PoET, щоб обрати вузли , які будуть виконувати перевірку транзакцій. Також, можна організовувати декілька транзакцій у пакети і обробляти їх у паралельних потоках, коли це необхідно. Однією з не менш важливих особливостей є підтримка смарт-контрактів (у тому числі й Ethereum-подібних) та зручна побудова процесорів транзакцій.

3 ДОСЛІДЖЕННЯ РОБОТИ БЛОКЧЕЙНУ В ПОЄДНАННІ З МЕРЕЖЕЮ LIGHTNING

Блокчейн – прогресивна технологія, що набуває великої поширеності та, звичайно, все більшого списку вимог. До найкритичніших на сьогодні недоліків даного протоколу можна віднести неспроможність до масштабованості та прогалини в збереженні конфіденційності даних всередині системи. Тож, була запропонована нова децентралізована розподілена система – "Lightning Network" (далі позначимо це поняття як "мережа Lightning")[40], яка має на меті мінімізувати перераховані недоліки та удосконалити стандартний протокол блокчейну в цілому. Дана технологія може змінити спосіб ведення бізнесу між людьми. Вона відкриває світ мікро-транзакцій, а також допомагає в реалізації додаткових традиційних бізнес-моделей, що раніше не мали змоги використовувати блокчейн через повільність його роботи, викриття деталей кожної транзакції загалу тощо. У цьому розділі детально досліджено механізми та підходи в мережі Lightning, як саме вона доповнює роботу протоколу блокчейн та які переваги при цьому отримують користувачі.

3.1 Загальний огляд протоколу

Багато хто вважає, що біткойн забезпечує своїм користувачам анонімність їх операцій, проте експерти криптовалют знають, що насправді це не так [41]. Блокчейн біткойну — це загальнодоступна прозора книга обліку, що ніколи не видаляє стару інформацію. Тим не менш, дотримання вимог та безпека стали все більш важливим питанням

у галузі криптовалют. Закони про боротьбу з відмиванням грошей (AML) та регуляції щодо принципу "знай про свого клієнта" (KYC) вимагають обміну даних для моніторингу депозитів та зняття коштів, перевірки походження та призначення капіталів клієнтів. Тож, було створено новий підхід до грошового обміну між користувачами блокчейну, що вперше був реалізований саме в біткойні – мережу Lightning.

Ключова відмінність, яку приносить використання мережі Lightning, як доповнення до блокчейну, полягає в тому, що не вся інформація обміну коштами між користувачами доступна усім учасникам системи. Канали в мережі Lightning синхронізуються лише між її вузлами, тому робота цієї технології швидша та ефективніша, ніж в традиційному блокчейні. Крім того, це дає більш високий ступінь конфіденційності інформації, що передається в системі. Дані каналу відомі тільки двом задіяним вузлам мережі. Якщо звичайна транзакція біткойну подібна до завантаження виписки з банку на загальнодоступний веб-сайт, то транзакція мережі Lightning більш схожа на показ конкретному продавцю лише тієї частини вашого гаманця, з якої йому сплачуються кошти. Таким чином, учасники все ще обмінюються інформацією, але її обсяги набагато менші.

Протокол біткойну базується на блокчейні, але саме через це має кілька ключових бар'єрів, які обмежують його функціональність:

– Використання блокчейну для проведення транзакцій може бути досить дорогим. Оскільки в ньому обробляється все більше й більше транзакцій, він стає перевантаженим, і користувачі повинні платити вищі збори, щоб підтвердити свою транзакцію в наступному блоці.

– Він відносно повільний. У випадку з біткойном підтвердження транзакції займає приблизно годину.

– Дані транзакції в блокчейні відомі всій мережі. А тому, кожен користувач має уявлення про всі фінансові операції інших користувачів, що не задовольняє вимогам багатьох бізнес-моделей сьогодення.

Мережа Lightning – це нова технологія, що працює в кооперації з блокчейном, та є прикладом протоколу "другого шару". Перший шар – це

блокчейн, другий – мережа Lightning. Слід зазначити, що крім біткойну дана технологія може працювати й на блокчейнах інших криптовалют. Тож, перерахуємо, які переваги приносить мережа Lightning в поєднанні з блокчейном:

– *Додає можливість миттєвих операцій.* Оскільки перевірка більше не повинна відбуватися у блокчейні, немає необхідності чекати, поки угода буде підтверджена. Це означає, що платіж відбувається миттєво. Дана перевага дозволяє активізувати обмін коштами більш "спонтанно".

– *Дозволяє мікро-ціноутворення.* Оскільки миттєві мікро-платежі відбуваються поза основним блокчейном, користувачам більше не потрібно платити за їх підтвердження. Ціноутворення може бути збільшене в одиницях, наприклад, таких малих як 0,000000001 BTC. Це відкриває нову можливість раніше недоступних мікро-операцій.

– *Покращує збереження конфіденційності даних.* На відміну від блокчейну, мережа Lightning надає більшу фінансову конфіденційність. Тепер користувачі не можуть зчитувати всі транзакції іншого користувача.

3.2 Архітектура мережі Lightning

Розглянемо основні компоненти даної технології, а саме поняття вузлів, каналів, рахунків та багатьох інших, більш детально.

Вузол мережі Lightning має два обов'язки: моніторинг основного блокчейну та взаємодія з іншими вузлами мережі Lightning, щоб здійснювати транзакції. Кожен вузол мережі повинен контролювати один чи декілька блокчейнів, на яких він знаходиться. Окремо відзначимо, що мережа Lightning може працювати поверх кількох блокчейнів.

Протокол вимагає, щоб вузол мережі контролював основний блокчейн, в якому він тримає токени. Якщо вузол не робить це належним чином, кошти користувача можуть бути викрадені. Також кожен учасник взаємодіє з іншими вузлами в одноранговій мережі (P2P), де проходить передача коштів від одного вузла іншому через так звані "канали". Кожен вузол відповідає за відстеження того, хто й що має в каналі. І якщо користувач хоче вийти з каналу, він виводить свої кошти назад в блокчейн.

Тепер розглянемо, як саме вузол мережі Lightning відрізняється від вузла у блокчейні біткойну. Найбільша відмінність полягає в тому, що вузол біткойну повинен перевіряти кожну транзакцію в мережі блокчейн, в той час як вузол Lightning верифікує лише дійсність тих транзакцій, з якими він безпосередньо взаємодіє. Тож другий підхід більш здатний до масштабування, а також надає більшу міру конфіденційності.

Рахунок в мережі Lightning – це спосіб отримання платежів у мережі. Він подібний до адреси в біткойні, але з деякими ключовими відмінностями.

Приклад рахунку Lightning: "lntb1u1pvjluezpp5qqqsyqcyq5rqwzqfqqqsyqcyq5rqwzqfqqqsyqcyq5rqwzqfqypqdq5xysxxatsyp3k7enxv4jsxqzpuaztrnwngzn3kdzw5hydlzf03qdgm2hdq27cqv3agm2awhz5se903vruatfhq77w3ls4evs3ch9zw97j25emudupq63nyw24cg27h2rspfj9srp8". Він утворений з двох складових: частина доступна для читання користувачами та частина даних. Ці компоненти розділені останнім знайденим у рахунку символом "1".

У нашому прикладі "читабельною" частиною є "lntb1u". Перша складова "lntb", вказує, в якій саме криптовалютній мережі валідний цей рахунок, у цьому випадку вона ідентифікує тестову мережу біткойну. Друга складова "1u" – це сума коштів, закодована в рахунку ("1u" рівний одному мікро-біткойну або 0,000001 біткойну).

Тепер перейдемо до частини мета-даних рахунку: "pwz5w78pp5e8w8cr5c30xzws92v36sk45znhjn098rtc4pea6ertnmvu25ng3sdpywd6hyetyvf5hgueqv3jk6meqd9h8vmmfvdjsxqrrssy29mzkzjfq27u67evzu893heqex737dhcapvcuantkztg6pnk77nrm72y7z0rs47wzc09vcnugk2ve6sr2ewvcrtqnh3yttv847qqvqpvv398x".

Вона може містити таку інформацію, як:

1) Час, коли було створено рахунок, що дає можливість визначити, наскільки він старий.

2) Ідентифікатор вузла, який отримує платіж.

3) Терміну дії рахунку. На відміну від біткойн-адрес, рахунки мережі Lightning можуть стати не дійсними через деякий час.

4) Опис даного рахунку. Це може бути довільна строка з корисною інформацією про платіж.

5) Резервна адреса біткойну. Якщо платіж завершується з помилкою у мережі Lightning, то можна перейти до звичайної транзакції в блокчейні біткойну.

6) Маршрут для здійснення платежу. Потрібно пам'ятати, що мережа Lightning відрізняється від блокчейну в тому, що необхідно знайти маршрут для проведення платежу. Одержувач платежу може запропонувати такий маршрут.

Крім того, рахунки захищені цифровими підписами. Це означає, що якщо хтось змінить рахунок в мережі Lightning, підпис буде визнано недійсним. Даний фактор досить критичний, оскільки в рахунку закодована конфіденційна інформація, наприклад, ідентифікатор вузла, геш оплати та запасна біткойн-адреса. Якщо користувачам вдалося би змінити ці значення, вони мали б можливість вкрасти кошти інших учасників.

