Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна

Довідник термінів і визначень технології блокчейн

Харків – 2020

УДК 004.031.43
П__

Рецензенти:

Г.А. Кучук – доктор технічних наук, професор кафедри обчислювальної техніки та програмування Національного технічного університету "ХПІ";

О. Г. Толстолузька – доктор технічних наук, старший науковий співробітник, професор кафедри теоретичної та прикладної системотехніки факультету комп'ютерних наук Харківський національний університет імені В. Н. Каразіна

Затверджено до друку рішенням Науково-методичної ради Харківського національного університету імені В. Н. Каразіна (протокол № __ від __.__.2020 р.)

П __ Довідник термінів і визначень технології блокчейн / авторів Кузнецов О.О., Полуяненк М.О., Краснобаєв В.А., Кошман С.О. – Харків : ХНУ імені В. Н. Каразіна, 2020. – __ с.

Довідник термінів і визначень технології блокчейн розроблено для студентів факультету комп'ютерних наук за спеціальністю «Кібербезпека». У довіднику докладно розглядаються основні терміни та визначення, надається тлумачення специфічних термінів притаманних блокчейн технології, докладно розглядаються найбільш популярні алгоритми консенсусу. Довідник містить важливі сучасні терміни і визначення та має на меті надати загальний оглядовий матеріал про основи блокчейн технології. Передбачається, що в результаті навчання студенти оволодіють початковими знаннями з дисципліни «Технології блокчейн», сформують уявлення про децентралізоване обробку та зберігання інформації .

УДК 004.031.43

ПЕРЕДМОВА

У 1991 році криптографами Стюартом Хаберт (Stuart Haber) і У. Скоттом Сторнеттом (W. Scott Stornetta) була представлена ідея пов'язаних, за допомогою криптографії, блоків в структуру даних в яку можливо лише додавати інформацію. Ідею було опубліковано в академічній статті «Як поставити мітку часу на цифровому документі» (How to Time-Stamp a Digital Document) [1]. Робота була сфокусована на документах з мітками часу, популярному варіанті використання технології блокчейн, навіть сьогодні.

Концепція сучасного блокчейна була запропонована у 2008-му році Сатоши Накамото [2]. Вперше реалізована вона була в 2009-му році в якості компонента криптовалюти – біткоіна. Саме блокчейн дозволив виключити з системи обороту біткоінів третю сторону – центральний сервер, банк або інший авторитетний орган. Завдяки даній технології стало можливим проводити обмін цінностей без необхідності довіри або центрального органу, що раніше здавалося неможливим, і що призвело до неймовірної популярності криптовалют.

Загальна капіталізація тільки деяких популярних криптовалют на сьогодні перевершує 100 мільярдів $ [3], а на піку популярності – у кінці 2017 року, тільки капіталізація біткоіна перевищувала 320 млрд $.

Але з падінням популярності криптовалюти, з її тіні почала проявлятися технологія блокчейн, що володіє потенціалом застосування не тільки у фінансовій сфері, а й у багатьох інших прикладних галузях.

Технологія блокчейн цікава тим, що пропонує новий варіант взаємин між суб'єктами діяльності, максимально виключає довіру між взаємодіючими сторонами або довіру до будь-якої третьої сторони. Основний потенціал технології полягає в її децентралізованому характері і здатності усування необхідності в довірі взаємодіючих сторін та ґрунтуючись лише на довірі до математики та реалізованим алгоритмам. Технологія блокчейн дозволяє вести реєстр, який заслуговує на довіру, тобто зміни у реєстрі не контролюються третіми особами – а тільки технологією, що використовується.

Термінологія, що використовується в блокчейн технологіях варіюється від однієї реалізації до іншої. У першої частині «Глосарій термінів» Довідника термінів і визначень технології блокчейн докладно розглядаються необхідні поняття та технології, що використовуються в блокчейн системах та має на меті надати загальний оглядовий матеріал про технологію блокчейн. У Довіднику дотримується наведеного глосарію використовуваних термінів, який в основному зібраний за рахунок посилань на існуючі роботи по блокчейн технологіям і термінам наведені в NISTIR 8202 «Огляд блокчейн технології» [4], Звіту дослідницької комісії ASC X9 [5], стандарту Німеччини «Валідація даних з використанням блокчейна» [6].

Ключовим аспектом технології блокчейна є визначення того, який користувач публікує наступний блок. Це вирішується шляхом реалізації однієї з багатьох можливих моделей консенсусу. В інклюзивних блокчейн мережах зазвичай існує безліч вузлів публікації, що конкурують одночасно за публікацію наступного блоку. Вони зазвичай роблять це, щоб виграти плату

за криптовалюту та / або транзакцію. Як правило, вони не довіряють користувачам, які можуть знати один одного тільки за їх публічними адресами. Кожен хто публікує вузол, швидше за все, мотивований прагненням до фінансової вигоди, а не добробутом інших вузлів публікацій або навіть самої мережі [4].

Консенсус є процедурою прийняття рішення. Його мета – забезпечити те, щоб всі учасники мережі погодили свій поточний стан після додавання нової інформації, блоку даних або пакета транзакцій. Іншими словами, консенсус-протокол гарантує те, що ланцюг вірний, і дає стимули для того, щоб учасники залишалися чесними. Це важлива структура для запобігання ситуації, коли хтось контролює всю систему, і вона гарантує те, що всі дотримуються правил мережі.

Можна виділити наступні основні ідеї, які закладені в механізми консенсусу [7]:

- Децентралізоване управління. Єдиний центральний орган не може забезпечити завершеність транзакції.
- Структура кворуму. Вузли обмінюються повідомленнями заздалегідь визначеними способами, які можуть включати етапи або рівні.
- Автентифікація. Цей процес надає засоби для перевірки особи учасників.
- Цілісність. Забезпечує перевірку цілісності транзакції (наприклад, математично за допомогою криптографії).
- Неспростовності. Надає засоби для перевірки того, що передбачуваний відправник дійсно відправив повідомлення.
- Конфіденційність. Гарантує, що тільки визначений одержувач може прочитати повідомлення.
- Відмовостійкість. Мережа працює ефективно і швидко, навіть якщо деякі вузли або сервери виходять з ладу або працюють повільно.
- Продуктивність. Враховує пропускну здатність, життєздатність, масштабованість та затримку.

В межах цих ідей існують значні відмінності між різними механізмами консенсусу. Ряд перерахованих вище параметрів реалізується за допомогою основних методів криптографії, які використовують математичні функції для забезпечення безпеки і конфіденційності. Ці методи включають симетричне і асиметричне шифрування і геш-функції.

Ключовою особливістю блокчейн технології є те, що немає необхідності в тому, щоб довірена третя сторона надавала стан системи – кожен користувач у системі може перевірити цілісність системи. Щоб додати новий блок в блокчейн систему, всі вузли з часом повинні прийти до спільної згоди, проте деякі тимчасові розбіжності можливі.

В інклюзивних блокчейн мережах модель консенсусу повинна працювати навіть у присутності, можливо, недобросовісних користувачів, оскільки ці користувачі можуть спробувати порушити або захопити ланцюжок

блоків. Для ексклюзивних блокчейн мереж можуть бути використані засоби правового захисту, якщо користувач діє зловмисно.

У деяких блокчейн мережах, наприклад, інклюзивних, не може існувати довіри між вузлами публікації. У цьому випадку може знадобитися узгоджена модель ресурсномістких процесів (час обчислень, інвестиції тощо) щоб визначити який учасник додає наступний блок до ланцюжка. Як правило, в міру підвищення рівня довіри зменшується потреба у використанні ресурсів в якості міри формування довіри. Для деяких ексклюзивних блокчейн реалізацій уявлення про консенсус виходить за рамки забезпечення достовірності блоків, але охоплює всі системи перевірок від пропозиції транзакції до її остаточного включення в блок.

Варто відзначити, що завдання розподіленого консенсусу не специфічна для блокчейн систем і має добре перевірені рішення для багатьох інших розподілених систем (наприклад, баз даних NoSQL) [8]. Навіть завдання консенсусу, в якому вузли можуть бути недобросовісними, – завдання візантійського консенсусу – вперше була сформульована в 80-х роках минулого століття, а методи її вирішення з’явилися в кінці 90-х.

Але блокчейн системи від попередніх напрацювань відрізняються умовами роботи мережі. У звичайних алгоритмах візантійського консенсусу у вузлів мережі є «особистості», що виражаються через цифрові підписи або подібні криптопримітиви, а сам список вузлів відомий заздалегідь або змінюється рідке, але передбачувано. У блокчейн системах все навпаки. Учасники мережі не тільки заздалегідь невідомі, але і можуть вільно підключатися або відключатися від мережі. При цьому блокчейн, будучи децентралізованою системою, що має певні властивості: стійкість до цензури (ніхто не може заборонити майнить криптовалюту) і об’єктивність (для визначення поточної версії реєстру транзакцій не потрібно довіра авторитетним джерелам – корінь довіри знаходиться в самому блокчейне). Із-за цього звичайні алгоритми візантійського консенсусу для блокчейн систем не підходять.

Як і всі розподілені системи, блокчейн системи стикаються з затримкою в мережі, помилками при передачі, помилками в програмному забезпеченні, лазівками в системі безпеки та хакерськими погрозами. Більш того, його децентралізований характер передбачає, що жодному з учасників системи не можна довіряти. Можуть з’явитися шкідливі вузли, а також різниця в даних через суперечливість інтересів.

Щоб протистояти цим потенційним помилкам, блокчейн система потребує ефективного механізму узгодження, щоб гарантувати, що у кожного вузла є копія дійсної версії реєстру. Традиційні механізми відмовостійкості, що стосуються певних проблем, можуть бути не в змозі повністю вирішити проблему, з якою стикаються розподілені та блокчейн системи. Потрібно універсальне комплексне рішення для забезпечення відмовостійкості.

У другий частині «протоколи консенсусу децентралізованих систем» довідника обговорюються кілька моделей консенсусу, а також найбільш поширений підхід до вирішення конфліктів.

ГЛОСАРІЙ ТЕРМІНІВ

Авторизація (Authorization) – в області комп'ютерної безпеки, це право, що надається користувачеві, яке підтримує доступ до цифрових об'єктів: фізичним, логічним, функціональним або контентним.

Адреса (Address) – короткий буквено-цифровий рядок, отриманий з відкритого ключа користувача з використанням геш-функції, з додатковими даними для виявлення помилок. Адреси використовуються для відправки і отримання цифрових активів.

Активи (Assets) – все, що може бути передано.

Алгоритм цифрового підпису на еліптичних кривих (Elliptic Curve Digital Signature Algorithm) – алгоритм з відкритим ключем для створення цифрового підпису, певний в групі точок еліптичної кривої. Використовуваний для підписання транзакцій в блокчейн протоколах.

Асиметрична криптографія (Asymmetric-key cryptograph) або *криптографія з відкритим ключем (Public-key cryptography)* – криптографічна система, в якій користувачі мають закритий ключ, що зберігається в секреті і використовується для генерації відкритого ключа (який є загальнодоступним). Користувачі можуть підписувати дані цифровим підписом за допомогою свого закритого ключа, і отримана підпис може бути перевірена будь-яким, хто використовує відповідний відкритий ключ.

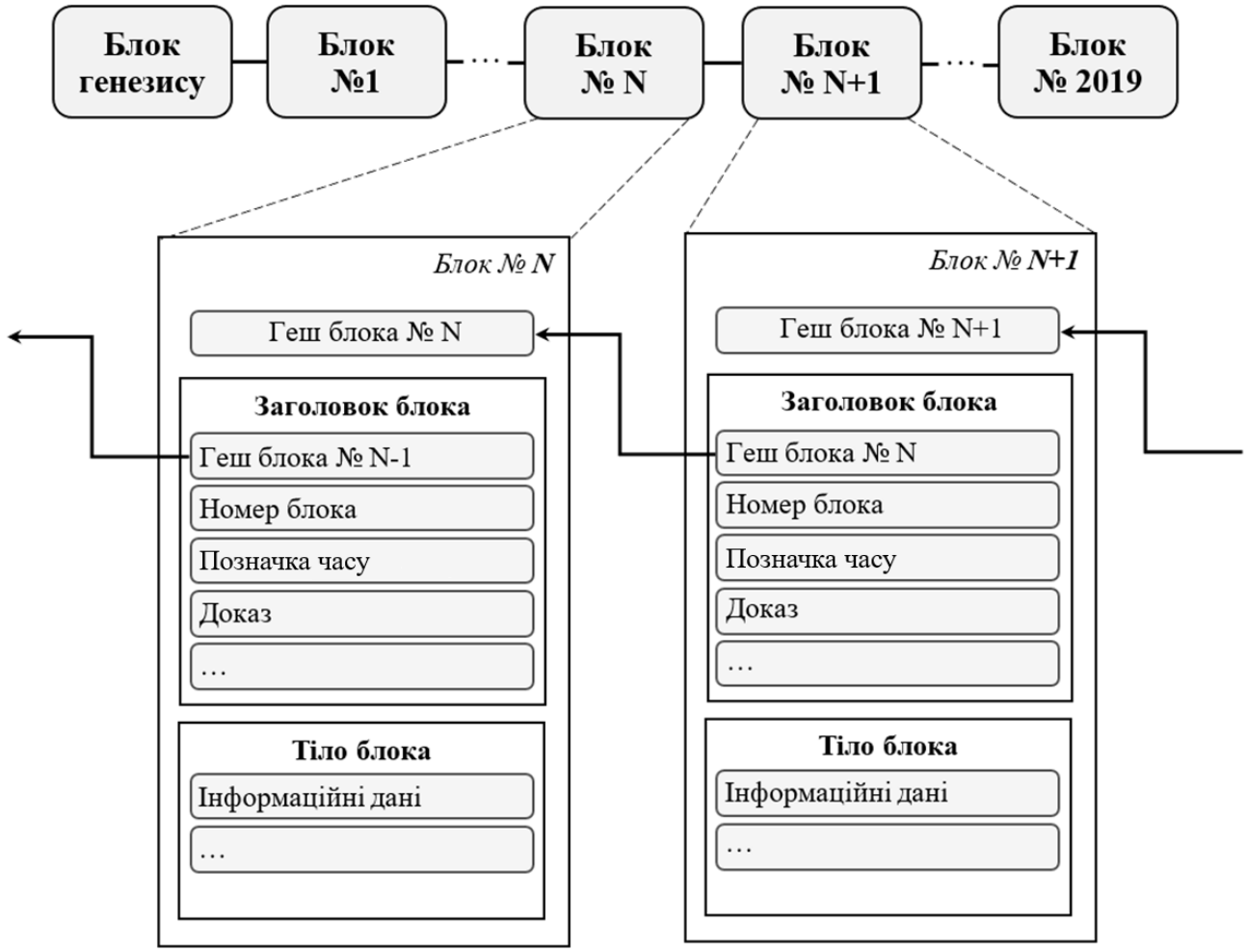
Аудируемість (Auditability) – можливість перевіряти в блокчейн системах дотримання правил і вести точний облік усіх транзакцій, процесів і дій.