Загалом, рахунки мережі Lightning є надзвичайно гнучкими та корисними. Вони надають набагато більше мета-інформації про оплату, ніж традиційна адреса біткойну. Це дозволяє користувачам мережі Lightning бути більш впевненими в тому, коли, де і як в мережі обробляється їх платіж.

Наступним поняттям в огляді архітектури технології є "канали" мережі Lightning. Саме вони дозволяють користувачам відправляти та отримувати гроші від інших вузлів мережі. Їх можна порівняти з підключеними один до одного грошовими потоками. Всі такі канали

пов'язують лише два вузли мережі Lightning. Для наочності та зручності назвемо їх Алісою і Бобом. Канал має інформацію про біткойн-баланс як Аліси, так і Боба, та відстежує, скільки коштів є у кожного з них.

Один з користувачів – Аліса або Боб – можуть відкрити канал Lightning, створивши спеціальну біткойн-транзакцію, яка призначена для блокування коштів на блокчейні та подальшого їх розблокування у мережі Lighting. Цей підхід гарантує, що гроші не будуть витрачатися у кількох місцях одночасно. Для користувачів блокчейну транзакція виглядає так само, як звичайна для мережі біткойн, тобто без реальних відмінностей, за винятком декількох технічних деталей. Слід зауважити, що все це робиться автоматично за допомогою гаманця біткойну, і звичайним користувачам не потрібно знати усі подробиці розробки. Коли створюється канал Lighting, учасник повинен визначити, скільки коштів він хоче в ньому мати. Наприклад, Аліса хоче зробити транзакцію у блокчейні, що дала б їй доступ до 100000 сатоші в мережі Lighting. Дана операція передбачає блокування такої ж суми у блокчейні біткойну та її розблокування в мережі Lightning.

Після створення каналу Lightning, його власники можуть відправляти та отримувати гроші через нього. Якщо Боб хоче надіслати Алісі 1000 сатоші (0.00001 біткойну), він потребує її рахунок. Використовуючи його, він може надіслати Алісі платіж, збільшивши її баланс на 1000 сатоші і зменшивши свій на ту ж суму. Обидві сторони каналу відслідковують баланс один одного, переконавшись, що цифри сходяться після обміну коштами.

Постає питання, чи потрібно тоді користувачу мати канал з кожним з тих, кому він хоче переказувати кошти. Однією з основних переваг є те, що всі канали пов'язані між собою. Припустимо, Боб переконав свою подругу Керол також приєднатися до мережі Lightning і створив з нею платіжний канал. Аліса вже має канал з Бобом, а він тепер має канал з Керол. Тож, Аліса і Керол можуть платити один одному через Боба, який виступає для них своєрідним "маршрутизатором". При здійсненні платежу

в мережі Lightning вузол шукає шлях між ним та його місцем призначення – отримувачем коштів. Це те, що називається маршрутизацією.

Як біткойн, так і мережа Lightning розроблені таким чином, щоб зловмисники не отримували ніякої вигоди. Якщо хтось намагається укласти недійсну угоду в мережі Lightning, наприклад, стверджувати, що деякі з грошей Аліси насправді належать Бобу, то є вбудований механізм покарання зловмисника, де жертва атаки може забрати у злочинця свої гроші. Більш детально цей механізм оглянемо трохи нижче.

Перейдемо до операції закриття каналу. Щоб це зробити, також необхідно створити звичайну транзакцію у блокчейні біткойну. Ця транзакція, у порівнянні зі створенням каналу, виконує зворотні дії, видаляючи кошти з мережі Lightning та розблокуючи їх в основному блокчейні. Канали мережі Lightning є джерелом її живлення, дозволяючи вузлам обмінюватися даними і вільно платити поміж собою. Без них вузли мережі знаходилися б окремо та не робили нічого корисного. Разом вони формують фінансову супермагістраль майбутнього, що дозволяє радикально поширювати нові дані та бізнес-моделі.

Також обговоримо, які існують засоби маршрутизації в контексті мережі Lightning. Як вже згадувалося, вона не була б такою практичною чи навіть потрібною, якщо учаснику треба було підключатися до кожного вузла в мережі, якому він хотів або відправити кошти, або отримати їх. Однією з основних особливостей та нововведень, що була закладена в специфікації мережі Lightning, є платіжні каналі, які зв'язані разом, щоб дати змогу здійснювати платежі без необхідності безпосереднього з'єднання всіх учасників.

Проілюструємо це на прикладі. Нехай маємо учасників Алісу, Боба та Керол, що зацікавлені у спільному грошовому обміні. Боб – власник магазину, де Аліса і Керол купують речі кілька разів на тиждень, тому вони відкрили з ним канал. Однак слід зазначити, що вони не дуже часто платять один одному, тому для них не було б ефективним створення прямого каналу. Тож, вони спрямовують платежі через Боба:

1) Аліса виставляє рахунок і робить запит до Керол на 1000 сатоші;

2) Керол додає цей рахунок до її гаманця;

3) оскільки вона не підключена безпосередньо до Аліси, її вузол перевіряє, чи є у неї канали, які вона може використати;

4) її вузол знаходить маршрут через Боба, який, в свою чергу, з'єднується з Алісою;

5) Керол відправляє гроші Бобу, а Боб посилає ті ж гроші (за винятком невеликої плати за передачу) Алісі.

Звичайно, все це здійснюється автоматично за допомогою вузлів Lightning, що дають можливість обробити операцію миттєво. Тепер Керол заплатила Алісі 1000 сатоші, не будучи напряму пов'язаною з нею. Це те, що розуміється під маршрутизацією в мережі Lightning. Проте слід обміркувати і випадок, коли під час комунікації вузлів серед них з'являється зловмисник. Що робити у випадку, коли учасники не могли взаємодіяти за допомогою чесного Бобу і повинні були використовувати послуги зловмисника Мелорі? Чи могла Мелорі вкрасти їх гроші? Якщо дати відповідь коротко, то ні. Всі платежі в мережі Lightning вимагають від одержувача (наприклад, Аліси) згенерувати випадкову таємну частину – секрет. Транзакції в мережі Lightning створені таким чином, що передача грошей здійснюється при розкритті секрету. У нашому прикладі Керол хоче передати Алісі через Боба 1000 сатоші. Аліса продукує секрет і Керол повідомляє Боба про те, що якщо він зможе сказати їй цей секрет, вона заплатить йому 1001 сатоші. Після цього Боб йде до Аліси та купує її секрет за 1000 сатоші. Він розкриває таємницю Керол та успішно стає посередником між Алісою і Керол, а також отримує прибуток у розмірі 1 сатоші.

Потрібно мати на увазі, що все, описане вище, виконується вузлами Lightning автоматично шляхом створення, виконання та перевірки смарт-контрактів. Це гарантує те, що всі учасники в системі повинні діяти відповідно до встановлених правил. Якщо при цьому хтось намагається вкрасти гроші, то смарт-контракти мають вбудовані

механізми покарання, як вже згадувалося в частині огляду на поняття каналів Lightning.

Відзначимо, що в каналі з двома вузлами обидва учасники підписують всі обміни коштами, до яких долучаються, тому що вони залучені в кожну зміну стану мережі, враховуючи випадок, коли вихід з Lightning одного з них призводить до закриття спільного каналу. З цієї причини в даній системі працює схема мульти-підпису, при якому обидва учасники підписують одне й те ж повідомлення m, що представляє собою остаточний стан балансу кожного користувача після виконання чергової операції. Протокол мережі Lightning повинен враховувати вірогідність викиду учасників з каналу на будь-якому етапі роботи системи. Більше того, учасники мережі підписують і ділять між собою частину транзакції, щоб кожен користувач за потреби міг згодом реконструювати її деталі. Це вимагає криптографічної схеми на основі агрегованих підписів. За визначенням Д. Бонеха [42], *схема агрегованого підпису* є цифровою схемою підпису з додатковою властивістю, яка полягає в тому, що послідовність підписів $\partial_1, ..., \partial_n$ деяких повідомлень m_i з використанням відкритого ключа pk_i можна закодувати в єдину компактну сукупність підписів ∂, що перевіряє те, що m_i було підписано з pk_i для всіх $i = 1, ..., n$. Під час процесу верифікації перевіряються всі входи (pk_1, m_1),...,(pk_n, m_n)та приймаються чи відхиляються також в сукупності.

Для того, щоб у мережі Lightning маршрутизація працювала належним чином, необхідно знайти канали з достатньою ліквідністю. Розглянемо детальніше, що саме це означає. Повертаючись до нашого прикладу: якщо Керол хоче платити Алісі через Боба, то очевидно, що Керол має достатньо грошей у гаманці. Менш очевидним є факт того, що Боб також повинен мати певну суму грошей, в даному випадку у своєму каналі з Алісою. Без цих коштів він не зможе передати платіж Алісі. Тож, усі транзакції в мережі Lightning потребують побудови шляхів через об'єднання каналів з кількістю грошей, достатньою для завершення

платежу. В якості компенсації за забезпечення ліквідності каналів вузли можуть вимагати плату за перенаправлення платежів. Ці збори, як правило, дуже малі, за порядком одноцифрового сатоші. Нагадаємо, що 1 біткойн дорівнює 100 мільйонам сатоші. Однак, якщо вузол добре зв'язаний і маршрутизує багато платежів, потенційно, оператор вузла може заробити невелику кількість грошей на його утриманні.

Маршрутизація дійсно є особливістю мережі Lightning. Концепція платіжних каналів існувала довгий час, але можливість децентралізованої і безпечної маршрутизації через кілька сторін стала проривом порівняно з попередніми технологічними ітераціями.

Існують деякі ключові відмінності між тарифами в блокчейні і грошовими зборами за маршрутизацію платежів в мережі Lightning. Кожен раз, коли транзакція створюється в блокчейні, потрібно сплатити сервісний збір майнерам. Це компенсація за їх роботу, в результаті якої забезпечується безпека та валідність усієї мережі. Щоразу, коли майнер виявляє блок з набором транзакцій, він отримує винагороду за операційні витрати відповідно до характеристик блоку. Завдяки структурі блокчейну транзакції назавжди архівуються в системі. Це ще одна річ, за яку платять користувачі, коли транзакція перевірена та додана до блокчейну – база даних, що не підвладна редагуванню та доводить, що транзакція справді сталася. Ресурси обмеженої кількості, що забезпечують роботу блокчейну, – це електроенергія та файловий простір. Мережа Lightning поглинає інший дефіцитний ресурс – капітал. Для можливості здійснення транзакції в мережі Lightning, потрібно мати доступні кошти, щоб направити платежі до місця їх призначення. Якщо у користувача немає достатньої ліквідності в маршрутах, то надіслати платіж неможливо.

Мережа Lightning сконфігурована таким чином, що компенсує своїм учасникам можливість забезпечувати ліквідність мережі. Якщо ви є вузлом, який хотів би здійснювати маршрутні платежі, то можете встановити плату, яку інші вузли повинні враховувати, щоб використовувати свій капітал для маршрутизації.

У мережі Lightning існує два види зборів, що разом формують сумарний сервісну плату за перенаправлення платіжки: базова плата та комісія на основі ліквідності, що використовується. Базова плата – це фіксована ставка, яка нараховується за кожну операцію, що перенаправляється через вузол користувача. Наприклад, з інших учасників, які планують платежі через ваш вузол, можна стягувати 300 сатоші. Якщо ж ви захочете обробляти більше платежів, можна прийняти рішення про зниження ставки до 100 сатоші. Кожен оператор вузла встановлює цю плату на основі того, скільки, на його думку, коштує його капітал. Плата провайдеру ліквідності – це плата, яку можна нараховувати на основі кількості ліквідності, яку учасник мережі використовує у вашому каналі. Вона уособлює плату за кожний сатоші, що надсилається через канал мережі Lightning. Прикладом може бути ставка в розмірі 0,01 сатоші за кожний сатоші, що надсилається через платіж.

Досі невідомо, як саме в майбутньому буде розвиватися ринкова плата в мережі Lightning, проте тарифи становлять основні на рівні протоколу інструменти в розпорядженні оператора вузла, що призначені для стягнення операційних плат. Традиційний вузол мережі біткойн не продукує ніяких грошей для свого власника, незалежно від того, якою пропускною можливістю він володіє чи скільки обчислювальної потужності має. Всі грошові збори призначені майнерам, а не звичайним вузлам. Проте в мережі Lightning немає поняття майнінгу і всі збори надходять до операторів вузлів. Це одна з ключових відмінностей між вузлами біткойну та Lightning і те, що можливо сприятиме децентралізації усієї системи.

3.3 Питання безпеки підходу

Мережа Lightning має іншу модель безпеки у порівнянні з традиційною в блокчейні. Транзакція в мережі Lightning – це підписана біткойн-транзакція зі спеціальним смарт-контрактом, який ще не був доданий до блокчейну. Зазвичай операції, що не були включені в блокчейн, вважаються незахищеними через те, що майнери не витрачали ніяких ресурсів на підтвердження валідності транзакції.

Смарт-контракт, що додається до транзакції Lightning, є відмінною ознакою між транзакціями в мережі Lightning та біткойні. Цей контракт дозволяє безпечно пов'язувати непідтверджені операції разом. Учасникам смарт-контракту дозволяється неодноразово оновлювати ланцюжок непідтверджених угод, при цьому дійсною залишається лише остання його версія. Також контракт дозволяє стягувати гроші з тих, хто намагався опублікувати в блокчейні старий стан транзакції.

Звичайно, бажано, щоб лише останній стан в каналі був тією єдиною транзакцією, яка буде дійсною в блокчейні. Оскільки транзакції формуються в каналі Lightning, деякі стани більш вигідні для вас як користувача, тоді як інші – для вашого контрагента. Дана властивість "останнього стану, як єдино вірного" досягається тим, що спеціальний смарт-контракт штрафує людей, які намагаються транслювати в мережу старі стани. У ньому йдеться про те, що якщо ваш контрагент транслює у блокчейн старий стан, ви можете забрати всі гроші з його рахунку. Цей механізм як і спонукає зловмисників відмовитись від бажання шахраювати в мережі, так і захищає "хороших акторів" системи, які надають послуги в обмін на грошові виплати.

Коли канал створюється чи закривається в мережі Lightning, він залишає слід у блокчейні в вигляді транзакції, що може бути проаналізована так само, як і звичайна транзакція біткойну. Зауважимо,

що цей слід дуже мало повідомляє стороннім про те, як користувачі перенаправили свої платежі в мережі Lightning. Хоча, просто дивлячись на блокчейн, не очевидно, які транзакції пов'язані з мережею Lightning, а які – ні: той, хто хоче зібрати разом шляхи виплат, може поєднати декілька пунктів даних з різних наборів інформації.

Якщо певна компанія захоче проаналізувати, що відбувається в мережі Lightning, вона може ініціювати свій вузол Lightning і спробувати отримати якомога більше комунікацій з різними вузлами для відправлення платежів через свій вузол. Це дає деяку інформацію для аналізу, потенційно повідомляючи щось про моделі витрат користувачів. Вузли маршрутизації здійснюють перерахування платежів в мережі Lightning, переконавшись, що можна робити виплати без прямого каналу з учасниками. Одним з важливих аспектів перенаправлення є те, що маршрутизатори не знають початкових або кінцевих точок транзакції, яку вони передають. Це пов'язано з деякими криптографічними схемами, які використовуються в маршрутизації, подібно до протоколу Tor[43]. Однак, якщо великий вузол визначить, до яких інших вузлів маршрутизації він підключений, то зможе зробити деякі припущення щодо того, які платежі передаються.

Слід підкреслити, що для створення вузла Lightning потрібно досить мало зусиль. Це породжує простір для конкуренції, де учасники ведуть боротьбу за забезпечення ліквідності каналів. Тому користувачі можуть обирати маршрутизатори, які не контролюють їх транзакції. Маршрутизацію в мережі Lightning можна порівняти з майнінгом у блокчейні біткойну. Існування обох механізмів можливе без отримання дозволу від когось іншого й обидва вони мають помірно малі вимоги для входу та потребують капіталу для функціонування, але в мережі Lightning досить лише капіталу.

Важливою відмінністю, яку варто зазначити, є різниця між публічними та приватними каналами. Якщо було налаштовано персональний вузол, який не доступний у режимі онлайн цілодобово й на

нього не нараховано значні кошти, то немає достатньої підстави розголошувати інформацію про свій канал усій мережі. У випадку , коли не буде перенаправлятися багато платежів, можна створювати приватні канали. Вони не виставляються напоказ, тому в них задіяні лише два вузи. Загалом, кінцеві користувачі, наприклад, мають можливість відкривати приватні канали для декількох великих постачальників ліквідності і окремий канал, про який нічого не розголошуватиме решті мережі. Сам вузол може навіть не бути відомий мережі. Це підвищує конфіденційність кінцевих користувачів, а також полегшує маршрутизацію.

Висновки до розділу 3

Мережа Lightning є більш приватною, ніж блокчейн біткойну, в основному через те, що транзакції відомі лише кільком обраним учасникам, а не всій мережі. Але все ж система розголошує деяку інформацію загалу і Lightning сам по собі не надає тієї міри приватності даних, як інші криптовалюти, орієнтовані на конфіденційність, прикладом яких є Zcash та Monero. Тому він потребує вдосконалення та правильно підібраної та сконфігурованої інтеграції.

4 ЗАПРОПОНОВАНІ ПІДХОДИ ЗАБЕЗПЕЧЕННЯ ПРИВАТНОСТІ ДАНИХ КОРИСТУВАЧІВ БЛОКЧЕЙНУ

У даному розділі ми розглянемо нові запропоновані підходи для підвищення конфіденційності блокчейн мережі на прикладі криптовалюти Монеро, поліпшення її масштабованості та швидкодії. Для цього наявні дві фокус-групи кінцевих користувачів, де перші – учасники, зацікавлені лише у власній анонімності, а другі – це низка компаній, що прагнуть вести торгівлю і публічно, й утворювати приватні мережі зі своїми постачальниками та іншими інстанціями, де повинен бути максимально прихований сам факт приналежності таких організацій до даної під-мережі. Обидві згадані групи мають одночасно й спільні потреби:

– приватність даних, що передаються в публічній мережі;

– швидкодія транзакцій;

– гнучкість та надійність технології;

– масштабованість системи;

– легкий перехід між публічною та приватними мережами;

– достатнього рівня верифікація транзакцій у будь-якій компоненті загальної системи тощо.