Автентифікація (Authentication) – процес перевірки особистості користувача або сервера.

Безпечність (*Safety*) – здатність облікової системи зберігати основні принципи функціонування і інтереси чесних учасників за будь-яких злочинних діях.

Бічні ланцюги (Sidechains) – блокчейн системи, які взаємодіють з іншим блокчейном. Бічні ланцюги дозволяють блокчейн системам виконувати функції, які пов'язані або прив'язані до іншого запису, не перевантажуючи вихідну блокчейн.

Блок (Block) – структура даних, що містить заголовок блоку і дані блоку(див. рисунок 1).



Рисунку 1 – Узагальнена структура ланцюжку блоків блокчейн систем

Блок даних (Block data) – частина блоку, яка містить набір перевірених транзакцій і подій реєстру.

Блок застарілий (*stale blocks*) – називають дійсні блоки, причому успішно здобутий, які не є частиною основного ланцюга. Вони можуть з'являтися природним шляхом, коли два майнера виробляють два блоки практично одночасно або є згенерованими зловмисником з метою атаки на блокчейн. Застарілі блоки перевірені, їх походження відстежує до генезис блоку, але вони просто не активні. Деякі вузли публікації вважали, що це кращий блок, але потім перейшли на інше розгалуження ланцюга. Вузли публікації, як правило, не отримують винагороди за формування застарілих блоків, як і сам блок не залучається до основного ланцюга (див. рисунок 2).

Блок сирітський (*orphan blocks*) – називають ті блоки, «батьківський блок» яких поки невідомий, тобто, вузол його ще не обробив або не має відомого батька в самому довгому ланцюжку блоків (див. рисунок 2).

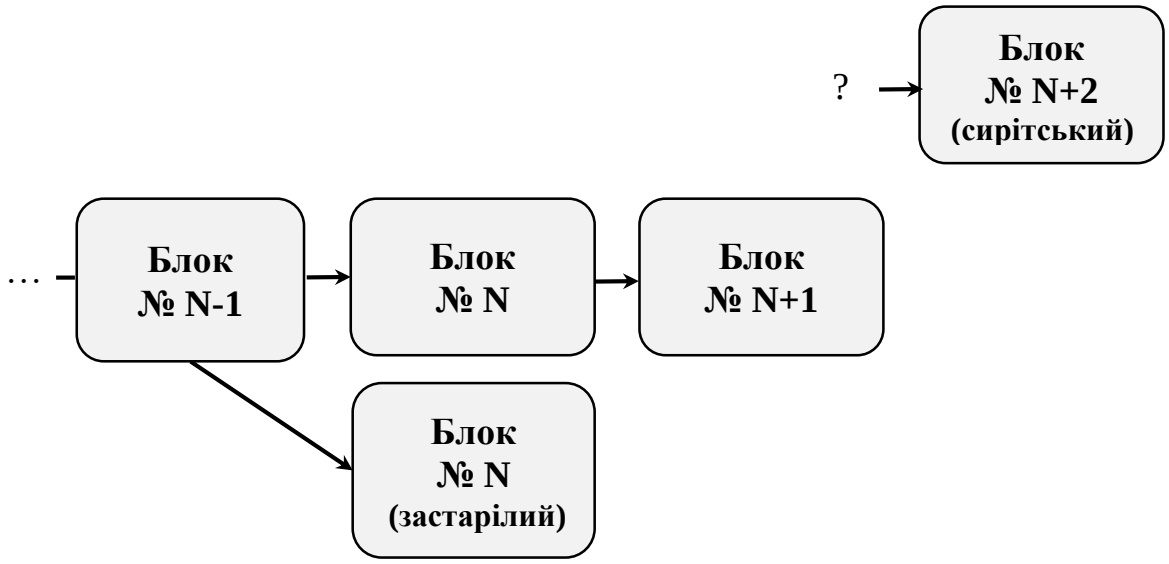


Рисунок 2 – Блокчейн ланцюжок з застарілими і сирітськими блоками

Блок споріднений (*uncle blocks*) – зазвичай асоціюються з протоколом GHOST [9] використовуваний в Ethereum і є еквівалентом застарілих блоків, але з невеликою відмінністю. Споріднені блоки як і раніше є дійсними блоками, які були створені і відхилені мережею. Однак, на відміну від застарілого блоку, де вузли публікації не отримують винагороду за їх виробництво, в цій реалізації вузли фактично отримують винагороду за створення родинного блоку.

Блокчейн (blockchain) – система, яка є захищеним від несанкціонованої модифікації цифровим реєстром, реалізованої в розподіленому вигляді.

Блокчейн мережа – мережу, в якій використовується блокчейн.

Блокчейн система – конкретна блокчейн система.

Блокчейн технологія – термін для опису технології блокчейн в найбільш загальній формі.

Валідатор блокчейна (Chain Validator) – вузол блокчейна, який володіє часткою блокчейн мережі. Кожен валідатор блокчейн мережі може вирішити, чи є транзакція дійсною, і може опитати всі транзакції, відправлені в свою блокчейн мережу.

Винагорода за блок (Block reward) – нагорода (зазвичай криптовалюта), що присуджується вузлам публікації за успішне додавання блоку в блокчейн.

Відмовостійкість (Fault tolerance) – властивість системи, що забезпечує правильну роботу навіть у разі збою компонентів.

Вузол (Node) – окрема система в блокчейн мережі.

Вузол публікації (Publishing node) або *Майнінг-вузол (mining node)* – вузол, який в доповнення до всіх обов’язків, необхідним для повного вузла, доручено розширити блокчейн шляхом створення і публікації нових блоків.

Гаманець (Wallet) – програмне забезпечення, що використовується для зберігання і управління асиметричними ключами і адресами, використовуваними для транзакцій.

Генезис блок (Genesis block) – перший блок блокчейн мережі, який записує початковий стан системи.

Геш швидкість (Hash rate) – кількість криптографічних геш-функцій, які процесор може обчислити за даний проміжок часу, зазвичай позначаються як геш-кодування в секунду.

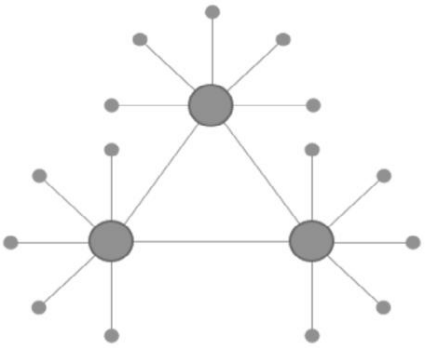
Геш-значення (Hash digest) – вихідні дані геш-функції (наприклад, $hash\,(дані) = геш$). Також відомий як геш повідомлення.

Геш-ланцюжок (Hash chain) – структура даних передбачає тільки додавання, де дані об'єднуються в блоки даних, причому нові блоки даних включають в себе геш попереднього блоку даних. Така структура даних доводить фальсифікації, тому що будь-яка модифікація блоку даних змінить геш-значення записане в наступному блоці.

Гешування (Hashing) – метод обчислення щодо унікального виходу (званого геш-значенням) для входу практично будь-якого розміру (файлу, тексту, зображення і т.п.) шляхом застосування криптографічної геш-функції до вхідних даних.

Депонування (Escrow) – процес утримання коштів або активів на сторонньому рахунку для захисту їх під час асинхронної транзакції. Якщо Боб хоче відправити гроші Алісі в обмін на файл, але вони не можуть провести обмін особисто, то як вони можуть довіряти один одному, щоб відправити гроші і файл один одному одночасно? Замість цього Боб відправляє гроші Єві, довіреної стороні, яка зберігає кошти, поки Боб не підтвердить, що отримав файл від Аліси. Потім вона відправляє Алісі гроші.

Децентралізована мережу (Decentralized network) – конфігурація мережі, в якій існує кілька центральних органів, які служать в якості централізованих вузлів для підмножини учасників. Оскільки тільки деякі учасники знаходяться у взаємодії з деяким центральним вузлом, втрата цього вузла завадить тільки цим учасникам спілкуватися, але не завадить спілкуванню всіх інших.



Древо Меркла (Merkle tree) – структура даних, в якій дані гешіруются і об'єднуються до тих пір, поки не буде єдиного кореневого гешу, що представляє всю структуру (див. рисунок 3).

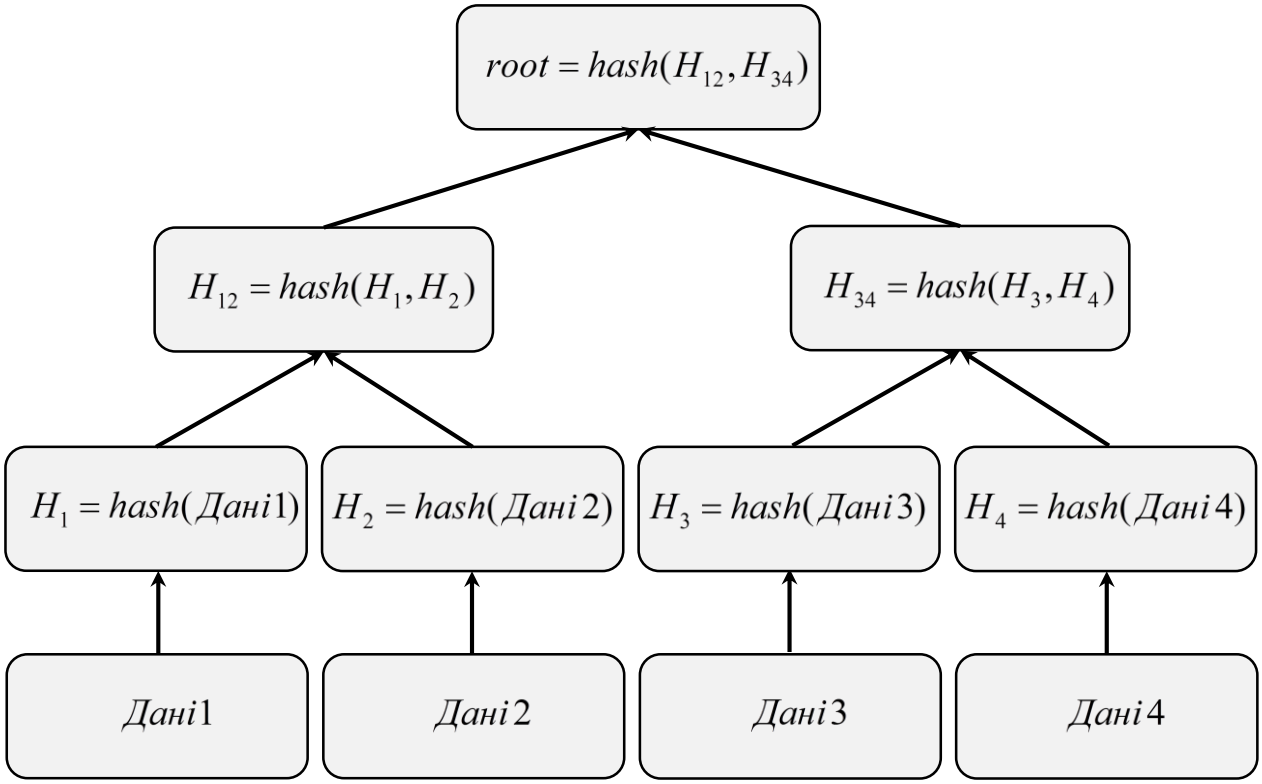


Рисунок 3 – Структура Древа Меркла

Ексклюзивний (Permissioned) – вказує на обмеження, тобто необхідність отримати дозвіл на участь в деякому процесі.

Ексклюзивний блокчейн (Permissioned blockchain) – система, в якій кожному вузлу і кожному користувачу повинно бути надано дозвіл на використання системи (зазвичай призначається адміністратором або консорціумом).

Електронна мітка часу (electronic time stamp) – дані в електронному вигляді, що зв'язують інші дані в електронному вигляді з конкретним часом, встановлюють докази того, що останні дані існували в певний проміжок часу.

Живучість (*Liveness*) – властивість, яка гарантує, що якщо усі чесні учасники хочуть додати запис в загальну базу даних, то в підсумку вона буде туди додано.

Жорстке розгалуження (Hard fork) – зміна в блокчейн реалізації, яке не є зворотно сумісним. Неоновлені вузли не можуть продовжувати транзакцію з оновленими вузлами.

Завершеність (*Finality*) – властивість, яке вказує на не відмінність прийнятого рішення або підтверджених даних.

Заголовок блоку (Block header) – частина блоку, яка містить інформацію о самому блоці (метадані блоку), зазвичай включає в себе тимчасову мітку, геш-представлення даних блоку, геш заголовка попереднього блоку і криптографічний одноразовий номер - *nonce* (при необхідності).

Захищеність (Tamper evident) – процес, який робить можливим легко виявити буть-які зміни в даних.

Ідентифікація (Identity) – збірник специфікацій, правил і юридичних зобов'язань необхідний для встановлення унікальності учасників будь-якої угоди.

Інклюзивний (Permissionless) – означає вільний доступ до прийняття участі в деякому процесі.

Інклюзивний або *загальнодоступний блокчейн (Permissionless blockchain)* – система, в якій права всіх користувачів рівні і не встановлені жодним адміністратором або консорціумом.

Клієнт (Client) – програмне забезпечення, яке користувач запускає на настільному комп'ютері, ноутбуці або мобільному пристрої для запуску програми.

Кліринг і розрахунок (Clearing and Settlement) – процес, через який відбувається обмін активів для оплати. Цей процес відбувається після укладання угоди і є невід'ємною частиною постторгового циклу.

Ключова пара (Key Pair) – відкритий ключ і відповідний йому закритий ключ, який використовується в криптографії з відкритим ключем.

Комісія за транзакцію (Transaction fee) – кількість криптовалюти, що стягується за обробку транзакції у блокчейн системі. Надається вузлу публікації для включення транзакції в блок.

Контрольна сума (Checksum) – значення обчислене з даних для виявлення помилок або маніпуляцій з цими даними.

Конфіденційність (Confidentiality) – в блокчейні це можливість зробити зміст транзакції недоступним для всіх, крім зацікавлених сторін транзакції.

Користувач блокчейн мережі (Blockchain network user) – будь-яка окрема людина, група, бізнес або організація, яка користується або підтримує блокчейн-вузол.

Криптовалюта (Cryptocurrency) - цифровий актив / кредит / одиниця в системі, що передається від одного користувача блокчейн мережі до іншого з застосуванням криптографії. У разі створення криптовалюти (наприклад, винагороди за майнінг) вузол публікації включає транзакцію, що відправляє новостворену криптовалюту одному або декільком користувачам блокчейн мережі. Ці активи передаються від одного користувача іншому з використанням цифрових підписів з парами асиметричних ключів.

Криптографічний ключ або *ключ (Cryptographic Key* or *Key)* – параметр, який визначає, можливо, з іншими параметрами, роботу криптографічної функції, такі як:

перетворення відкритого тексту в зашифрований і навпаки;

синхронізоване створення ключових даних;

обчислення або підтвердження електронного підпису.

Криптографічний одноразовий номер (*Cryptographic nonce*) – довільне число, яке використовується один раз.