Основною задачею доповнення стандартного протоколу блокчейн новими підходами є загальне удосконалення виконання трьох основних умов щодо приватності даних учасників мережі: підвищення рівня анонімності користувачів, збереження конфіденційності їх даних та підтримка незв'язності транзакцій. Головним чином, саме приховання тим чи іншим способом частини транзакцій клієнтів допоможе звузити канали дослідження їх фінансової діяльності в блокчейні, тим самим виконуючи поставлені задачі.

Для побудови моделей нових підходів застосовані вже проаналізовані в попередніх розділах технології приватного блокчейну з обмеженим

доступом та мережа Lightning у поєднанні з криптовалютою Монеро. Всі перераховані протоколи лише частково вирішують задачу захисту конфіденційності даних користувачів та в основному орієнтовані на різні типи їх вимог. На перший погляд, об'єднання понять приватного та публічного блокчейнів здається недоречним, проте іноді цікаві та своєчасні рішення породжуються з на перший погляд абсолютно протилежних підходів – саме їх взаємодоповнення допомагає спільно вирішити набагато більшу низку проблем.

4.1 Огляд концепції

З плином часу блокчейн, як і перша його реалізація – біткойн, отримали успіх планетарного рівня. Вірно, ця технологія справді відкрила технологічно новий світогляд, адже підійшла до проблем сьогодення – проблем окремих і суспільства загалом, під зовсім іншим кутом. Замість довіри до інституцій влади, фінансових установ, пропонується рішення, побудоване на підрахунках, математиці та криптографії. На противагу ієрархіям та централізованому управлінню представляється інший, сміливіший, прозоріший та розподілений підхід. У ньому наголошується, що кожен, хто хоче приєднатися до глобальної мережі блокчейну, повинен підпорядковуватися представленим правилам та вкладати свій внесок у роботу всієї системи.

Тож з року в рік технологія блокчейн розвивається, стає популярнішою, і згодом вже кожна велика або мала компанія шукає способи її використання. Але не всіх блокчейн влаштовує у чистому первозданному вигляді. Навіть досконалі рішенням в один момент часу можуть потребувати модифікацій у майбутньому.

Біткойн має деякі незмінні конфігурації у своєму протоколі,

прописані в його ключових компонентах. Це, наприклад, сталий розмір блока та фіксована можлива кількість згенерованих монет, що зводить нанівець спроби модифікувати сам протокол для підтримки кращої масштабованості. Ті самі помилки в конфігурації були повторені в інших криптовалютах, що базуються на PoW: Lightcoin, BitcoinCash тощо. Навіть класичний Ethereum, що зробив розмір динамічним, все ж не вирішив задачу з обмеженою кількістю можливо згенерованих монет. Тому в випадку біткойну було запропоновано інше рішення – додати мережу другого рівня під назвою Lightning Network. Ця технологія не тільки додає можливість масштабування, а й об'єднує низку транзакцій між двома користувачами в одну. Таким чином, результативність методів поведінкових аналізів користувача та викриття його ідентичності значно зменшується, бо стає менше даних до опрацювання. Саме ця перевага і буде використана в новому гібридному підході.

Монеро було обрано для інтеграції з новими протоколами не випадковим чином, а тому що ця криптовалюта надає перелік методів забезпечення приватності користувачів по замовчуванню, на відміну від інших альтернативних криптовалют, орієнтованих на приватність даних, наприклад, ZCash (де приватність транзакцій вмикається по бажанню клієнта). Крім того, Монеро має динамічний розмір блоків, тому пропускна спроможність мережі ($\approx$ 1600 транзакцій в секунду при розмірі блоку в 100 Кілобайт) набагато більша, ніж в інших таких криптовалютах, як біткойн чи Ethereum ($\approx$ 5 - 16 транзакцій в секунду при розмірі блоку в 1 Мегабайт), які також базуються на протоколі PoW. Звичайно, технологія Монеро потребує вдосконалення, і у цьому розділі буде описано які проблеми вона має і що пропонується для їх вирішення.

Гібридний підхід дозволяє переносити кошти з блокчейну Монеро у вторинний блокчейн і навпаки. Такий переказ насправді є ілюзією: кошти не передаються, а тимчасово блокуються на блокчейні Монеро, тоді як еквівалентна кількість "токенів" розблоковується у вторинному блокчейні. І навпаки, оригінальні монети можуть бути розблоковані, коли відповідна

кількість токенів на вторинному блокчейні знову заблокуються. Таким чином, підмережі головного блокчейну маніпулюють лише посиланням на гроші користувача. Проблемою даного підходу може бути те, що воно буде реалізоване на практиці лише тоді, коли вторинний блокчейн матиме скінченний цикл розрахунків. Проте рішення цьому є, і воно криється у завданні додаткових параметрів в протоколі, що будуть лімітувати сторонню від головного блокчейну діяльність користувача.

4.2 Перший підхід: використання блокчейну з обмеженим доступом

Публічні блокчейни, орієнтовані на забезпечення приватності даних та ідентичностей своїх користувачів до сьогодні були направлені на підвищення їх анонімності, проте, як і в реальному житті, бувають випадки, коли задля досягнення спільної мети організації об'єднуються в окрему від всіх групу та, звичайно, знайомі один з одним. Задачею нового підходу буде реалізувати таку ситуацію в площині технології блокчейн. При цьому повинна виконуватися умова, що сутність користувача, тобто зв'язок між його ідентичністю та публічними ідентифікаторами, буде відомий лише на період взаємодії з ним в приватній мережі. Таким чином, при його переході в основний блокчейн, він знову набуває ананонімності та може заперечувати свою належність до будь-якого вторинного блокчейну. Для реалізації цього нам допоможуть існуючі механізми в технології Монеро. До цієї криптовалюти, де приховання ідентичності користувачів та методи забезпечення цього прописані в протоколі, повинні додатися інструменти генерації одноразових публічних ідентифікаторів для участі у вторинній мережі.

Приватний та публічний блокчейни по своєму дизайну та архітектурі

були створені вирішувати різні задачі: приватна мережа – для деякої закритої групи компаній, що не викривають загалу ніякої внутрішньої інформації; публічний блокчейн (наприклад, Монеро) - навпаки. Опишемо наступну бізнес-ситуацію, яка поєднає обидві ці вимоги. Компанії IBM, Intel, Microsoft та деякі інші бажають створити унікальний сумісний проект, проте хочуть максимально приховати це від інших компаній, щоб уникнути конкуренції, не виказувати ідею проекту загалу, а коли реалізувати її – посісти провідні місця в своїй галузі. Всі вони мають облікові записи в Монеро й передбачену базовим протоколом змогу обмінюватися публічними ідентифікаторами, щоб утворити довготривалі відносини один з одним.

У цьому підході будемо використовувати вже розглянутий в даній роботі Hyperledger Sawtooth – децентралізовану платформу рішень, що володіє потрібними нам властивостями:

– наявна опція вмикати режим приватного блокчейну;

– можливість формувати власну політику доступу в мережі та потім змінювати її при спільній згоді всіх її учасників;

– динамічність компоненти, що включає той чи інший алгоритм консенсусу, в залежності від потреб клієнтів;

– приховання транзакційної діяльності від публічної спільноти

– швидкодія транзакцій;

– знижена вартість обміну коштами;

– гнучкі процесори транзакцій з можливістю написання різного типу смарт-контрактів, де міститься бізнес-логіка проекту.

Очевидно, що головною передумовою ініціації мережі є спільна домовленість групи користувачів об'єднатися та обмінюватися коштами в середині приватної мережі на узгоджених засадах, що будуть формувати початкову конфігурацію мережі.

Як і в кожному угрупуванні, потрібно щоб його хтось ініціював. Перший вузол з основної мережі буде відповідати за конфігурацію наступних його компонент приватного блокчейну:

– *Політику дозволів*, що компонує перелік атрибутів, які характеризують права кожного конкретного учасника даної мережі. На основі цих даних визначаються конкретні дозволи та обов'язки вузла, що приєднується до системи. Серед дозволів можуть бути права на отримання повних даних з блокчейну, створення транзакцій певного типу тощо.

– *Обрання вузлів-валідаторів*, відповідальних за зчитування та оновлення загального стану блокчейну. Вони обираються серед потенційних учасників системи за заздалегідь укладеною домовленістю.

– *Компонування та реалізацію смарт-контрактів* – програм, які інкапсулюють в собі загальну бізнес-логіку системи. Це може бути один чи декілька лістингів коду, що виконуються верифікаційними вузлами під час обробки транзакцій в мережі та інших специфічних для них операцій.

Певним чином такий підхід означає започаткування інфраструктури поверх основного блокчейну, що володіє елементами централізації – того, що поняття блокчейну першочергово прагне звести до нуля. Той, хто налаштовує блокчейн з обмеженим доступом, має перевагу у визначенні того, як насправді в подальшому буде виглядати ця підмережа, які там дозволені операції та, найголовніше, хто зможе долучитися до цієї системи. Проте саме цього і прагне приватне угрупування компаній – готової платформи, до якої потрібно лише долучитися, використовувати її відповідно до своїх вимог та брати участь в різного роду голосуваннях у прийнятті спільних рішень. І з цієї точки зору вже не важливо, хто саме створив під-мережу, адже її можна сконфігурувати так, що певна група нових учасників, що приєднуються, будуть мати такі ж права, як і в першого вузла. Все залежить від побажань клієнта. Кожен може відмовитися долучатись до каналу, політика якого не є задовільною. Тому в інтересах всіх учасників грати, дотримуючись правил, і створювати те, що буде влаштовувати всіх в об'єднанні.

Алгоритм приєднання та обробки транзакцій користувача при гібридному підході, відображений на Рисунку 4.1:

– *Етап блокування коштів в основному блокчейні* Bch_{main}.

1) Користувач X блокчейну Bch_{main} створює транзакцію Tr_{lock_main} для блокування частини своїх коштів. У ній фігурують X як відправник, а обрана приватна мережа як отримувач. Підкреслимо, що ці дані "затемнені" завдяки протоколу Монеро.

2) Майнери, згідно алгоритму PoW, верифікують її дійсність та при успішному результаті – додають до одного з нових блоків. Серед всіх інших перевірок, відбувається верифікація суми в транзакції на відповідність до лімітів, передбачених протоколом основного блокчейну як його вбудованих параметрів. Особливу увагу треба звернути на те, що голосування серед вузлів Bch_{main} на цьому кроці не відбувається, натомість воно буде проведено пізніше, для транзакції розблокування коштів.

3) Майнери Bch_{main} повідомляють отримувача – приватну мережу $Bch_{secondary}$ про бажання користувача приєднатися до неї. Якщо цей запит буде відхилено, то автоматично формується транзакція про розблокування заявлених раніше коштів, деталі якої описані нижче (пункти 11-13).

– *Етап розблокування коштів у вторинному блокчейні.*

4) Створюється специфічна транзакція Tr_{join} вже на стороні вторинного блокчейну з обмеженим доступом $Bch_{secondary}$, що знаменує приєднання користувача X до цієї мережі. В транзакції міститься інформація про його поточний рахунок, t_{lock} – мітку часу запиту транзакції блокування коштів в блокчейні Bch_{main} та іншу мета-інформацію. Надалі саме це буде початковим джерелом правди щодо його балансу в мережі $Bch_{secondary}$.

5) Відбувається процес ще одного досягнення консенсусу, проте вже серед вузлів блокчейну $Bch_{secondary}$. При несхваленні, формується транзакція на розблокування коштів в основній мережі.

6) Якщо згодом дана транзакція успішно підтверджується більшістю, вона оновлює загальний стан $Bch_{secondary}$ і всі вузли з ним синхронізуються. На цьому кроці визначаються дозволи та обов'язки X,

якими цей користувач зможе оперувати, а його кошти рахуються розблокованими в $Bch_{secondary}$ у вигляді еквівалентних токенів.

– *Здійснення транзакційної діяльності.*

7) Після розблокування коштів X, користувач може здійснювати моментальні угоди (Tr_X), які будуть верифікуватися іншими користувачами $Bch_{secondary}$, акцентуючи увагу на стані його балансу та мітці часу t_{lock}, що вказує, допоки ці кошти можуть бути використані в цій мережі.

– *Блокування коштів у вторинному блокчейні.*

8) При настанні однієї з наступних умов: мітка часу t_{lock} має значення, що перевищує встановлений ліміт (рахунок X при цьому стає неактивним) або користувач X сам прагне покинути вторинну мережу – створюється нова спеціальна транзакція Tr_{detach} з метою заблокування коштів у блокчейні $Bch_{secondary}$.

9) Повторюється стандартний процес верифікації цієї операції, генерація нового блоку та досягнення узгодження серед вузлів.

– *Розблокування частини балансу рахунку користувача X блокчейні* Bch_{main}.

10) Вторинний блокчейн повідомляє основну мережу про наявність запитів на повернення частини раніше заблокованих коштів X в Bch_{main}.

11) Відбувається підрахунок кінцевого стану частини рахунку користувача X з урахуванням історії його успішних операцій в $Bch_{secondary}$.

12) В блокчейні Bch_{main} створюється ряд транзакцій ($Tr_{X1}, Tr_{X2}...$), що відповідають за виплати коштів користувачам, які мали обмін грошима з X в $Bch_{secondary}$. Кожна така транзакція є результатом об'єднання повторних транзакцій між X та іншим користувачем. В останню чергу генерується специфічна транзакція Tr_{unlock} для розблокування залишку користувача, якщо він залишився.

13) Майнери проводять процес верифікації транзакцій та додають їх в новий блок. Після, всі вузли голосують за прийняття даного блоку і зміну

глобального стану, та через деякий час мережа приходить до загального узгодження.

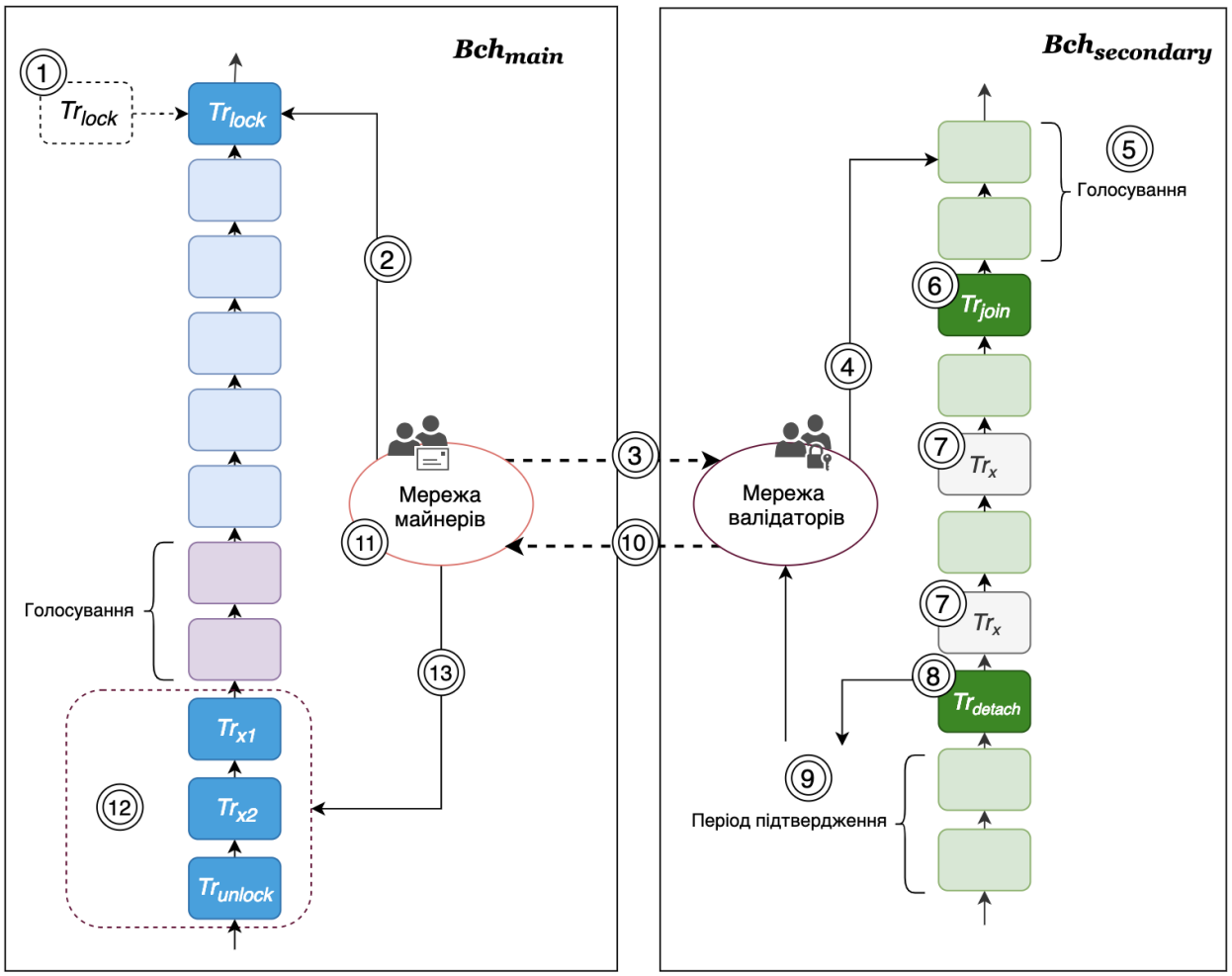

Рисунок 4.1 – Структура і процес обробки транзакцій у гібридному підході

Ліміти на суми коштів, якими можуть оперувати користувачі при переході у вторинний блокчейн, та перевірка міток часу відкриття там рахунків існують для того, щоб запобігти ситуації масового блокування монет в мережі, що могло б її дестабілізувати. Конкретні значення лімітів можуть змінюватися динамічно в залежності від багатьох факторів: кількості користувачів у публічній мережі, собівартість базової монети, активності учасників (об'єм транзакцій на секунду) як в базовій, так і вторинній мережі, навантаження на вторинну мережу в сенсі загальної

кількості вкладників та їх коштів тощо.