Криптографічна геш-функція (Cryptographic hash function) – функція, яка відображає бітовий рядок довільної довжини в бітовий рядок фіксованої довжини. Схвалені геш-функції задовольняють наступним властивостям (див. [10]):

1. (Стійкий до прообразу) В обчислювальному відношенні неможливо обчислити правильне вхідне значення при деякому вихідному значенні (геш-функція «одностороння»).
2. (Стійкий до другого прообразу) В обчислювальному відношенні неможливо знайти вхід, який гешірует до певного вихідного значення.
3. (Стійкість до колізій) В обчислювальному відношенні неможливо знайти будь-які два різних входу, які відображаються в один і той же вихід.

Криптографія (Cryptography) – дисципліна, яка включає принципи, засоби та методи для перетворення даних, щоб приховати їх інформаційний зміст, запобігти їх несанкціоновану модифікацію, використання.

Криптографія з відкритим ключем (*Public key cryptography*) – див. Асиметрична криптографія.

Легкий вузол (Lightweight node) – блокчейн вузол, який не зберігає повну копію блокчейна і часто передає свої дані повним вузлів для обробки.

Майнінг (Mining) – процес вирішення завдання в рамках роботи моделі консенсусу.

Мітка часу (time-stamp) – цифрові дані в блокчейн системі, які пов'язують інші цифрові дані з конкретним періодом часу, встановлюючи свідоцтво того, що останні дані існували в певний момент часу.

Модель консенсусу (Consensus model) – процес досягнення угоди в розподіленої системі про дійсний стан системи. Також відомий як *алгоритм консенсусу* (*consensus algorithm*), *механізм консенсусу* (*consensus mechanism*), *метод консенсусу* (*consensus method*).

М'яке розгалуження (Soft fork) – зміна в реалізації блокчейн системи, яка зворотно сумісна. Не оновлені вузли можуть продовжувати працювати в мережі з оновленими вузлами.

Недобросовісні вузли (користувачі) – вузли (користувачі) блокчейн мережі які не підкоряються встановленим в даній мережі правилам і протоколам.

Незмінність (Immutable) – дані, які можуть бути тільки записані, але не змінені або видалені.

Обмін (Exchange) – центральний ресурс для обміну різних форм грошей та інших активів. Обміни біткоінів зазвичай використовуються для обміну криптовалюти на інші (зазвичай фіатні) валюти.

Одноранговa, децентралізована, або *пірінгова* (*peer-to-peer*, P2P – рівний до рівного) *мережа* – це оверлінійна комп'ютерна мережа, заснована на рівноправ'ї учасників. Часто в такій мережі відсутні виділені сервери, а кожен вузол є як клієнтом, так і виконує функції сервера. На відміну від архітектури клієнт-сервера, така організація дозволяє зберігати

працездатність мережі при будь-якій кількості і будь-якому поєднанні доступних вузлів.

Опір несанкціонованому втручанню (Tamper resistant) – процес, який робить зміни даних важкими (важкими для виконання), дорогими (дорогими для виконання) або і тим, і іншим.

Оракули (Oracles) – це агент, який знаходить і підтверджує реальні події і передає ці дані в блокчейн для використання смарт-контрактів. Цей агент може бути програмним, апаратним і людським. Смарт-контракти в блокчейн системах не можуть самостійно отримати доступ до зовнішньої мережі, тому оракули знаходяться між смарт-контрактом і зовнішнім світом, надаючи дані, необхідні для смарт-контракту, щоб підтвердити подію і відправити свої команди в зовнішні системи.

Орієнтований ациклічний граф (DAG - Directed Acyclic Graph) – одна з форм запису реєстру в блокчейн системах, має топологічне сортування, де кожен блок знаходиться в певному порядку. Конструкція складається з вершин, що з'єднуються ребрами. Кожне ребро направлено від більш раннього ребра до більш пізнього (див. рисунок 4).

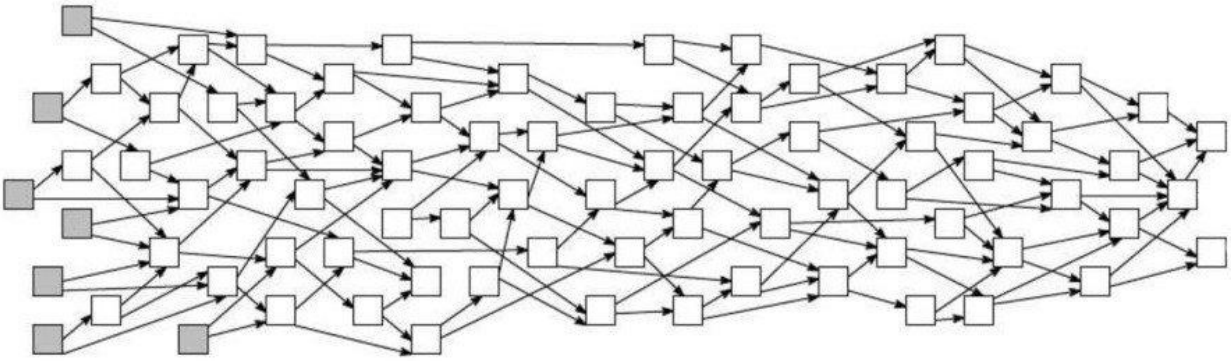


Рисунок 4 – Орієнтований ациклічний граф

Підпис (Signature) – цифровий геш, створений шляхом гешування ключів або інших компонентів разом, щоб довести, що транзакція сталася з певною адресою.

Підтверджена транзакція (Confirmed) – стан транзакції або блоку, коли досягнуто консенсусу щодо статусу включення в блокчейн.

Підтвердження (Confirmation) – підтвердження означає, що транзакція в блокчейн системі була перевірена мережею. Це відбувається через процес консенсусу, специфічний для даної блокчейн системи. Як тільки транзакція підтверджена, її не можна легко змінити або провести подвійну витрату.

Підтверджуючий запис (Endorsement) – набір цифрових підписів від підтверджуючих рівноправних користувачів блокчейн мережі, які встановлюють, що транзакція задовольняє політиці підтвердження.

Підтвердження користувача (Endorsing Peer) – вузол, який підтверджує транзакцію до її здійснення.

Підланцюжок (Subchain) – це пов'язаний, але логічно окремий ланцюжок всередині блокчейн системи.

Повний вузол (Full node) – вузол блокчейн мережі, який зберігає дані блокчейна, передає дані іншим вузлам і гарантує, що знову додані блоки є дійсними і автентичними.

Подвійна трата атака *(Double spend attack)* – атака, коли користувач блокчейн мережі намагається двічі використати один і той же цифровий актив.

Подвійна трата проблема *(Double spend problem)* – можливість створювати транзакції з одним і тим же набором цифрових активів більш ніж один раз.

Політика підтвердження або політика валідації (Endorsement Policy) – правила перевірки транзакції на валідність.

Постійність (*Persistence*) – здатність облікової системи зберігати незмінність кінцевого стану своєї бази даних навіть після того, як всі її валідатори відмовили.

Права доступу (Permissions) – допустимі дії користувача (наприклад, читання, запис, виконання).

Проект Linux Hyperledger (Linux Hyperledger project) – відкритий вихідний код, спільні зусилля з просування технології блокчейна шляхом визначення і врахування важливих функцій для міжгалузевого відкритого стандарту для розподілених реєстрів, які можуть змінити спосіб проведення бізнес-транзакцій в глобальному масштабі. Hyperledger служить фундаментальним кодом для продуктів, сервісів і рішень IBM Blockchain.

Протокол взаємодії в мережі блокчейн – це набір правил.

Протоколи допомагають:

забезпечити життєздатність транзакцій в мережі;

усунути можливість подвійної витрати;

упевнитися, що учасників не шахраюють.

Протокол – це сума:

детермінованих логічних правил;

криптографії та шифрування як основи безпеки;

соціального заохочення, щоб підтримувати мережу протоколу.

Пул транзакцій, що очікують (Pending transaction pool) або *пул пам'яті (memory pool, mempool)* – розподілена черга, в якій транзакції-кандидати чекають, поки вони не будуть додані в блокчейн.

Реєстр (Ledger) – запис транзакцій.

Рішення конфлікту (Conflict resolution) – зумовлений метод для досягнення консенсусу щодо стану системи. Наприклад, коли частини учасників системи стверджують, що існує Стан_А, а решта учасників стверджують, що існує Стан_В, виникає конфлікт (див. рисунок 5). Система автоматично вирішить цей конфлікт, вибравши «дійсне» стан того, з якої групи додається наступний блок даних. Усі транзакції, «втрачені» у не вибраному стані, додаються назад в пул очікування транзакцій.

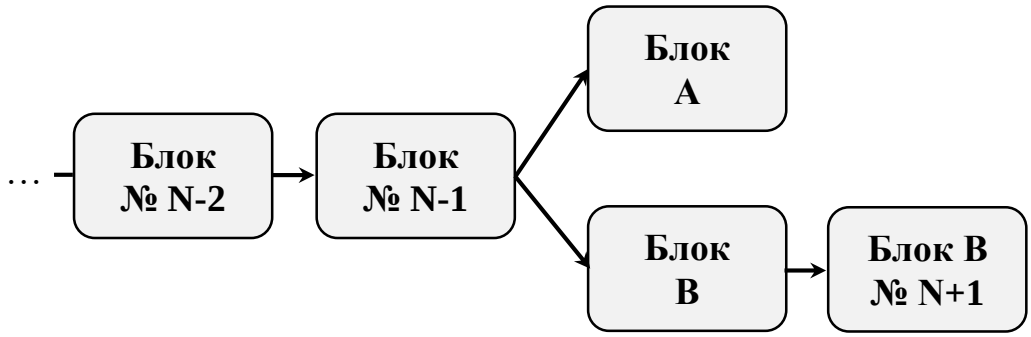
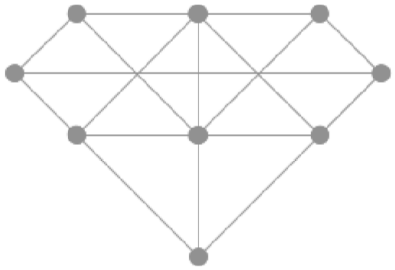


Рисунок 5 – Ланцюжок з блоком_n (B) додає наступний блок, ланцюжок з блоком_n (A) стає застарілим

Розгалуження (Fork) – зміна в програмному забезпеченні блокчейн мережі (зазвичай алгоритм консенсусу). Зміни можуть бути сумісні – см. м'яке розгалуження (Soft Fork), або зміни можуть не мати зворотну сумісність – см. жорстке розгалуження (Hard Fork).



Розподілена мережа (Distributed network) або *одноранговa мережа (Peer-to-peer network)* – конфігурація мережі, в якій кожен учасник може спілкуватися один з одним, не використовуючи централізовану точку. Оскільки існує кілька шляхів спілкування, втрата будь-якого учасника не завадить спілкуванню.

Розподілений реєстр (Distributed Ledger) – тип бази даних, яка поширюється по декільком серверам, країнам або установам. Записи зберігаються однин за одним в безперервному реєстрі. Дані розподіленого реєстру можуть бути «ексклюзивними» або «інклюзивними», що дозволяє контролювати, хто може їх переглядати.

Сертифікат (Certificate) – цифровий документ, який пов'язує відкритий ключ з ідентифікаційною інформацією власника сертифіката, тим самим дозволяючи власнику сертифіката проходити автентифікацію. Сертифікат видається Центром сертифікації та має цифровий підпис цього органу.

Сібіл атака (*Sybil Attack*) – атака комп'ютерної безпеки (не обмежуючись блокчейн мережами), коли зловмисник може створити безліч вузлів (тобто створити декілька ідентифікаторів), щоб отримати вплив і здійснювати певний контроль.

Система винагород (Reward system) або *система стимуляції (incentive system)* – винагороди користувачам за діяльність в блокчейн мережі (зазвичай використовується в якості системи винагороди за успішну публікацію блоків).

Смарт контракт або *розумний (інтелектуальний) контракт (Smart Contract)* – набір бізнес-термінів, які вбудовані в блокчейн і виконуються з

транзакціями. Смарт контракт може також включати цифрове представлення набору бізнес-правил і визначати умови, при яких відбувається передача. Розумний контракт виконується вузлами в блокчейн мережі, всі вузли повинні отримувати однакові результати для виконання, а результати виконання записуються в блокчейн.

Список відкликаних сертифікатів (Certificate Revocation List) – список цифрових сертифікатів, які були відкликані Центром сертифікації, яким вказані сертифікати були видані, до запланованої дати закінчення терміну їх дії, і сертифікатам більше не слід довіряти. Список відкликаних сертифікатів є типом чорного списку і використовуються різними кінцевими точками, включаючи веб-браузери, щоб перевірити, чи є сертифікат дійсним і заслуговує на довіру

Стан блокчейна (state of a blockchain) – сукупність всіх транзакцій, для яких ймовірність залишитися в складі блокчейна після досягнення консенсусу вважається досить високою.

Примітка 1: Імовірність залежить від обраного узгодженого механізму і системних властивостей.

Примітка 2: Часто стан блокчейна являє собою набір всіх транзакцій з достатньою кількістю підтверджень, щоб вважатися дійсними.

Технологія розподіленого реєстру (Distributed Ledger Technology) – технологія, що дозволяє будь-якому учаснику мережі побачити записи реєстру всіх учасників мережі.

Транзакція (Transaction) – запис події, такої як передача активів (цифрова валюта, одиниці майна тощо) між сторонами, або створення нових активів.

Тьюринг повний (Turing complete) – система (комп'ютерна система, мова програмування тощо), яку можна використовувати для будь-якого алгоритму, незалежно від складності, при пошуку рішення.

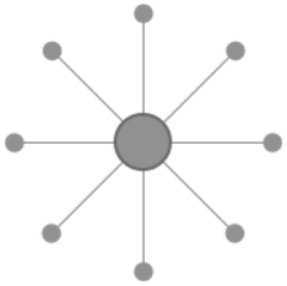
Управління ключами (Key Management) – генерація, зберігання, безпечне поширення і застосування ключових даних у відповідності з політикою безпеки.

Учасник (Participant) – дійова особа, яка може отримати доступ до реєстру: читати записи або додавати записи.

Фіатна валюта (Fiat currency) – будь-які гроші, оголошені урядом як дійсні для виконання фінансових зобов'язань, такі як USD або EUR.

Центр сертифікації (Certificate Authority) – організація, якій довіряє один або кілька інших організацій для створення і призначення сертифікатів.

Централізована мережа (Centralized network) – конфігурація мережі, в якій учасники повинні спілкуватися з центральним органом, щоб взаємодіяти один з одним. Оскільки всі учасники повинні проходити через єдине централізоване джерело, втрата цього джерела завадить спілкуванню всіх учасників.



Цифровий актив (Digital asset) – будь-який актив, який є чисто цифровим або є цифрове представлення фізичного активу.

Цифровий підпис (Digital signature) – криптографічний метод, який використовує асиметричні ключі для визначення авторства (тобто користувачі можуть перевірити, що повідомлення було підписано за допомогою закритого ключа, відповідає вказаному загальнодоступному ключу), неспростовність (користувач не може заперечувати, що відправив повідомлення) і цілісність (підтверджує, що повідомлення не було змінено під час передачі).

Цифровий сертифікат (Certificate Digital) – відкритий ключ і ідентифікаційні дані об’єкта, а також деяка інша інформація, яка зв’язується однозначним чином і підписується закритим ключем Центру сертифікації що видав сертифікат.

Чейнкод (Chaincode) – код виконання (контрольований децентралізований застосунок), який розгорнуто в блокчейн мережі, де він виконується і перевіряється валідаторами блокчейн мережі спільно з процесом консенсусу. Розробники можуть використовувати чейнкод для взаємодії із загальним реєстром блокчейн мережі, розробки бізнес-контрактів, визначень активів і децентралізованих застосунків з колективним управлінням.

ПРОТОКОЛИ КОНСЕНСУСУ ДЕЦЕНТРАЛІЗОВАНИХ СИСТЕМ

Протокол консенсусу відповідно до моделі «Доказ виконаної роботи» (PoW)

Опис алгоритму консенсусу наведено у [4, 8, 11, 12, 13].

Принцип: важко знайти рішення, але легко перевірити результат.

Продуктивність: низька.

Середовище: інклюзивні блокчейн мережі.

Завершеність: імовірнісна.

Приклад використання: Bitcoin, Bitcoin Cash [14], Ethereum [15], Litecoin, Electroneum, Zcash, Monero, Ravencoin [16], SUQA [17] і багато інших.

Опис моделі консенсусу PoW

«Доказ виконаної роботи» (PoW – proof of work) було «винайдено» ще на початку 90-х і застосовувалося в контексті захисту від спаму і DoS-атак. Наприклад, один варіант доказу роботи (Hashcash) був запропонований англійським криптографом Адамом Беком (Adam Back) [18].

У моделі «Доказ виконаної роботи» користувач публікує наступний блок, якщо першим вирішує складну задачу. Рішенням цієї задачі є «доказом» того, що він виконав роботу. Завдання розроблена таким чином, що вирішити його було складно, але перевірити, що рішення правильно – легко. Це дозволяє всім іншим повним вузлам легко перевіряти будь-які запропоновані наступні блоки, і будь-який запропонований блок, який не задовольняє завданню, буде відхилено.

Поширеним методом завдання є вимога, щоб геш-значення заголовка блоку було менше цільового значення. Вузли публікації вносять безліч невеликих змін в заголовок свого блоку (наприклад, змінюючи параметр nonce), намагаючись знайти геш-значення, що відповідає вимозі. Для кожної спроби вузол публікації повинен обчислювати геш для всього заголовка блоку. Багаторазове гешування заголовку блоку стає трудомістким процесом. Цільове значення може бути змінено з плином часу, щоб відрегулювати складність (підвищити або знизити), впливаючи тим самим на частоту публікації блоків.

Наприклад, Біткойн, який використовує модель «Доказу виконаної роботи», коригує складність завдання кожні 2016 блоків, щоб впливати на швидкість публікації блоків – у середньому один раз в десять хвилин. Коригування рівня складності завдання, і, по суті, або збільшує, або зменшує кількість необхідних ведучих нулів. Збільшуючи число ведучих нулів, це збільшує складність завдання, тому що будь-яке рішення повинно бути менше рівня складності – тобто, існує менше можливих рішень. Зменшуючи кількість провідних нулів, він знижує рівень складності, тому що є більше можливих рішень. Ця установка призначена для підтримки обчислювальної складності задачі і, отже, для підтримання основного механізму – безпеки мережі Біткоін.

Доступна обчислювальна потужність з часом збільшується, як і кількість публікованих вузлів, тому складність завдання, як правило, зростає.

Як приклад на рисунку 6 представлена зміна складності вирішення завдання для мережі Біткоін [19].

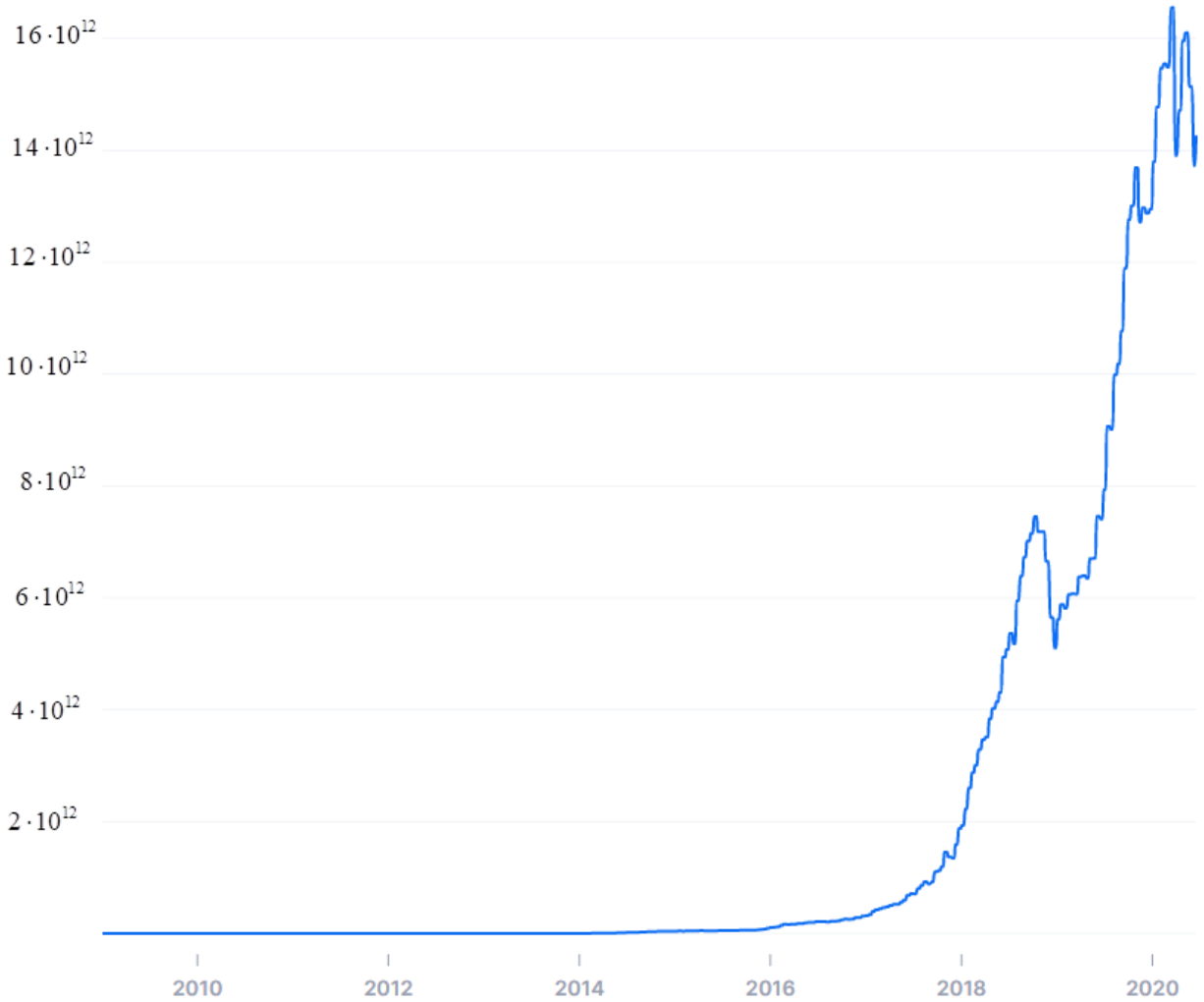


Рисунок 6 – Складність майнінга для мережі Біткойн.
Скріншот з Blockchain.com (https://www.blockchain.com/ru/charts/difficulty)
станом на 18.06.2020

Коригування цільової складності спрямовані на те, щоб гарантувати, що жоден об’єкт не зможе взяти на себе виробництво блоків, але в результаті обчислення, вирішення завдань вимагає значного споживання ресурсів.

Важливим аспектом цієї моделі є те, що робота, закладена в задачу, не впливає на ймовірність вирішення поточних або майбутніх завдань, оскільки завдання незалежні. Це означає, що, коли користувач отримує сформований і перевірений блок від іншого користувача, у нього з’являється стимул відмовитися від своєї поточної роботи і замість цього почати будувати наступний за нещодавно отриманим блоком, тому що він знає, що інші вузли публікації також будуть будувати наступний блок.

Як приклад розглянемо завдання, в якій за допомогою алгоритму SHA-256 комп'ютер повинен знайти значення геш-функції, відповідне наступним цільовим критеріям (відомим як рівень складності):

$$SHA256\ (\text{«блок»} + Nonce) = \text{геш} - \text{значення, що починається з «000000»}$$

У цьому прикладі до текстового рядка «блок» додається одноразове значення (nonce), а потім обчислюється геш-значення. Це відносно просте завдання, і нижче наведено приклад рішення:

$SHA256(\text{блокчейн\_0}) =$
$= 0xbd4824d8ee63fc82392a6441444166d22ed84eaa6dab11d4923075975acab938$
(не вирішено)
$SHA256(\text{блокчейн\_1}) =$
$= 0xdb0b9c1cb5e9c680dfff7482f1a8efad0e786f41b6b89a758fb26d9e223e0a10$
(не вирішено)
...
$SHA256(\text{блокчейн\_10730895}) =$
$= 0x\mathbf{000000}ca1415e0bec568f6f605fcc83d18cac7a4e6c219a957c10c6879d67587$
(вирішено)

Щоб вирішити цю задачу, потрібно 10 730 896 припущень (виконаних за 54 секунди на відносно старому обладнанні, починаючи з 0 і тестуючи одне значення за раз). У цьому прикладі кожне додаткове значення «провідний нуль» збільшує складність. Збільшуючи мету ще на один провідний нуль («0000000»), тим же апаратним засобам знадобилося 934 224 175 спроб для вирішення завдання (виконаної за 1 годину 18 хвилин 12 секунд):

$SHA256(\text{блокчейн\_934224174}) =$
$= 0x\mathbf{0000000}e2ae7e4240df80692b7e586ea7a977eacbd031819d0e603257edb3a81$

В даний час не відомо більш швидких рішень для вирішення такого завдання ніж повний перебір. Вузли публікації повинні витрачати обчислювальні потужності, час і ресурси, щоб знайти необхідне значення nonce для вирішення завдання.

Після того, як вузол публікації виконав цю роботу, вони відправляють свій блок з допустимим значенням nonce на повні вузли в блокчейн мережу. Повні вузли перевіряють, що новий блок задовольняє вимозі складності, потім додають блок в свою копію ланцюжка блоків і повторно відправляють блок на рівноправні вузли з яким підтримують зв'язко. Таким чином, новий блок швидко розподіляється по мережі вузлів, що беруть участь у підтримки блокчейн мережі. Перевірка значення nonce проста, так як потрібно тільки один геш, щоб перевірити, чи задовольняє він вирішенню завдання.

Для багатьох доказів проведеної роботи блокчейн мереж, заснованих на роботі, вузли публікації, як правило, об'єднуються в «пули» (англ. pools – басейни) або «колективи» (collectives), в результаті чого вони працюють

разом, щоб вирішувати завдання і розподіляти винагороду. Це можливо, тому що робота може бути розподілена між двома або більше вузлами в пулі, щоб розподілити робоче навантаження і винагороду. Розбиваючи приклад програми на частини, кожен вузол може взяти рівну кількість діапазону значень для одноразового номера при перевірці:

- Вузол 1: перевіряє nonce від 0000000000 до 0536870911.
- Вузол 2: перевіряє nonce від 0536870912 до 1073741823.
- Вузол 3: перевіряє nonce від 1073741824 до 1610612735.
- Вузол 4: перевіряє nonce від 1610612736 до 2147483647.

Наступний результат був знайдений першим для вирішення даного завдання:

$$SHA256(\text{блокчейн\_}1700876653) =$$
$$= 0x\mathbf{0000000}3ba55d20c9cbd1b6fb34dd81c3553360ed918d07acf16dc9e75d7c7f1$$

Це абсолютно новий nonce, але він все ж вирішив завдання. Знадобилося 90 263 918 переборів (виконано за 10 хвилин 14 секунд). Розподіл роботи між великою кількістю обчислювальних машин дає набагато кращі результати, а також більш послідовну винагороду за підтвердження моделі роботи.

Назва «Доказ виконаної роботи» відображає той факт, що для знаходження nonce треба зробити обчислювальну роботу, очікувану кількість якої вимірне. Наприклад, якщо потрібно, щоб перші 16 біт геш-значення дорівнювали нулю, то в середньому потрібно перебрати 65 536 значень nonce.

Використання складного, в обчислювальному відношенні, завдання допомагає боротися з «Сібіл-атакою». Моделі доказу виконаної роботи бореться з цим, фокусуючись на вплив мережі на кількість обчислювальної потужності (апаратне забезпечення, яке коштує грошей), змішане з системою лотереї (більшість апаратних засобів збільшує вірогідність, але не гарантує її) в порівнянні з мережевими ідентифікаторами (які, як правило, безкоштовні для створення).

Протокол GHOST

Протокол GHOST [20] (англ. Greedy Heaviest Observed Subtree – Жадібне і найвагоміше видиме поддерево) в Ethereum був введений в 2013 році як спосіб боротьби з тим, що алгоритм формування ланцюжка блоків з коротким проміжком між доданими блоками страждав від великої кількості застарілих блоків. Незважаючи на те, що ці блоки були правильні, але в кінцевому підсумку відкинуті, так як не увійшли в довший ланцюжок. Протокол також бореться з проблемою централізації – чим більше пул, тим менше витрачається часу, тим частіше вони отримують перевагу перед іншими блоками, роблячи сам блок і негайно запускаючи програму для створення такого блоку.

GHOST включає в себе застарілі блоки і перевіряє, які з ланцюжків має велику довжину або має найвищу кумулятивну складність. Централізація

вирішується шляхом надання винагород за блок (87,5%), а предок застарілого блоку також отримує винагороду (12,5%) від блоку.

Ethereum реалізує спрощену версію GHOST [21], яка працює тільки на семи рівнях. У блоці повинні вказуватися його предки і кількість застарілих блоків. Застарілий блок, на який посилається новий блок, повинен бути прямим нащадком цього нового блоку, а так само нащадком блоків, які на сім блоків нижче його по висоті. При цьому, він не може бути прямим предком формованого блоку. Додатковими умовами для застарілих блоків є: блок повинен мати діючий заголовок блоку; блок повинен відрізнятися від всіх інших застарілих блоків і формуватися по-новому.