Слід окремо відзначити, що для підтримки працездатності усієї системи повинні бути присутні не тільки кількісні обмеження при обробці транзакцій, а й процес примусової провокації ”закриття” зв’язку з вторинним блокчейном при настанні неприйнятних умов для базової мережі. Наприклад, це може статися наступним чином: час блокування коштів користувача в основній системі добіг кінця, проте він не проявив ініціативу до їх розблокування, тоді валідатори при верифікації вторинної мережі виявлять, що рахунки деяких користувачів неактивні, адже їх мітки часу на вході в блокчейн уже застарілі. Тоді для даної групи вони формують транзакцію виходу з приватного блокчейну з урахуванням усіх угод за час активного рахунку кожного. Крім того це є й запитом на розблокування решти або всіх коштів в основному блокчейні. Таким чином формується пакет вихідних з приватного блокчейну транзакцій. Про пакет цих транзакцій, що зовні схожі на всі інші в мережі Монеро, повідомляється майнерами, що її верифікують та згодом додають до блоку, що врешті-решт змінить загальний стан блокчейну.

Також слід більш детально розглянути нове поняття токенів, яке вводиться у вторинному блокчейні. Вони становлять еквіваленті до основних монет гроші, проте можуть бути більш дрібними. Це дозволяє створювати так звані мікро-платежі, що робить транзакційну діяльність користувачів більш гнучкою, враховуючи порівняно постійний ріст ціни на основну монету в фіатній валюті.

Важливо зазначити, що кожен раз, коли один і той же користувач блокує гроші в основному блокчейні та переходить у вторинний, верифікація його ідентичності та балансу на тимчасовому рахунку повторюється кожен раз.

Окремо зауважимо, що звичайний процес обробки транзакцій блокування та розблокування коштів, що базується на протоколі PoW. Розглянемо ці два етапи у випадку гібридного підходу:

1) *Блокування коштів.* Майнери верифікують дані транзакції та

додають її до нового блоку, після чого вона автоматично публікується в блокчейн – без етапу голосування спільноти, адже записується лише факт блокування частини коштів користувача, не реальний грошовий переказ. З іншої сторони, ця операція означає тимчасову передачу контролю над цими грошима приватній підмережі.

2) *Розблокування коштів.* Після того, як валідатори вторинної мережі повідомляють основний блокчейн про запит на розблокування коштів, майнери підхоплюють одну чи список транзакцій, що утворилися в результаті діяльності користувача в приватному блокчейні. Вони їх усі верифікують та кожну додають до нового блоку. На даному етапі вже присутнє голосування серед вузлів основної мережі. При цьому, кожна така транзакція повинна обов'язково отримати й голос від свого відправника, щоб уникнути випадків помилково сформованих транзакцій вторинним блокчейном.

4.3 Другий підхід: застосування мережі Lightning

Тепер приведемо приклад іншого бізнес випадку, коли користувачі прагнуть побудувати довготривалі відносини один з одним, не виказуючи свої ідентичності. Для цього використаємо новітню технологію – мережу Lightning, що буде працювати на основі криптовалюти Монеро.

Перерахуємо переваги мережі Lightning, якими буде доповнена криптовалюта Монеро:

– зменшення обсягів інформації про фінансову діяльність користувачів, відомої загалу;

– швидкодія транзакцій, тобто змога проводити миттєві операції;

– наявність мікро-транзакцій – утворення проміжних токенів;

– знижена вартість транзакцій.

Розпочнемо огляд структури та процесів застосування мережі Lightning для Монеро з поняття рахунку користувача. Він містить стандартні поля: мітка часу створення цього рахунку, термін його дії, ідентифікатор екземпляру мережі Lightning, до якого він приєднався, тощо. Та на відміну від традиційних рахунків мережі Lightning, в ньому немає посилання на резервну публічну адресу користувача в основній мережі та змоги пропонувати свої платіжні маршрути, остільки ідентифікатори всіх вузлів приховані протоколом Монеро.

Робота вузла Монеро з мережею Lightning розпочинається з встановлення в нього смарт-контракту – програмного коду, що допоможе полегшити його інтеграцію з новою технологією та буде опрацьовувати в ній транзакції, описувати правила перевірки валідності обміну, визначати алгоритм акумулювання однотипних транзакцій та політику штрафів, що будуть накладатися на недобросовісних учасників, які порушують правила обміну коштами в мережі. Смарт-контракти можуть реалізовувати більш складну бізнес логіку, ніж простий обмін коштами, тому бувають різного типу. Безпосередньо код їх програм публікується для кожного різновиду мереж Lightning окремо і перебуває у публічному доступі для забезпечення прозорості своєї системи.

Опишемо процес приєднання користувача до мережі Lightning, при умові, що він має обліковий запис в блокчейні Монеро. Варто зауважити, що етапи блокування та розблокування коштів в основному блокчейні відбуваються подібно до цих процесів у гібридному підході, який описаний в попередньому підрозділі (4.2). Єдиною відмінністю є те, що при формуванні транзакції для блокування коштів вказується публічний ідентифікатор не приватного блокчейну, а обраного екземпляра мережі Lightning. Тепер перейдемо безпосередньо до розгляду алгоритму:

1) Блокування частини коштів m користувача X в основному блокчейні Bch_{main}.

2) Встановлення смарт-контракту обраного екземпляру мережі Lightning (LN).

3) Розблокування еквівалентної до m кількості токенів в мережі LN. З цього моменту вузол X займається і моніторингом Bch_{main}, і перевіркою даних своїх контрагентів під час обміну коштами.

4) Виконання смарт-контрактів при будь-якій операції в мережі LN.

5) Вихід з системи в активний чи пасивний спосіб. Останній передбачає собою ситуацію, коли термін дії рахунку LN добігає кінця або користувач діє проти правил протоколу мережі та умисно чи ні чинить шахрайство. Тоді автоматично формується заявка на вихід з системи. При активному способі користувач сам є ініціатором закриття каналу LN, де він обмінювався коштами. В обох випадках, на виході з мережі токени блокуються, аналізуються всі перекази між X й іншими учасниками та формуються список кандидатів на транзакції блокчейну Bch_{main}.

6) Основна мережа отримує пакет транзакцій користувача X, виконує їх та розблоковує при цьому решту, що залишилася від m після виходу з LN.

Щоб зобразити запропонований підхід, потрібно визначити набір дій, які користувач може виконати в даній системі, а реалізація протоколу вирішить правила для дій, які необхідно вжити, і типів операцій, які необхідно побудувати, для кожної такої операції:

– $create_i(T)$: учасник u_i створює транзакцію T та зберігає її локально;

– $sign_i(T)$: учасник u_i підписує транзакцію T;

– $broadcast_i(T)$: транзакція T відправляється всім учасникам n-каналів;

– $deliver_i^j$(T): транзакція T відправлена від користувача u_i до учасника u_j;

– $publish_i(T)$: транзакція T публікується учасником u_i в ланцюгу та вона зберігається в блокчейні.

Для для забезпечення надійності нового підходу потрібна наявність певних обмежуючих параметрів в протоколі основного блокчейну, які будуть обмежувати діяльність користувачів в мережі Lightning:

– Кількість транзакцій в одному екземплярі мережі Lightning, призначених одному й тому ж вузлу. Це обмеження накладається на обидві сторони: відправника та отримувача, щоб лімітувати використання їх одноразових публічних ключів.

– Пропускну спроможність кожного вузла, щоб запобігти ситуації централізації інформації в маршрутизаторах, тобто навіть про проміжні платежі.

– Кількість активних вузлів в мережі Lightning на всіх його екземплярах. Обмежуючи їх кількість, зменшується обсяги витоку інформації про платежі користувачів у випадку, коли зловмисник захоче створити якомога більше таких вузлів для збору цих даних.

– Час активності рахунку користувача в мережі Lightning, що тотожно визначенню періоду блокування його коштів в основному блокчейні.

Єдиною точкою системи, де більше всього накопичується інформації про певні однотипні платежі, що здійснюються в мережі Lightning, це канал, в якому повторно використовуються одноразові публічні ключі користувача. Проте через те, що канал ділять лише два користувачі, де другий вузол швидше за все є тільки маршрутизатором платіжок, – у такому випадку вірогідність якісно прослідкувати за фінансовою активністю досить мізерна.

З іншого боку, при використанні мережі Lightning відбувається реорганізація процесу обробки платежів та зміна влади над цим. Відтепер опрацювання мікро-транзакцій це відповідальність не майнерів, а звичайних вузлів, які виконують роль маршрутизаторів. Так, як постійно зростають витрати на генерацію блоків в криптовалютах на основі PoW, майнери змушені об'єднуватися в так звані майнінг-пули, які вносять чималу централізацію в блокчейн. Тому новий підхід, крім всього іншого, сприяє децентралізації всієї системи.

4.4 Порівняння характеристик запропонованих та існуючих підходів

Щоб якісно оцінити існуючі та нові підходи, порівняємо окремі їх характеристики. Для цього візьмемо найбільш поширені криптовалюти на основі протоколу PoW та розглянуті в цій роботі блокчейни з обмеженим доступом. Результати приведені в Таблиці 4.1.

Обидва нові підходи – використання блокчейну з обмеженим доступом та інтеграція з мережею Lightning – можуть бути вбудовані в існуючі блокчейни одночасно. Вони, на приклад, потенційно мають змогу функціонувати на основі багатьох криптовалют, також орієнтованих на забезпечення приватності даних користувачів. Крім Монеро, іншим яскравим прикладом таких технологій є Zcash, що використовує алгоритм доказу нульового розголошення для приховання ідентичностей та сум переказів в транзакціях.