Реалізацію протоколу GHOST можливо дослідити на прикладі (див. рисунок 7)

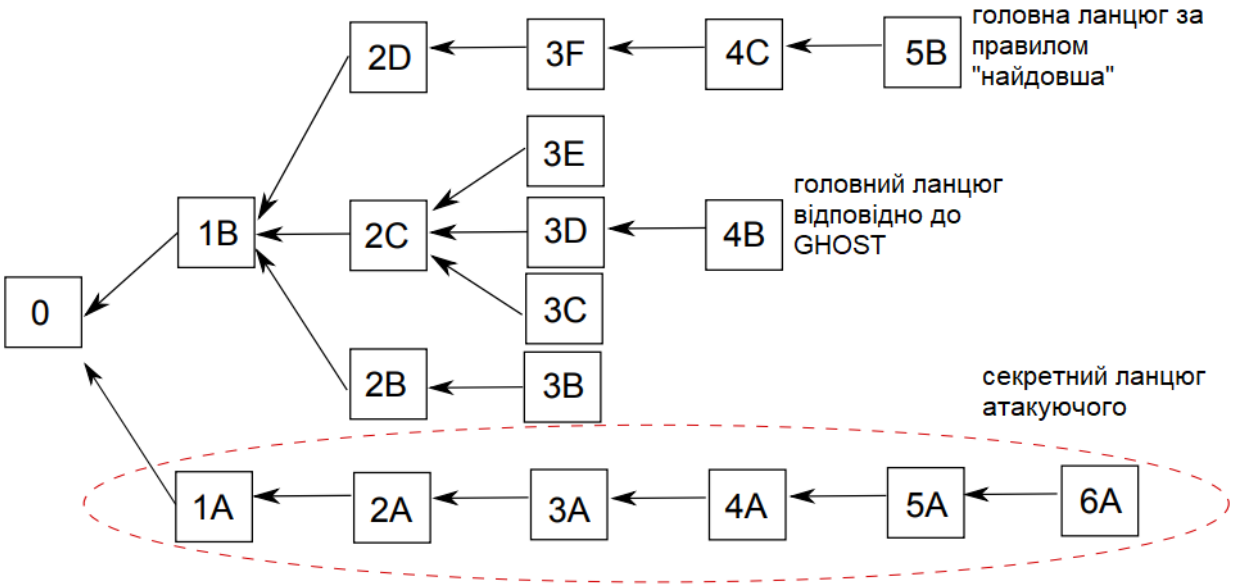


Рисунок 7 – Дерево блоків, в якому розрізняються найдовша ланцюг і ланцюг, обраний протоколом GHOST. Ланцюжок зловмисника може переключити найдовший ланцюжок, але не тій, який обрано GHOST.

В цьому випадку зловмисник може формувати свій ланцюжок (зазначену на малюнку червоним кольором) і зробити її найдовшим, але він не буде виграшним. Чесні користувачі будуть слідувати тому ланцюжку, якій вони будували до цього.

У чесних користувачів в цьому випадку може бути два ланцюжки. Найдовший (1B-2D-3F-4C-5B), але вона буде коротше, ніж ланцюжок зловмисника. Протокол GHOST орієнтується не на найдовший ланцюжок, а на кількість блоків в дереві, утвореному поточним ланцюжком. Враховується не тільки довжина самого ланцюжка, але і блоки на різних її висотах. Таким чином, виходить не лінійний ланцюжок, а дерево. Враховується кількість блоків, яке в ньому знаходиться.

Якщо подивитися на ланцюжок 1B-2C-3D-4B, то бачимо, що його супроводжують блоки 3E і 3C. По кількості блоків і витраченої кумулятивної

роботі цій ланцюжок відповідає найбільшим вкладеним зусиллям і саме він буде прийнятий за основну. Чесні користувачі продовжать вважати його основним, незважаючи на спроби зловмисника атакувати мережу. У традиційному консенсусі PoW така атака була б успішною, але для GHOST вона не становить загрози.

Проте недоліком GHOST залишається той факт, що частина блоків все ще втрачається. В даному випадку ланцюжок 2D-3F-4C-5B все одно буде відкинута [16]. Отже, питання відкидання блоків чесних користувачів залишається відкритим.

Протоколи SPECTRE і PHANTOM

З метою збільшення частоти формування блоків і вирішення проблеми з відкиданням блоків чесних користувачів було запропоновано ще два PoW-протоколу: SPECTRE [22] і PHANTOM [23].

Вони використовують не деревовидну структуру, а спрямований ациклічний граф (DAG – directed acyclic graph). В цьому випадку вузол публікації включає в заголовок свого блоку покажчики на блоки, на які ще не посилаються інші вузли публікації, принаймні в тому стані мережі, яке він бачить на поточний момент часу, і відправляє блок далі.

Тому виходить така структура, в якій включаються взагалі всі блоки, які бачить вузол публікації на поточний момент часу. Тут майнінгові потужності чесних користувачів взагалі не губляться. Більш того, забезпечується висока пропускна здатність мережі і високий рівень безпеки. Перевагою такого підходу є той факт, що система більш децентралізована [24].

У випадку з SPECTRE і PHANTOM майнінговие пули взагалі не потрібні. Якщо ми проведемо паралель з сучасною мережею біткоіна, то кожен користувач мережі, який використовує протоколи SPECTRE і PHANTOM, має шанс випустити один блок на добу. Таким чином, в майнинговых пулах потреба взагалі відпадає і потенційно приходимо до максимально децентралізованої мережі.

Отже, протоколи SPECTRE і PHANTOM схильні до забезпечення високого рівня децентралізації і високу пропускну здатність мережі, але потрібно зберігати великий обсяг інформації.

Протокол консенсусу відповідно до моделі «Доказ частки володіння» (PoS)

Опис алгоритму консенсусу наведено у [4, 11, 13, 25].

Принцип: мережа довіряє вузлу публікації, який ставить свої власні ресурси в заставу за можливість створювати блоки: чим більше частка, тим вище ймовірність, що мережа дозволить створення блоку.

Продуктивність: висока.

Середа: інклюзивні / ексклюзивні блокчейн мережі.

Завершеність: імовірнісна.

Приклад використання: NXT [26], Tezos [27], незабаром Ethereum [28].

Опис моделі консенсусу PoS

Модель консенсусу «Доказ частки володіння» (PoS – Proof of Stake) заснована на ідеї про те, що чим більше частка яку користувач вклав в систему, тим вище ймовірність того, що користувач буде зацікавлений в успішній роботі системи, і тим менш імовірно, що він захоче її порушити. Частка – це кількість кріптовалюти, яку користувач блокчейн мережі вклав в систему (з допомогою різних засобів, таких як блокування через спеціальний тип транзакції, відправка на певний адрес або утримання в спеціальному програмному забезпеченні). У доказі частки володіння блокчейн мережі використовують суму частки, яку користувач має в якості визначального фактора для публікації нових блоків. Таким чином, ймовірність того, що користувач блокчейн мережі публікує новий блок, пов'язана з відношенням його частки до загальної кількості кріптовалюти блокчейн мережі.

При використанні цієї моделі консенсусу немає необхідності виконувати ресурсоємні обчислення (включаючи час, електроенергію і обчислювальну потужність), які використовуються в доказі виконаної роботи. Оскільки ця консенсусна модель використовує менше ресурсів, деякі блокчейн мережі вирішили відмовитися від винагороди за створення блоків. Системи спроектовані таким чином, що вся криптовалюта вже розподілена серед користувачів, а нова криптовалюта генерується в постійному темпі. У таких системах винагороду за публікацію блоків зазвичай являє собою винагороду за транзакції, надані користувачем.

Способи використання частки в блокчейн мережах можуть варіюватися. Незалежно від точного підходу, користувачі з більшою часткою скоріш за все будуть публікувати нові блоки.

Найпростіший PoS-алгоритм вимагає від майнера, щоб той підписав новий блок приватним ключем, який відповідає адресі, на якій лежать його монети [29]. Блок вважається валідним, якщо

$$sha256(PREVHASH + ADDRESS + TIMESTAMP) \leq 2^{256} * BALANCE/DIFFICULTY,$$

де $PREVHASH$ – геш попереднього блоку;
$ADDRESS$ – адреса автора підпису з балансом $BALANCE$;
$TIMESTAMP$ – поточний час в секундах в Unix форматі;
$DIFFICULTY$ –змінний параметр, який використовується для налаштування частоти успішних підписів.

Коли вибір вузла, який публікує блок є випадковим процесом (який іноді називають доказом частки володіння на основі ланцюжка), блокчейн мережа буде аналізувати всіх користувачів, що мають частку, і вибирати серед них ґрунтуючись на відносної поставленої частки до загальної кількості поставленої кріптовалюти. Таким чином, якби у користувача було 42% всієї частки в блокчейн системі, його вибирали б 42% часу; ті, у яких 1% – будуть обрані 1% часу.

Як і у випадку з алгоритмом консенсусу «Доказу виконаної роботи», завершення транзакції в алгоритмі «Докази частки володіння» є ймовірнісним. Хоча транзакції відносно швидкі, в порівнянні з транзакціями в мережі біткоіна, для цього все ще потрібні токени. Більш того, скептики вказують на той факт, що вузли публікації з великими частками будуть вибиратися частіше і, отже, будуть отримувати ще більше токенов: багаті стають багатшими.

Видобуток криптовалюти за допомогою даного алгоритму також прийнято називати майнінгом, хоча цей процес більше схожий на відкриття депозиту в банку. Для того щоб запустити PoS-майнінг, достатньо мати на балансі певну кількість кріптовалюти, яка кладеться на депозит, після чого необхідно тримати локальний гаманець активним, тобто стати вузлом мережі. Початкові методи POS-майнінга не припускали обов'язкових депозитів, але подвійні витрати змінили ситуацію. Депозити стали необхідні в якості застави, яку втрачали шахраї при спробі здійснення повторної витрати.

В якості модифікації даного алгоритму консенсусу застосовується вимога, що одного разу поставлена криптовалюта більше не може бути витрачена. Є також багато інших модифікацій даного алгоритму консенсусу, розглянемо деякі з них.

Візантійський відмово стійкий доказ частки (BFTPoS)

Коли вибір вузла публікує блок є системою багаторазового голосування (іноді званої *Byzantine fault tolerance proof of stake – BFTPoS*, Візантійський відмово стійкий доказ частки [30]), виникає додаткова складність. Блокчейн мережа вибере декількох користувачів для створення запропонованих блоків. Потім всі зацікавлені користувачі голосують за запропонований блок. Може пройти кілька раундів голосування, перш ніж буде прийнятий новий блок. Цей метод дозволяє всім зацікавленим користувачам мати право голосу в процесі вибору блоку для кожного нового блоку.

Доказ віку монет (CAPoS)

Запропоновано проводити вибір вузла публікації через систему *доказу віку монет* (*CAPoS – coin age proof of stake*), частка криптовалюти, які беруть участь в доказі має властивість віку. Після закінчення певного часу (наприклад, 30 днів) частка кріптовалюти бере участь в алгоритмі консенсусу може зараховуватися на користь користувача-власника, який буде обраний для публікації наступного блоку. Тоді у кріптовалюти скидається вік, і її не можна використовувати знову, поки не пройде необхідний час. Цей метод дозволяє користувачам з великою кількістю акцій публікувати більше блоків, але не домінувати в системі – оскільки у них є таймер відновлення, пов'язаний з кожною монетою кріптовалюти, яка враховується при створенні блоків. Старі монети і великі групи монет збільшать ймовірність вибору для публікації наступного блоку. Щоб перешкоджати тому, щоб зацікавлені особи накопичували застарілі кріптовалюти, зазвичай існує вбудований максимум імовірності виграшу.

Коли вибір вузла публікує блок, здійснюється через *систему делегатів*, користувачі голосують за те, щоб вузли стали вузлами публікації, створюючи блоки від їх імені. Виборче право користувачів блокчейн мережі залежить від їх частки участі, тому чим більше їх частка, тим більшу вагу має голос. Вузли, які отримали найбільшу кількість голосів, стають вузлами публікації і можуть перевіряти і публікувати блоки. Користувачі блокчейн мережі можуть також голосувати проти встановленого вузла публікації, щоб спробувати видалити їх з набору вузлів публікації. Голосування за вузли публікації є безперервним, і останній вузол публікації, який залишився, може бути досить конкурентоспроможним. Загроза втрати статусу вузла публікації і, отже, винагороди та репутації є постійна, тому у вузлів публікації є стимул не діяти злочинно. Крім того, користувачі блокчейн мережі голосують за делегатів, які беруть участь в управлінні блокчейном. Делегати запропонують зміни і поліпшення, за які проголосують користувачі блокчейн мережі.

Орендований докази частки володіння (LPoS)

LPoS (Leased Proof-of-Stake) – це ще одна гібридна форма алгоритму PoS [13, 25]. У звичайній версії PoS, кожен вузол тримає певну кількість монет, щоб мати право згенерувати наступний блок в блокчейн системі. Але оскільки учасники з незначною часткою володіння мають низьку ймовірність бути обраними для формування наступного блоку, вони фактично не можуть майнити нові монети. З цієї ж причини учасникам з малою часткою володіння не вигідно брати участь у формуванні блоків, що призводить до малої кількості активних майнерів в мережі.

Цю проблему може вирішити оренда власних коштів більш великим вузлам публікації. Чим більше сума, що здається в оренду вузлу публікації, тим вище ймовірність того, що цей вузол публікації буде обраний для створення наступного блоку. Якщо даний вузол публікації обраний для створення такого блоку, тоді орендодавець отримає відсоток від суми транзакції, що збирається вузлом публікацій. Таким чином, орендодавець отримує можливість брати участь в майнингу і отримувати прибуток, а публікуючий вузол – більш високу ймовірність створення такого блоку.

Система LPoS дозволяє орендодавцям будь-який час робити з монетами, все, що завгодно: витратити їх або обміняти на альткоины. У цьому випадку «орендний» договір автоматично анулюється, і власник орендованих монет більше не може розраховувати на частку.

Але варто пам’ятати, чим більше вузлів, що генерують блоки, тим вище безпеку мережі. Об’єднання потужностей у великі пули підвищує ймовірність проведення успішних атак, які пов’язаних з розгалуженням головного ланцюга блоків.

Приклад використання: блокчейн система кріптовалюти WAVES [31].

Делегований доказ частки володіння (DPoS)

DPoS (Delegated Proof-of-Stake) алгоритм консенсусу відрізняється від LPoS тим, що кожен учасник, який заплатив мінімальний внесок, може делегувати свого кандидата на видобуток блоків. Це можна порівняти з виборами: чий кандидат потрапив до списку обраних, той і отримує дивіденди.

Він дозволяє створювати блоки на високій швидкості і обробляти більшу кількість транзакцій в секунду, у порівнянні з іншими алгоритмами консенсусу, за рахунок зменшення кількості валідаторів. Під час голосування власники монет обирають валідаторів транзакцій, які формуватимуть блоки.

Вага кожного голосування визначається сумою активів голосуючого. Власники монет можуть проголосувати за кандидатів у будь-який час. Це визначає високу стійкість мережі: якщо більшість виконавців зазнають невдачі, спільнота відразу ж проголосує за їх заміну. Проте підтвердження готових блоків все ще лежить на плечах всіх інших учасників мережі.

Приклад використання: EOS [32], BitShares [31], Lisk [33] (використовується список з 101 майнера), RaiBlock [34], Cardano [35], Steem [36].

Ієрархічний делегований PoS (HDPoS)

Модель консенсусу «Ієрархічний делегований доказ частки володіння» (HDPoS – Hierarchical Delegated Proof of Stake) пропонується в проекті Vite [37]. Vite використовує структуру реєстру у вигляді спрямованого ациклічного графа поряд з додатковою структурою – «знімком ланцюга» і алгоритмом консенсусу HDPoS. Сенс цього алгоритму полягає в тому, щоб розділити функції формування консенсусу на локальний консенсус і глобальний консенсус (див. рисунок 8):

локальний консенсус генерує блоки, відповідні транзакціям запиту і транзакції відповіді в обліковому записі користувача або контокорентного облікового запису, і записує їх до реєстру блокчейну;

глобальний консенсус знімає дані в реєстрі і генерує блоки знімків.

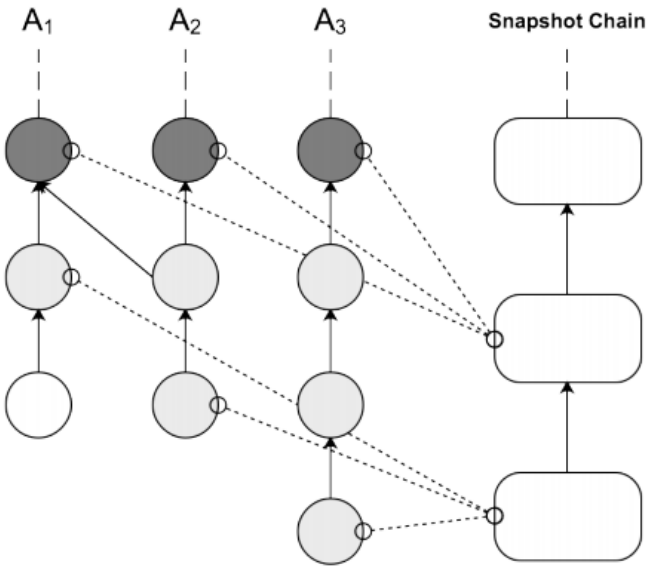


Рисунок 8 – Снепшот ланцюга

Користувачі можуть бути об’єднані в різні узгоджені групи і гнучка вибирати різні узгоджені параметри:

консенсус групи знімків. Найважливіша група консенсусу у всій системі. Відповідає за досягнення консенсусу в ланцюжку знімків;

приватна консенсус група. Блоки можуть бути створені тільки власником закритого ключа облікового запису. За замовчуванням всі облікові записи користувачів належать до закритої групи консенсусу. Основна заслуга цього підходу полягає в тому, що ініціація ситуації подвійних витрат є винятковою відповідальністю користувача (в гіршому випадку відповідальність за помилку програми). Це зменшує ймовірність появи другої гілки блоків;

делегована група консенсусу. Замість користувачів є проксі-вузли. В мережі також є загальнодоступна група консенсусу, яка «допомагає упакувати транзакції для всіх інших облікових записів, які не створили індивідуальну групу консенсусу».

Пріоритет глобального консенсусу вище пріоритету місцевого консенсусу.

Приклад використання: Vite [37].

Протокол досягнення консенсусу Ouroboros

Ouroboros [38] – це протокол на базі PoS, який забезпечує досягнення консенсусу між вузлами публікації транзакцій цифровий валюти Cardano [39, 40].

Спочатку розглянемо більш простий варіант, орієнтований на статистичну ставку (static stake), коли передбачається, що існуючий розподіл монет не змінюється.

Є певний генезис блок та користувачі, які формують нові транзакції, але вони не впливають істотним чином на розподіл монет.

Генезис блок містить в собі дані з деяким випадковим значенням, за допомогою якого відбувається вибір вузлів публікації. Вони дозволяють випускати блок в певний момент часу. Вузол публікації, що отримав таке право, збирає транзакції, отримуючи їх від іншого вузла, перевіряє коректність і випускає блок в мережу. Якщо він бачить кілька ланцюжків, то він вибирає найдовшу з них і приєднує блок до неї.

В контексті ситуації, пов’язаної зі статистичної ставкою, можливо використовувати такий підхід на протязі певного періоду часу, але потім розподіл частки (stake distribution) між різними користувачами може помінятися. Інакше кажучи, частина грошей переходить від одних користувачів до інших, і потрібно скорегувати ймовірність отримання права вибору блоку.

Примітка: статистична ставка має на увазі, що деякий проміжок часу, ставка (використовувана в доказі частки володіння) вузла публікації

вважається незмінним. Вузол може в цей час брати участь у прийнятті рішення і здійснювати платежі, але кількість монет у його частки, а отже, і вага його голосу залишаться незмінними до наступного періоду часу.

У разі динамічної ставки (dynamic stake) час ділиться на слоти, а слоти діляться на епохи. Тривалість однієї епохи приблизно прирівнюється до тривалості одного дня. Це співвідношення визначено з того, що протягом цього проміжку часу розподіл монет не може істотно змінитися.

По завершенні епохи фіксується поточний розподіл монет у кожного з користувачів. Крім того, генерується нове значення випадковості, щоб гарантувати, що в наступній епосі користувачі, які отримують право генерувати блоки, будуть дійсно обрані випадковим чином відповідно до кількості наявних у них монет.

Подібним чином відбувається захист від так званих grinding-атак, коли конкретний користувач може перебрати різні варіанти блоків, різні варіанти випадковості, щоб сформувати такий ланцюжок, в якій він може максимізувати свій прибуток. До таких атак потенційно вразливі платформи, засновані на POS-протоколах, такі як Peercoin [41] и NXT [26].

Творці даного алгоритму вирішили перераховані вище проблеми. Вузли публікації запускають між собою спеціальний протокол, який називається Протокол конфіденційного обчислення (MPC – multi-party computation) і дає можливість згенерувати випадковість спільно. Цей протокол також є доказово стійким, він заснований на вже давно відомих підходах.

Я стверджують розробники, протокол Ouroboros забезпечує стійкість за умови більшості чесних вузлів публікації в системі. Якщо чесні учасники, які працюють над випуском блоків, контролюють більше 50% монет в системі, протокол можна вважати захищеним.

Пропускна здатність облікової системи буде обмежена тільки затримками при синхронізації мережі. Зараз для збору транзакцій і випуску блоків досить звичайного персонального комп’ютера. У перспективі ці обчислення можна буде здійснювати навіть на звичайному смартфоні.

До обмежень протоколу Ouroboros можна віднести той факт, що перша версія протоколу є синхронною. Це означає, що повідомлення між учасниками повинні доставлятися в обмежений проміжок часу. Якщо в мережі будуть з’являтися більш тривалі затримки, ніж закладені в правилах, це може знизити безпеку. Проте вже заплановано використання наступної версії протоколу – Ouroboros Praos. В ній навіть при збільшенні затримок в мережі гарантується підвищена безпека.

Приклад використання: Cardano [39].

BFT-протоколи

Опис алгоритму консенсусу наведено у [25, 42, 43].

Принцип: проста і швидка реалізація алгоритму BFT для ексклюзивних мереж.

Продуктивність: висока.

Середовище: ексклюзивні блокчейн системи.

Завершеність: негайна.

Опис моделі консенсусу BFT

Наступний тип протоколів, які застосовуються – це візантійський протокол забезпечення відмовостійкості або BFT-протоколи (Byzantine Fault Tolerance).

Назва має походження з жартівливого опису проблеми, яке було зроблено в 80-х роках минулого століття при розробці протоколу для високонадійних систем [44]. Був наведений приклад про візантійських генералів, які організовують атаку на якесь місто. Завдання генералів полягало в тому, щоб прийти до єдиного рішення. Але потрібно було врахувати, що серед них є кілька зрадників. Зрадники могли вести себе довільним чином і, більш того, вони могли впливати на передачу повідомлень між чесними генералами, для яких важливо або одночасно атакувати місто, або одночасно відступити. Якщо якийсь із чесних генералів будуть діяти неузгоджено з іншими чесними генералами, то противник переможе армію по частинах. Тому їм необхідно було прийти до узгодженого рішення. Постановка задачі таким чином була зроблена в роботі, звідки і пішла така назва алгоритму досягнення консенсусу.

Для BFT-протоколів передбачається взаємодія рівноправних вузлів мережі. Передбачається, що є певна кількість зловмисників, які можуть діяти скоординовано. Вони невідомі чесним учасникам. У мережі можливі відмови і затримки, тобто частина повідомлень може пропадати, а частина може приходити з великою затримкою. Однак накладається обмеження на кількість кроків, протягом якого всі чесні вузли повинні прийти до спільного рішення.

У прикладі завдання є два варіанти: атакувати або відступити. Допускається, що чесні вузли становлять понад 2/3 учасників. Протокол дозволяє підписувати блок, навіть якщо 1/3 учасників зазнають невдачі або діють зловмисно. Лампорт довів [45], що в системі з f неправильно працюючими процесами («нелояльними генералами») можна досягти згоди тільки при наявності $2 \cdot f + 1$ вірно працюючими процесами («лояльними генералами»), тобто строго більше двох третин від загального числа процесів.

BFT-протоколи гарантовано приходять до загального рішення, яке в майбутньому не може бути скасовано.

Practical BFT

Перший протокол, який був застосований на практиці, називався Practical Byzantine Fault Tolerance [42, 46]. Він був запропонований в 1999 році. У нього досить просте функціонування (див. рисунок 9). Клієнт звертається до сервера, якого вибирає лідером. Лідер передає ці повідомлення іншим серверам. Після цього сервера повідомляють один одному про те, що прийшли певні повідомлення і їх необхідно додати до спільної бази даних. Кожен з

вузлів повинен підтвердити, що він отримав підтвердження від 2/3 інших учасників.

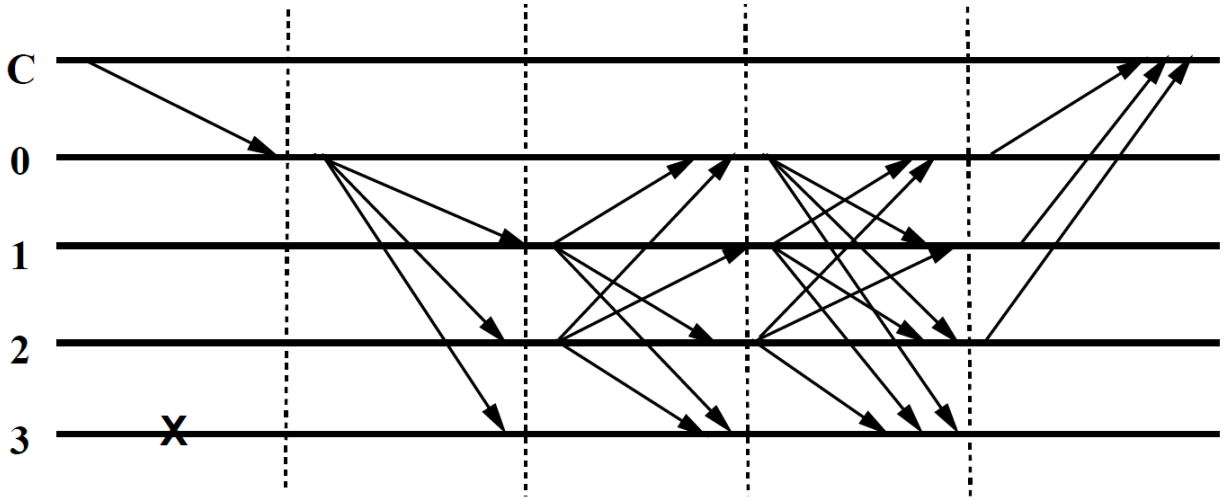


Рисунок 9 – Снепшот ланцюга

Коли вузол отримав підтвердження від 2/3 інших учасників про включення повідомлення в свою копію бази даних, то воно вноситься в локальну копію даного вузла.

Гіпотетично, в цій системі повідомлення можуть зазнавати втрати, пошкодження, затримки і повторення. Крім того, порядок відправлення може не обов'язково відповідати порядку прийому повідомлень. Дії вузлів можуть бути довільними, вони можуть приєднуватися і виходити з мережі в будь-який час.

Існують також модифікації, наприклад протокол Sieve [47], який є вдосконаленою версією Practical BFT. Його особливість в тому, що він вміє обробляти недетерміновані алгоритми і їх результати, тобто такі, які мають кілька шляхів обробки тих же вхідних даних.

Інший варіант це XFT – спрощений Practical BFT [48], стійкі до збоїв системи, і об'єднує синхронні і асинхронні протоколи для мережевих комунікацій.

Протокол XFT спрощує модель атаки і робить BFT практичним та ефективним для практичних сценаріїв. Протоколи BFT припускають сильного противника, який може контролювати зламані вузли, а також обмін повідомленнями по всій мережі.

Practical BFT ліг в основу алгоритмів консенсусу Hyperledger [49], Ripple [50], Stellar [51], IoT Chain [52] та Tendermint [53].

HoneyBadger BFT

HoneyBadger BFT [54] був розроблений як вдосконалена версія одного з існуючих алгоритмів. Він використовує ряд особливостей, в тому числі криптографічний захист всіх повідомлень, які передаються по мережі. Транзакції дешифруються тільки після досягнення певної угоди по ним.

HoneyBadger складається з двох протоколів: RBC (Reliable Broadcast – надійна трансляція) і BA (Byzantine Agreement – візантійська угода).

За допомогою RBC-протоколу повідомлення доставляються по мережі. Цей протокол завершує свою роботу тільки після того, як отримає підтвердження, що як мінімум 2/3 чесних вузлів отримали необхідний набір повідомлень. Після цього вузли переходять до BA-протоколу і узгодження самих транзакцій, які далі направляються в мережу.

Даний протокол забезпечує надійний криптографічний захист. Він добре працює в мережах з поганою якістю передачі даних, де є пропуски переданих повідомлень і затримки у їх доставці, а також в мережах з серйозним зловмисних втручанням в роботу.

Тут немає лідера. Протокол страждає при спробах значного масштабування. Результатом подальшого розвитку стали протоколи Algorand і Hashgraph.

Algorand

Протокол Algorand [55] був запропонований в 2017 році. Він використовує той же BFT-підхід, але він орієнтований на те, що серед всіх учасників випадковим чином вибирається деякий підкомітет, який виконує підтвердження транзакцій. Причому це підтвердження відбувається в кілька етапів, на кожному з яких вибирається свій підкомітет.

Протокол гарантує з високою імовірністю, близькою до одиниці, що підкомітет є чесним, якщо понад ⅔ учасників мережі є чесними, тобто кожен з підкомітетів також буде мати ⅔ чесних учасника. Тут немає лідера, тому і немає точки для атаки типу «відмова в обслуговуванні» (DoS-атаки). Цей протокол забезпечує високу пропускну здатність облікової системи, швидке підтвердження транзакцій.

Передбачається, що зловмисник може вибрати довільний вузол і заявити, що саме він працює в його інтересах. Зловмисник може обирати ті вузли, які працюватимуть на нього (обов’язковою для коректної роботи є умова, що чесних учасників залишається більше за особу). І навіть в таких умовах протокол забезпечує стійкість.