Підсумовуючи плюси і мінуси кожного методу захисту конфіденційності даних в складній системі блокчейн, яка повинна задовольняти вимоги щодо приватності користувачів з бажаними властивостями, хотіли б зробити наступні три зауваження. По-перше, жодна технологія не є панацеєю для забезпечення конфіденційності в блокчейні. Таким чином, відповідні методи захисту приватних даних користувачів повинні обиратися на основі вимог клієнтів та контекст застосування. Загалом, поєднання декількох технологій працює більш ефективно, ніж використання однієї технології. По-друге, не існує технології, яка не має дефектів або є досконалою у всіх аспектах. Коли новий підхід додається до складної системи, він завжди викликає появу інших проблем або нових форм атак. Тому вимагається уважне ставлення до потенційних збитків, спричинених інтеграцією деяких методів безпеки в блокчейні. По-третє, завжди існує компроміс між забезпеченням

конфіденційності даних та ефективністю. Потрібно просувати ті алгоритми та підходи, які поліпшують безпеку блокчейну і в той же час сприяють його розвитку та можливості практичного розгортання таких додатків з прийнятною продуктивністю.

Таблиця 4.1 – Порівняння характеристик технологій на основі алгоритму PoW та блокчейнів з обмеженим доступом

	Біткойн	Біткойн з мережею Lightning	Монеро	Dash	Zcash	Ethereum	Hyperledger Fabric	Hyperledger Sawtooth	Використання "Гібридного підходу" на основі Монеро
Тип	Публічний блокчейн	Публічний блокчейн	Публічний блокчейн	Публічний блокчейн	Публічний блокчейн	Публічний блокчейн	Фреймворк	Фреймворк	Публічний блокчейн з елемантами приватної мережі
Незмінний	✓	✓	✓	✓	✓	✓	✓	✓	✓
Можна вести аудит	✓	✓	✓	✓	×	✓	✓	✓	✓
Має обмежений доступ	×	×	×	✓	×	×	✓	✓	×/✓
Рівень анонімності користувачів	низький	низький	високий	низький	високий	низький	низький	низький	високий
Рівень приховання даних транзакцій	низький	середній	високий	низький	високий	низький	низький	низький	високий
Рівень підтримки та модифікацій налаштувань	дуже низький	дуже низький	низький	низький	низький	середній	високий	високий	високий
Алгоритм консенсусу	PoW	PoW	PoW	PoW+PoS	PoW + Zk-snark	PoW	PBFT	PoET	PoW
Має підтримку смарт-контрактів	лімітовані (scripts)	лімітовані (scripts та смарт-контр акти мережі Lightning)	Ні	Ні	Ні	повнота за Тюрингом	Так ("chaincode")	Так ("chaincode")	Так ("chaincode")
Алгоритм генерування гешу	SHA-256	SHA-256	CryptoNight	X11	Zero Knowledge Proof	Ethash	*	*	CryptoNight (*)
Час генерації блоку	10 хв	10 хв	2 хв	2.5 хв	2.5 хв	16 сек	5 сек*	20 сек*	2 хв (20 сек**)
Час підтвердження транзакції	1 година	1 година	20 хв	30 хв	15 хв	3 хв	миттєво	миттєво	20 хв (миттєво**)
Рівень масштабованості	низький	середній	високий	низький	низький	середній	високий	високий	високий
Використання алгоритму доказу нульового розголошення	×	×	✓	×	✓	×	-	-	✓
Вірогідність атаки 51%	низька	низька	середня	середня	середня	низька	низька	низька	середня
Стійкість до DDoS	✓	✓	✓	✓	✓	✓	×	×	✓
Візантійська відмовостійкість (BFT)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Квантова стійкість	×	×	✓	×	×	×	×	×	✓
Можливість випускати власні токени	×	✓	×	×	×	✓	✓	✓	✓
P2P комунікація	✓	✓	✓	✓	✓	✓	✓	✓	✓
Можливість мультипідпису	✓	✓	✓	✓	×	✓	✓	✓	✓
Вартість транзакцій	висока	середня	середня	середня	середня	середня	0	0	нижче середньої
Підтримка мікротранзакцій	×	×	×	×	×	×	✓	✓	✓
Підтримка регулярних платежів	×	×	×	×	×	×	✓	✓	✓

Зазначимо декілька приміток до Таблиці 4.1:

* – характеристика може змінитися або конфігурується клієнтом.

** – у випадку приватного блокчейну або застосування мережі Lightning.

Практичні реалізації гібридного підходу та використання мережі Lightning на основі криптовалюти Монеро (або її альтернативи, також орієнтованої на забезпечення приватності користувачів) дадуть більш широке розуміння про їх ефективність. Проте це потребує модифікації існуючого протоколу Монеро та її дизайну, тобто створення нової криптовалюти як форку від неї. Крім того, необхідно реалізувати компоненти з приватним блокчейном та мережею Lightning у вигляді незалежних модулів, які будуть легко інтегруватися з основним блокчейном. Тож, для цього необхідні чимала фінансова підтримка та тривала розробка.

Висновки до розділу 4

У даному розділі було запропоновано декілька нових підходів, які доповнюють існуючі рішення на основі блокчейну, удосконалюючи та роблячи його більш універсальним. Деякі користувачі хотіли б отримати переваги блокчейну, проте мати його доступним лише для обмеженої групи людей. Так з'явилося поняття приватних блокчейнів з обмеженим доступом. Є й інший бізнес випадок, коли користувачі потребують і публічної звітності і миттєвого опрацювання транзакцій. Щоб вирішити дану задачу, було спроектовано мережу Lightning. Обидві технології були використані для формування нових методологій для задоволення однієї з найбільш важливих вимог користувача – надійності мережі. Поєднання приватності та публічності блокчейнів та інших видів децентралізованих

мереж може створити саме той симбіоз технологій, що вирішить проблеми багатьох груп користувачів, які прагнуть публічно обмінюватися коштами з іншими учасниками, при цьому не виказуючи деякі перекази (сторонню діяльність) загалу. Саме для цього можуть бути використані нові підходи, що базуються на блокчейні з обмеженим доступом та мережі Lightning.

Потрібно пам'ятати, що описані модулі не обов'язково використовувати на початку роботи з основним блокчейном, дані додатки – це незалежні компоненти, що відповідають за задоволення різних бізнес-цілей, тому інтегруються по бажанню користувача окремо.

ВИСНОВКИ

Сьогодення вимогливе до технічних рішень, в тому числі в питанні приватності та анонімності даних, а особливо коли справа стосується фінансової діяльності. Технологія блокчейн і справді революційна та дає змогу не модифікувати старі підходи, а подивитися на проблему взаємовідносин користувачів та сторонніх інстанцій зовсім під іншим кутом. Раніше задача була в тому, як для своєї системи знайти найбільш надійних постачальників ресурсів, регуляторів та верифікаторів, зараз – яку форму технології блокчейн застосувати, адже його спільнота ефективно замінює всі перераховані ролі в системі. Саме тому написана ця робота: щоб розвивати та адаптувати блокчейн під теперішні виклики суспільства. Для цього не потрібно будувати модель довершеної технології, на практиці під час реалізації все одно прийдеться чимось жертвувати на перевагу пріоритетним характеристикам. Тож було обрано синтез адаптивних підходів, що передбачав:

– виокремлення бізнес-стратегій та сфер, в яких блокчейн не може використовуватися чи застосовуються тільки частково через певні ліміти технології;

– узагальнене та детальне оформлення вимог користувачів вищевказаних систем;

– побудову нових моделей блокчейну, які в тих чи інших умовах будуть забезпечувати максимум вимог користувачів.

У даній роботі проведено огляд поточних проблемних областей технології блокчейн, зокрема на тему конфіденційності даних. Було підготовлено перелік потенційних рішень на тему дослідження та більш детально розглянуто криптовалюту Монеро з деякими її модифікаціями та доповненнями.

Незважаючи на вже реалізовані в Монеро алгоритми збереження приватності, є принаймні 4 різних аналізи, розроблених для виявлення

прихованої інформації в середовищі даної криптовалюти. Ці дослідження були успішно здійснені завдяки прозорості даних у блокчейні, а також проблемі ліквідності та ідентифікації поведінки користувачів. Тож, Монеро не можна назвати цілком приватною мережею. Проте потрібно постійно наближатися до цього, причому крім того зробити так, щоб система одночасно працювала ефективно та при цьому задовольняла ширше коло вимог користувачів.

Гібридний підхід та використання мережі Lightning у поєднанні з криптовалютою Монеро утворили платформу, що може ефективно реалізувати широке коло бізнес-проектів та забезпечити приватність користувачів та даних, якими вони маніпулюють. Нові підходи орієнтовані не лише на збереження конфіденційності своїх клієнтів, але й на удосконалення технології блокчейн загалом, а саме: підвищення децентралізації мережі, швидкодію транзакцій, можливість мікро-операцій тощо.

Важливо пам'ятати, що поняття приватності не статична та фіксована річ. Це ціль, за яку постійно ведеться боротьба між криптографами та зловмисниками. Абсолютної приватності не існує. Тому наша задача – адаптуватися до технологічного середовища, яке змінюється, та пропонувати нові алгоритми й підходи, що будуть забезпечувати надійність та ефективність роботи систем.

ПЕРЕЛІК ПОСИЛАНЬ

1. Bitcoin and Cryptocurrencies: Law Enforcement Investigative Guide [Електронний ресурс]. — Режим доступу: http://www.iacpcybercenter.org/wp-content/uploads/2018/03/Bitcoin.pdf.