Оскільки це BFT-протокол, то в будь-якому випадку він забезпечує властивості безпеки і стійкості. А інші його властивості будуть залежати від затримок в передачі даних по каналах зв’язку.

Якщо цей протокол працює в проблемній мережі, то існує загроза живучості, пов’язана з тим, що нові транзакції можуть не отримувати підтвердження. Якщо в мережі з’являються дуже великі затримки чи зловмисник отримує можливість викидати ті повідомлення, які йому потрібно викинути для досягнення мети, то це ніяк не позначиться на старих (вже підтверджених) транзакціях. Їх він скасувати не зможе. Суть в тому, що він зможе відхиляти саме прийняття нових транзакцій.

HashGraph

У протоколі Hashgraph [56] використовується концепція DAG. Однак ключовою відмінністю HashGraph є протокол «gossip about gossip», де вузли отримують набір транзакцій з міткою часу, про які «знає» інший вузол. Для роботи такого алгоритму всі учасники в мережі повинні бути відомими. В результаті синхронізації кожен вузол зберігає всю інформацію та історію отримання цієї інформації усіма вузлами мережі. Як тільки вузол бачить в своїй історії, що конкретне повідомлення вже було отримано і перевірено більшістю, немає сумнівів, що воно дійсно. Продуктивність даного протоколу дуже висока. Завершеність залежить від раунду.

HashGraph гарантує, що за кінцеве число кроків, тобто раундів роботи протоколу, всі вузли приходять до єдиного рішення. Протокол працює швидко і добре масштабується. Він вимагає більшого обсягу трафіку, ніж Algorand, але в той же час є більш ефективним. HashGraph забезпечує властивості безпеки і стійкості, тобто додавання нових транзакцій і прихід до єдиного вирішення. Він працює навіть в умовах повністю асинхронної мережі, коли повідомлення можуть бути затримані на дуже великий проміжок часу або взагалі викинуті.

Крім того, в мережі відсутній лідер, що дозволяє зловмисникові виводити з ладу ті вузли, які йому виявляться цікавими, може змушувати їх працювати відповідно до своїх намірів. Однак поки зберігається 2/3 чесних користувачів, протокол забезпечує високий рівень безпеки.

Делегований BFT-протокол (DBFT)

Модель консенсусу Practical BFT має перевагу у порівнянні з протоколами на основі алгоритму PoW і PoS, але з огляду на тисячі вузлів публікацій, він все одно буде боротися за вирішення проблеми швидкості, тому розробники запропонували делеговану модель BFT (DBFT – Delegated Byzantine Fault Tolerance), яка має дуже високу продуктивність та негайну завершеність [11].

Принцип роботи DBFT протоколу полягає в наступному: попередньо вибираються «довірені» вузли публікації, які вже між собою підтримують механізм консенсус. Навіть якщо 1/3 «довірених» вузлів публікацій є недобросовісними або недоступними протокол залишається працездатним.

Заздалегідь визначені вузли публікації в цьому протоколі консенсусу дозволяють значно випередити інші протоколи. У Ethereum в середньому 5-20 транзакціями в секунду [57] та на NEO до 10 000 транзакціями в секунду. У разі, якщо вузол публікації перестає виконувати свої функції, учасники можуть делегувати новий вузол. Варто зазначити, що хоч цей протокол розрахований на публічне оточення, він є більш централізованим.

Примітка: оскільки NEO працює на протоколі PoS DBFT, члени мережі не тільки делегують повноваження вузлу публікації, але також отримують нативний токен GAS, як частина частки цих вузлів.

Приклад використання: NEO [58], TON [59].

Федеративна візантійська угода (FBA)

Принцип «федеративної візантійської угоди» (FBA – Federated Byzantine Agreement) полягає в тому, що кожен учасник вибирає певне число інших довірених груп, формуючи довірене коло, в якому блок перевіряється і підписується конкретним кворумом підписантів, після чого він вважається прийнятим. Має високу продуктивність та негайну завершеність [11].

Протокол не вимагає дозволу або заздалегідь відомого набору учасників, на відміну від PBFT і інших варіацій BFT. FBA дозволяє будь-кому приєднатися до мережі. Транзакції в цьому протоколі публікуються фіксованою кількістю учасників, які вибираються з тих, хто в той момент знаходяться в мережі.

Примітно, що за правилами FBA існують шлюзи (Gateways) і мейкери (Market-Makers), які забезпечують чесність і ліквідність мережі. Перші виступають в ролі традиційних банків, які володіють фінансовими засобами та створюють їх еквівалент в віртуальних токенах. Другі – ведуть облікові записи з численними шлюзами і відразу в декількох валютах.

Приклад використання: Ripple [50], Stellar [60].

Протоколи консенсусу відповідно до кругової моделі (Round Robin)

Принцип: вузли мережі публікують блоки по черзі.

Продуктивність: висока.

Середовище: ексклюзивні блокчейн системи.

Завершеність: негайна.

Кругова модель консенсусу (Round Robin Consensus Model) використовується деякими ексклюзивними блокчейн мережами. В рамках цієї моделі консенсусу вузли по черзі створюють блоки. Кругова модель консенсусу має довгу історію, засновану на архітектурі розподілених систем. Для обробки ситуацій, коли вузол публікації недоступний для публікації блоку при настанні його черги, ці системи можуть включати обмеження по часу, що дозволяє доступним вузлам публікувати блоки, щоб недоступні вузли не приводили до зупинки публікації блоку. Така модель гарантує, що жоден вузол не створить більшість блоків. Він виграє від простого підходу, не має криптографічних завдань і має низькі вимоги до енергоспоживання. Має високу продуктивність та негайну завершеність.

Оскільки існує потреба в довірі між вузлами, циклічний перебір не працює належним чином в мережах з інклюзивним блокчейном. Це пов’язано з тим, що недобросовісні вузли можуть безперервно додавати додаткові вузли, щоб збільшити свої шанси на публікацію нових блоків. У гіршому випадку вони можуть використовувати такий підхід, щоб порушити правильну роботу блокчейн мережі.

Приклад використання: Multichain [61], Tendermint [62].

Протоколи консенсусу з альтернативними моделями доказу

Доказ володінням простору (PoSpace)

Продуктивність: висока.

Завершеність: імовірнісна.

Крім «Доказів виконаної роботи» та «Доказів частки володіння», розвиваються і інші алгоритми «Доказу ...» [8], які полягають в використанні обмеженого обчислювального ресурсу, відмінного від обчислювальної потужності або валюти. Наприклад, Брем Коен (Bram Cohen), творець протоколу BitTorrent, запропонував використовувати для консенсусу в блокчейн системах доказ локального зберігання файлів (proof-of-space) [63], тобто замінити обчислювальну потужність в PoW на дисковий простір [64].

Майнинг на жорстких дисках реалізований в блокчейн криптовалюті BurstCoin [65].

Доказ ресурсів (PoCapacity)

Продуктивність: висока.

Завершеність: імовірнісна.

Алгоритм консенсусу доказ ресурсів (PoCapacity – Proof-of-Capacity) по суті є розширенням алгоритму PoSpace. Використовується алгоритм, згідно з яким вузли мережі надають частину своєї пам’яті або дискового простору для вирішення певних завдань. Запропонований ще в 2014 році, цей алгоритм є одночасно енергоефективним і позиціонується як рівномірно розподілений.

Цей алгоритм схожий з PoW, тільки замість чистого надання обчислювальної потужності використовується виконання завдань, пов’язаних як з рішенням криптографічних функцій, так і з виконанням завдань, що задіють великі масиви пам’яті і простору. Концепція «мегабайти як ресурси» передбачає використання значного обсягу пам’яті, щоб заповнити його даними. Чим більше пам’яті виділить учасник, тим вищі його шанси опублікувати блок.

PoCapacity алгоритм використовується у таких криптовалютах, як: Storj [66] (Призначена для хмарного зберігання даних), Burst [67] (спеціалізується на оренді вільного місця на жорсткому диску).

Доказ розташування (PoL)

Продуктивність: середня.

Завершеність: негайна.

Протокол «Докази розташування» (PoL – Proof-of-Location) використовує «маячки», щоб помітити вузол в синхронізований стан, а потім відзначити тимчасовим штампом її присутність.

Механізм PoL дозволяє користувачам закріпити за собою конкретну GPS-локацію і таким чином провести свою автентифікацію в мережі. Цікаво те, що протокол спирається на BFT-маячки, які записують геолокацію і маркери часу в блокчейн систему, що запобігає пошкодженню та шахрайству в системі.

Приклад використання: FOAM [68, 69], Platin [70].

Доказ важливості (PoI)

Продуктивність: висока.

Завершеність: імовірнісна.

В основі протоколу «Доказ важливості» (PoI – Proof-of-Importance) лежить алгоритм PoS, але з додатковими властивостями, які впливають на рейтинг учасника. Алгоритм PoI включає в себе три компоненти:

кількість токенів на рахунку;

активність операцій рахунку;

час, проведений власником рахунку в мережі.

Хоча перший параметр відіграє важливу роль в рейтингу для перевірки транзакцій, другий і третій параметри досить слабкі, але все ж допомагають встановити «важливість» облікового запису. Чим менше сума токенів, тим сильніше вплив інших параметрів.

Тому, вузол публікацій який вносить сотні тисяч токенів, може збільшити коефіцієнт значущості майже в 3 рази через її активності і постійної присутності в мережі. З іншого боку, це не має ніякого значення для тих, хто володіє сотнями мільйонів токенів в своєму аккаунті.

Приклад використання: NEM [71].

Доказ витраченого часу (PoET)

Продуктивність: середня.

Завершеність: імовірнісна.

Intel розробив власний блокчейн під назвою IntelLedger. Алгоритм консенсусу IntelLedger називається «Доказ витраченого часу» (PoET – Proof-of-Elapsed-Time).

В рамках моделі консенсусу «Доказ витраченого часу» кожен вузол публікації запитує час очікування у безпечного апаратного джерела часу в своїй комп’ютерній системі. Безпечне апаратне джерело часу генерує випадкове значення часу очікування і повертає його програмному забезпеченню вузла публікації. Вузли публікації беруть випадковий час, яке їм дають, і протягом цього періоду простоюють. Як тільки вузол публікації виходить зі стану очікування, він створює і публікує блок в мережі ланцюжка блоків, сповіщаючи інші вузли про нові блоки. Будь-який вузол публікацій, який все ще не використовується, перестане чекати, і весь процес почнеться заново.

За інформацією компанії Intel, алгоритм PoET можна масштабувати до тисяч вузлів, і він буде коректно працювати на будь-якому процесорі Intel, що підтримує SGX.

Сьогодні механізм консенсусу PoET присутній в одному з Hyperledger-продуктів.

Приклад використання: Intel.

Доказ частки та часу (PoST)

Продуктивність: середня.

Завершеність: імовірнісна.

У протоколі доказ частки часу (PoST – Proof-of-Stake-Time) розмір геша менше, ніж кратна кількість монет, частка часу і мета. Таким чином, учасники з меншою кількістю токенів, як і раніше мають можливість брати участь в майнингу в проектах на основі протоколу PoS. Це дещо схоже на протокол Proof-of-Importance (PoI) (Доказ важливості), але з невеликими відмінностями. Наприклад, коли потужність мережі нижче, час простою збільшується [25].

Приклад використання: VeriCoin [72].

Доказ активу (PoAsset)

Продуктивність: висока.

Завершеність: негайна.

Принцип «Доказ активу» (PoAsset – Proof-of-Asset) заснований на токенізації активів, часто фізичних товарів.

Оскільки природа реєстру в блокчейн системах не допускає помилок обліку, реєстр може об’єднувати фізичний актив або сертифікат з технологією блокчейна при співвідношенні 1:1.

За допомогою алгоритму PoAsset можна токенізіровать: золото; право власності на землю; права володіння; акції, облігації, боргові кредити та інші похідні фінансові інструменти.

Приклад використання: Digix [73], BANKEX [74].

Доказ авторитету (PoAuthority) або Доказ ідентичності (PoIdentity)

Продуктивність: висока.

Завершеність: імовірнісна.

Консенсус-модель «Доказ авторитету» (PoAuthority – Proof-of-Authority), також званий «Доказом ідентичності» (PoIdentity – Proof-of-Identity) полягає в тому, що право публікувати транзакцій мають тільки відомі учасники, що володіють необхідним авторитетом серед учасників блокчейн системи і будь-яким чином заслужили їх довіру.

В основі лежить довіра до вузлів публікації через їх відомий зв’язок з ідентичністю реального світу. Вузли публікації повинні мати свої посвідчення, доведені і перевірені в блокчейн мережі (наприклад, ідентифікуючи документи, які були перевірені та завірені нотаріально і включені в блокчейн). Ідея полягає в тому, що вузол публікацій ставить свою ідентичність / репутацію для публікації нових блоків.

Користувачі блокчейн мережі безпосередньо впливає на репутацію вузла публікацій, ґрунтуючись на його поведінці. Вузли публікації можуть втратити репутацію, діючи таким чином, з яким користувачі блокчейн мережі не згодні, точно так, як вони можуть отримати репутацію, діючи таким чином, з яким згодні користувачі блокчейн мережі. Чим нижче репутація, тим менше ймовірність публікації блоку. Тому в інтересах вузла публікації підтримувати

високу репутацію. Цей алгоритм застосовується тільки до ексклюзивних блокчейн мереж з високим рівнем довіри.

Це один з варіантів BFT-подібної приватній блокчейн системи, де схвалені акаунти мають право перевіряти транзакції і публікувати блоки, при цьому процес повністю автоматизований. Протокол PoAuthority дозволяє учасникам заробити право стати вузлом публікації, тому існує стимул зберегти авторитет і надалі.

Приклад використання: POA network [75], Parity [76].

Доказ мозкової діяльності (PoBrain)

Продуктивність: середня.

Завершеність: імовірнісна.

Принцип «Доказ мозкової діяльності» (PoBrain – Proof-of-Brain) складається в мотивації учасників створювати і керувати контентом, який буде зберігатися в блокчейні.

Такі блокчейн системи більшою мірою спрямовані на засоби масової інформації або соціальні мережі і пов'язують творців контенту з рекламодавцями, намагаючись виключити механізми накрутки і піар системи.

Протокол «Доказ мозкової діяльності» заснований на активності користувачів і заохочує якісний контент на відповідних платформах. Майнинг відбувається шляхом створення або взаємодії з контентом через голосування (лайки і коментарі) або перегляди. Чим більше лайків, коментарів або підтверджених переглядів на сторінці, тим більше монет може бути намайнено. Творці даного механізму ґрунтувалися на ідеї колективного розуму, що відповідно до задуму авторів робить цей алгоритм як розумним, так і соціальним.

Приклад використання: Basic Attention Token [77].

Доказ вкладу (PoCo)

Продуктивність: низька.

Завершеність: негайна.

Доказ вкладу заснований на потужності комп'ютера в мережі і подібний до протоколу Proof-of-Research (доказ проведеного дослідження), який винагороджує добровольців за те, що вони витрачають свою комп'ютерну потужність на великі наукові обчислення. Наприклад, на дослідження даних астрономічних спостережень в пошуку позаземного розуму SETI@Home [78], дослідження пульсарів Einstein@Home [79], а також складні обчислення для IBM World Community Grid [80] на платформі BOINC [81].