2. Mt. Gox [Електронний ресурс]. — Режим доступу: https://en.bitcoin.it/wiki/Mt._Gox.

3. Silk Road: A Cautionary Tale about Online Anonymity [Електронний ресурс]. — Режим доступу: https://medium.com/@marcell74/the-silk-road-a-real-thriller-and-the-truth-about-the-anonymity-of-bitcoin-198b519ca3

4. Elliptic: Detect And Prevent Criminal Activity In Cryptocurrency [Електронний ресурс]. — Режим доступу: https://www.elliptic.co/.

5. Ethereum Project [Електронний ресурс]. — Режим доступу: https://www.ethereum.org/.

6. CoinJoin [Електронний ресурс]. — Режим доступу: https://en.bitcoin.it/wiki/CoinJoin.

7. CoinJoin Security Research [Електронний ресурс]. — Режим доступу: https://pdfs.semanticscholar.org/0325/0ef8dfa44bb26a43df0e7e846324286e35e5.pdf.

8. What's a Sybil Attack & How Do Blockchains Mitigate Them? [Електронний ресурс]. — Режим доступу: https://coincentral.com/sybil-attack-blockchain/.

9. Monero: Privacy in the blockchain [Електронний ресурс]. — Режим доступу: https://eprint.iacr.org/2018/535.pdf.

10. CryptoNote [Електронний ресурс]. — Режим доступу: https://cryptonote.org/.

11. Monero Ring Attack: Recreating Zero Mixin Transaction Effect [Електронний ресурс]. — Режим доступу: https://eprint.iacr.org/2018/348.pdf.

12. A Traceability Analysis of Monero's Blockchain [Електронний

ресурс]. — Режим доступу: https://eprint.iacr.org/2017/338.pdf.

13. On the (Im)possibility of Obfuscating Programs [Електронний ресурс]. — Режим доступу: https://www.iacr.org/archive/crypto2001/21390001.pdf.

14. Indistinguishability Code Obfuscation research [Електронний ресурс]. — Режим доступу: https://eprint.iacr.org/2013/451.pdf.

15. Intel Software Guard Extensions [Електронний ресурс]. — Режим доступу: https://software.intel.com/sgx.

16. Ekiden: A Platform for Confidentiality-Preserving, Trustworthy, and Performant Smart Contract Execution [Електронний ресурс]. — Режим доступу: https://arxiv.org/abs/1804.05141.

17. S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system,"2008, [Електронний ресурс]. — Режим доступу: https://bitcoin.org/bitcoin.pdf.

18. Proof of work[Електронний ресурс]. — Режим доступу: https://en.bitcoin.it/wiki/Proof_of_work.

19. N. van Saberhagen, "Cryptonote v 2. 0,"2013. — Режим доступу: https://downloads.getmonero.org/whitepaper_annotated.pdf. S. Meiklejohn et al., "A Fistful of Bitcoins: Characterizing Payments Among Men with No Names,"USENIX ;login:, 2013.

20. A. Kumar, C. Fischer, S. Tople, and P. Saxena, "A Traceability Analysis of Monero's Blockchain,"IACR Cryptology ePrint Archive, vol. 2017, p. 338, 2017.

21. A. Miller, M. Möser, K. Lee, and A. Narayanan, "An Empirical Analysis of Linkability in the Monero Blockchain,"arXiv preprintarXiv:1704.04299, 2017.

22. RingCT [Електронний ресурс]. — Режим доступу: https://ww.getmonero.org/resources/moneropedia/ringCT.html.

23. A note on fees [Електронний ресурс]. — Режим доступу: https://ww.getmonero.org/2017/12/11/A-note-on-fees.html.

24. Monero Avg. Transaction Fee historical chart [Електронний ресурс]. — Режим доступу: https://bitinfocharts.com/comparison/

monero-transactionfees.html.

25. What is a SYN flood attack [Електронний ресурс]. — Режим доступу: https://www.imperva.com/learn/application-security/syn-flood/.

26. Kovri [Електронний ресурс]. — Режим доступу: https://www.getmonero.org/resources/moneropedia/kovri.html.

27. Проект невидимый интернет (I2P) [Електронний ресурс]. — Режим доступу: https://geti2p.net/ru/about/intro.

28. Hyperledger Sawtooth [Електронний ресурс]. — Режим доступу: https://sawtooth.hyperledger.org/docs/core/releases/1.0/introduction.html.

29. Corda Network [Електронний ресурс]. — Режим доступу: https://www.r3.com/.

30. Hyperledger Fabric [Електронний ресурс]. — Режим доступу: https://www.hyperledger.org/projects/fabric.

31. Hyperledger Burrow [Електронний ресурс]. — Режим доступу: https://www.hyperledger.org/projects/hyperledger-burrow.

32. What is Nakamoto Consensus? Complete Beginner's Guide [Електронний ресурс]. — Режим доступу: https://blockonomi.com/nakamoto-consensus/.

33. Byzantine Fault Tolerance [Електронний ресурс]. — Режим доступу: https://en.wikipedia.org/wiki/Byzantine_fault.

34. Hash Functions, Merkle Trees and Radix Tries. Things to Know before Tackling "Merkle Tries" [Електронний ресурс]. — Режим доступу: https://medium.com/orbs-network/designing-a-state-database-blockchains-part-i-the-basics-90d814d6973b.

35. Secp256k1[Електронний ресурс]. — Режим доступу: https://en.bitcoin.it/wiki/Secp256k1.

36. Gossip protocol[Електронний ресурс]. — Режим доступу: https://en.wikipedia.org/wiki/Gossip_protocol.

37. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile [Електронний ресурс]. — Режим доступу: https://tools.ietf.org/html/rfc5280.

38. Everything You Need to Know About SSL Certificates [Електронний ресурс]. — Режим доступу: https://www.verisign.com/en_US/website-presence/online/ssl-certificates/index.xhtml

39. Apache Kafka [Електронний ресурс]. — Режим доступу: https://kafka.apache.org/.

40. Lightning Network: Scalable, Instant Bitcoin/Blockchain Transactions [Електронний ресурс]. — Режим доступу: https://lightning.network/.

41. Evaluating User Privacy in Bitcoin [Електронний ресурс]. — Режим доступу: https://eprint.iacr.org/2012/596.

42. The components of the Wired Spanning Forest are recurrent [Електронний ресурс]. — Режим доступу: https://link.springer.com/article/10.1007%2Fs00440-002-0236-0.

43. About Tor project [Електронний ресурс] . — Режим доступу: https://www.torproject.org/.

ДОДАТОК А

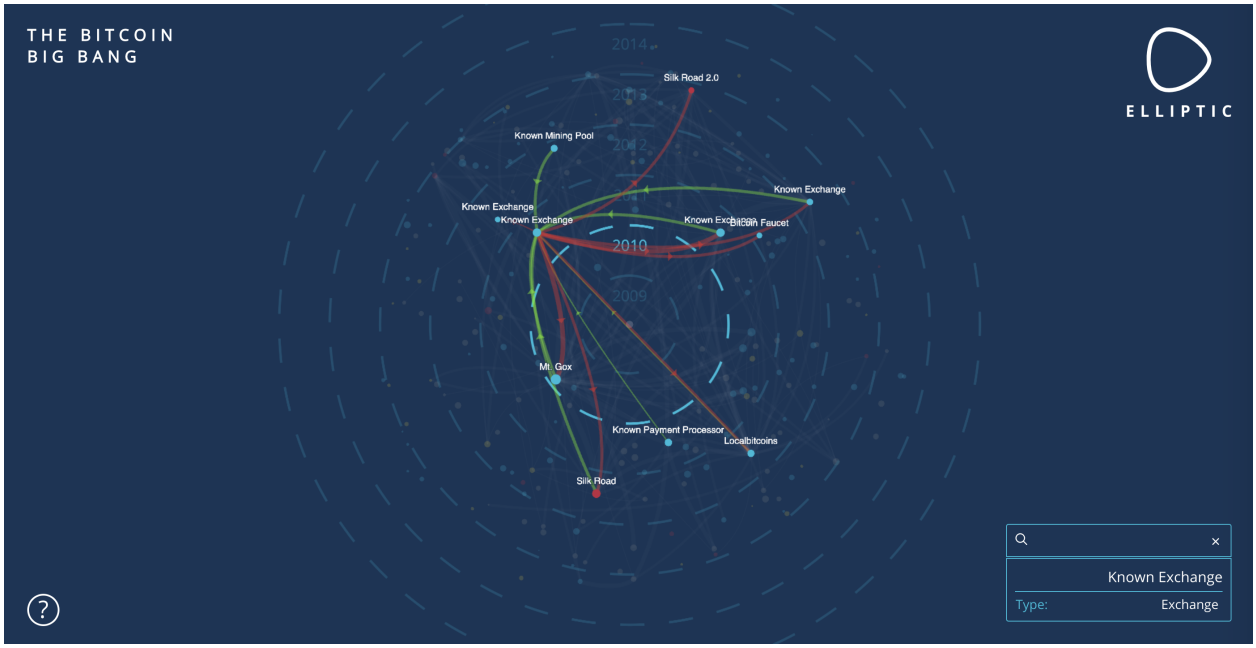

Рисунок А.1 – Графічне зображення результатів аналізу грошових потоків в біткойнах через перелік фінансових установ, що були отримані компанією Elliptic станом на 2010 рік