Приклад використання: iExec [82], CyberVein [83].

Доказ спалювання (PoBurn)

Концепція доказу спалювання (PoB – Proof-of-Burn) [84] полягає в тому, що вузли публікації доводять, що вони спалили монети (тобто, що вони

відправили монети на невитрачений адреса, який можливо перевірити). Хоча це дорого з точки зору людини, PoBurn не споживає ніяких ресурсів, крім «спалених» монет [85, 86].

Чим більше монет «спалює» вузол, тим вище у нього буде можливість здобути наступний блок. Згодом частка вузла в системі зменшує свою значимість, вимагаючи, щоб вузол спалювали більше монет, щоб збільшити шанси бути обраним для публікації наступного блоку.

Ця «оцінка» називається «Ефективні спалені монети». Щоб уникнути невідповідну вигоду для ранніх користувачів, після того, як учасник «спалює» монети, його значення «ефективних спалених монет» зменшується на кожен блок: починаючи з 100% від кількості спалених монет і зменшується до нуля протягом року. Це означає, що у вас буде багато часу, щоб відновити «спалені» монети за допомогою нагород за опубліковані блоки.

Приклад використання: Slimcoin [87, 88].

Гібридні моделі консенсусу

Існують і гібридні алгоритми досягнення консенсусу, наприклад деякі з уже згаданих раніше, суміщає PoW і PoS у криптовалюті Peercoin [41]; DPOS і BFT використовуються платформою BitShares [31] і Steem [36].

Доказ активності (PoActivity)

Продуктивність: низька.

Завершеність: імовірнісна.

Гібрид PoW і PoS використовується в «Доказ активності» (PoActivity – Proof-of-Activity) [89]. Протокол PoActivity прагне забезпечити баланс між майнерами та звичайними учасниками мережі.

Алгоритм PoActivity побудований на наступній послідовності дій:

1. POW-майнер шукає геш відповідної складності.
2. Знайдений геш відправляється в мережу, при цьому будучи ні блоком, а лише першим кроком, своєрідним шаблоном, необхідним для його створення.
3. Геш, що складається з 256 псевдовипадкових біт, інтерпретується як N чисел, до кожного з яких у відповідність ставиться один вузол.
4. Встановлюється взаємно однозначний зв’язок між кожним вузлом і публічним ключем його поточного власника.
5. Як тільки вся N кількість власників проставлять свої підписи на цьому блоці, на виході виходить повноцінний блок.
6. У разі, якщо один з вузлів не доступний або не бере участі у майнингу, то інші майнер продовжують генерувати шаблони з різними комбінаціями вузлів-кандидатів.
7. У якийсь момент необхідний блок буде підписаний потрібну кількість разів. Нагорода за блок розподіляється між POW-майнером та всіма N вузлами.

Алгоритм «Обмеження довіри» (LC)

«Обмеження довіри» [90] (LC – Limited Confidence) – механізм створення контрольних точок в мережі, що обмежує можливість перезапису блоків мережі блокчейн далі заданого ліміту від останнього блоку. Являє собою систему автоматичного створення контрольних точок блокчейн ланцюга. В основі алгоритму лежить система, яка забороняє перезапис ланцюжка блоків, старіше певного заданого порогу. Наприклад, при установці порога в 5 хвилин, буде можливий перезапис блоків, не більше 4 хвилин 59 секунд або можна встановити поріг в перезапису десяти останніх блоків.

Схожі реалізації контрольних точок блокчейн ланцюга реалізовані в деяких криптовалюта, наприклад, Peercoin [40].

Доказ активності з обмеженою довірою (LCPoA)

Доказ активності з обмеженим довірою, (LCPoA – Limited Confidence Proof-of-Activity) – гібридний алгоритм консенсусу мережі блокчейн, що складається з двох технічних елементів: «Доказ активності» (Proof-of-Activity) і «Обмеження довіри» (Limited Confidence) [91].

Метод захисту ланцюжка блоків в блокчейн системах, заснований на модифікації алгоритму Proof-of-Work, в сторону зменшення витрати обчислювальних ресурсів – потрібен підбір геш блоку, але в якості додаткового значення nonce використовується поточна мітка часу (unix timestamp). Такий метод дозволяє легко перевіряти інформацію про те, скільки часу було витрачено на створення блоку. У зв’язці з обмеженням додавання блоків молодше, наприклад, однієї секунди, це забезпечує одночасно просту для вирішення на будь-якому пристрої, але досить тривалу для захисту мережі задачу.

Особливість полягає в використанні в якості nonce - Unix timestamp (кількість мілісекунд, що пройшли з початку 1 січня 1970 року), а також в додаткові перевірки і обмеження:

Як загальний час використовується пояс GTM + 0.

В блок необхідно записувати інформацію про час старту генерації блоку.

У timestamp блоку записується nonce блоку.

Перевіряються параметри:

Збіг гешу блоку з фільтром дозволених гешів.

Час старту генерації блоку не більше поточного часу мережі, і не більше timestamp блоку.

Timestamp не більш поточного часу мережі, і не менше початку часу генерації блоку.

Timestamp і час старту генерації блоку не менше timestamp попереднього блоку.

Приклад використання: IZZZIO [92].

Loopchain Fault Tolerance (LFT)

Продуктивність: висока.

Завершеність: негайна.

Протокол Loopchain Fault Tolerance (LFT) є продовженням Tendermint [53] який об’єднує DPoS і PBFT. LFT нагадує алгоритм Round Robin своєї трьохетапною системою голосування (попередній, попередньо фіксований, фіксований), але скорочений до 2 кроків. У голосуванні бере участь обмежена кількість вузлів. LFT використовує техніку «Spinning» (Обертання), щоб спростити заплутаний алгоритм вибору первинної вузлів публікації.

Приклад використання: ICON [93].

ПЕРЕЛІК ПОСИЛАНЬ

1. Haber S., Stornetta S. How to Time-Stamp a Digital Document // Journal of Cryptology, Vol. 3, № 2, pp. 99-111, 1991 https://www.anf.es/pdf/Haber_Stornetta.pdf
2. Nakamoto, S., Bitcoin: A Peer-to-Peer Electronic Cash System, 2008. Available from: https://bitcoin.org/bitcoin.pdf
3. BitInfoCharts. Статистика криптовалют https://bitinfocharts.com/
4. NISTIR 8202 Blockchain Technology Overview https://doi.org/10.6028/NIST.IR.8202
5. ASC X9 Study Group Report Distributed Ledger and Blockchain Technology Study Group https://x9.org/wp-content/uploads/2018/04/Distributed-Ledger-and-Blockchain-Technology-Study-Group-Report-FINAL.pdf
6. DIN SPEC 3104:2019-04 Blockchain-basierte Datenvalidierung https://dx.doi.org/10.31030/3042007
7. Consensus - Immutable agreement for the Internet of value https://assets.kpmg.com/content/dam/kpmg/pdf/2016/06/kpmg-blockchain-consensus-mechanism.pdf
8. Алгоритмы консенсуса: Подтверждение доли и доказательство работы https://habr.com/company/bitfury/blog/327468/
9.
10. NIST SP 800-175B Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms, http://dx.doi.org/10.6028/NIST.SP.800-175B
11 12 консенсус-протоколов для распределенных систем https://dou.ua/lenta/articles/12-konsensus-protocols/
12. Blockchain https://www.blockchain.com
13 Токарев Д. Обзор алгоритмов консенсуса криптовалют 20.02.2018 https://bitcryptonews.ru/blogs/cryptocurrency/obzor-algoritmov-konsensusa-kriptovalyut
14. BitcoinCash https://www.bitcoincash.org/
15. Ethereum https://www.ethereum.org/
16. Ravencoin https://ravencoin.org/
17. SUQA https://suqa.org/
18. Adam Back A partial hash collision based postage scheme http://www.hashcash.org/papers/announce.txt
19. Сложность майнинга для сети биткойн – https://www.blockchain.com/ru/charts/difficulty?timespan=all
20. Sompolinsky Y, Zohar A. Secure High-Rate Transaction Processing in Bitcoin https://eprint.iacr.org/2013/881.pdf
21. Modified GHOST Implementation https://github.com/ethereum/wiki/wiki/White-Paper#modified-ghost-implementation

22. Sompolinsky Y., Lewenberg Y., Zohar A. SPECTRE: Serialization of Proof-of-work Events: Confirming Transactions via Recursive Elections https://eprint.iacr.org/2016/1159.pdf
23. Sompolinsky Y., Zohar A. PHANTOM, GHOSTDAG: Two Scalable BlockDAG protocols https://eprint.iacr.org/2018/104.pdf
24. Обзор актуальных протоколов достижения консенсуса в децентрализованной среде 03.08.2018 https://habr.com/en/company/distributedlab/blog/419185/
25. Еще 13 консенсус-протоколов для распределенных систем. Часть 2 25 января 2019 https://dou.ua/lenta/articles/konsensus-protocols-2/
26. NXT https://nxtplatform.org/
27. Tezos https://tezos.com/
28. Ethereum https://www.ethereum.org/
29. Проблемы криптовалют: proof-of-stake и доказательство хранения https://bitnovosti.com/2014/11/18/vitalik-buterin-hard-problems-of-cryptocurrency-4/
30. Bahsoun, J.P., Guerraoui, R., and Shoker, A., “Making BFT Protocols Really Adaptive,” 2015 IEEE International Parallel and Distributed Processing Symposium, Hyderabad, India, pp. 904-913, 2015. https://doi.org/10.1109/IPDPS.2015.21
31. Waves platform https://wavesplatform.com/
32. EOS https://eos.io/
33. Lisk https://lisk.io/
34. RaiBlock https://www.raiblocks.net/
35. Cardano https://www.cardano.org/
36. Steem https://steem.io/
37. Vite: Высокопроизводительная Асинхронная Децентрализованная Платформа для Приложений https://www.vite.org/whitepaper/vite_russian.pdf
38. Kiayias Aggelos, Russell Alexander, David Bernardo, Oliynykov Roman. Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol https://eprint.iacr.org/2016/889.pdf
39. Cardano Settlement Layer Documentation. Ouroboros proof of stake algorithm https://cardanodocs.com/cardano/proof-of-stake/
40. Larimer D. Независимый обзор алгоритма Ouroboros проекта Cardan. https://steemit.com/cardamon/@blockchained/nezavisimyi-obzor-algoritma-ouroboros-proekta-cardano-daniel-larimer
41. Peercoin https://peercoin.net/
42. Castro M. and Liskov B. Practical Byzantine Fault Tolerance Tolerance https://www.comp.nus.edu.sg/~rahul/allfiles/cs6234-16-pbft.pdf
43. Zhang E. A Byzantine Fault Tolerance Algorithm for Blockchain https://docs.neo.org/en-us/basic/consensus/whitepaper.html
44. Leslie Lamport, Robert Shostak, and Marshall Pease. The Byzantine Generals Problem. ACM Transactions on Programming Languages and Systems, Vol. 4,

No. 3, July 1982, Pages 382-401 https://www.microsoft.com/en-us/research/uploads/prod/2016/12/The-Byzantine-Generals-Problem.pdf
45. Leslie Lamport, Robert Shostak, and Marshall Pease. The Byzantine Generals Problem. ACM Transactions on Programming Languages and Systems, Vol. 4, No. 3, July 1982, Pages 382-401 https://www.microsoft.com/en-us/research/uploads/prod/2016/12/The-Byzantine-Generals-Problem.pdf
46. Castro M., Liskov B. Practical Byzantine Fault Tolerance. Appears in the Proceedings of the Third Symposium on Operating Systems Design and Implementation, New Orleans, USA, February 1999 http://pmg.csail.mit.edu/papers/osdi99.pdf
47. Cachin C., Schubert S., Vukolic M. Non-determinism in Byzantine Fault-Tolerant Replication https://arxiv.org/pdf/1603.07351.pdf
48. XFT: Practical Fault Tolerance Beyond Crashes http://www.eurecom.fr/en/publication/4575/download/sec-publi-4575_1.pdf
49. Hyperledger https://www.hyperledger.org/
50. Ripple https://ripple.com/
51. Stellar https://www.stellar.org/
52. IoT Chain A high-security lite IoT OS White Paper https://iotchain.io/whitepaper/ITCWHITEPAPER.pdf
53. Tendermint https://github.com/tendermint/tendermint
54. The Honey Badger of BFT Protocols https://github.com/amiller/HoneyBadgerBFT
55. Whitepapers Algorand https://www.algorand.com/docs/whitepapers
56. Hedera: A Public Hashgraph Network & Governing Council https://www.hedera.com/hh-whitepaper-v1.5-190219.pdf
57. Эфириум (ETH) цены и статистика https://bitinfocharts.com/ru/ethereum/
58. NEO https://neo.org/
59. Durov N. Telegram Open Network https://denull.ru/telegram/ton-tech.pdf
60. Stellar https://www.stellar.org/
61. Multichain https://www.multichain.com/
62. Tendermint https://tendermint.com/
63. Hamza Abusalah, Joël Alwen, Bram Cohen, Danylo Khilko, Krzysztof Pietrzak, and Leonid Reyzin. Beyond Hellman’s Time-Memory Trade-Offs with Applications to Proofs of Space https://eprint.iacr.org/2017/893.pdf
64. Stefan Dziembowski, Sebastian Faust, Vladimir Kolmogorov, and Krzysztof Pietrzak. Proofs of Space https://eprint.iacr.org/2013/796.pdf
65. BurstCoin https://www.burst-coin.org/
66. Storj https://storj.io/Storj https://storj.io/
67. Burst https://www.burst-coin.org/
68. FOAM https://map.foam.space
69. Blog FOAM https://blog.foam.space/
70. Platin https://platin.io/
71. NEM https://nem.io/
72. VeriCoin https://vericoin.info/
73. Digix https://digix.global/

74. BANKEX https://bankex.com/en/
75. POA network https://poa.network/
76. Parity https://www.parity.io/
77. Basic Attention Token https://basicattentiontoken.org/
78. SETI@Home https://setiathome.berkeley.edu/
79. Einstein@Home https://einsteinathome.org/
80. IBM World Community Grid https://www.worldcommunitygrid.org/
81. BOINC https://boinc.berkeley.edu/
82. iExec https://iex.ec/
83. CyberVein https://www.cybervein.org/
84. Proof of burn https://en.bitcoin.it/wiki/Proof_of_burn
85. Slimcoin-project https://github.com/slimcoin-project/Slimcoin/wiki
86. Proof-of-burn https://ru.bitcoinwiki.org/wiki/Proof-of-burn
87. Slimcoin http://slimco.in/
88. Slimcoin A Peer-to-Peer Crypto-Currency with Proof-of-Burn http://web.archive.org/web/20171010011956/, http://www.slimcoin.club/whitepaper.pdf
89. Bentov I., Lee C., Mizrahi A., Rosenfeld M. Proof of Activity: Extending Bitcoin's Proof of Work via Proof of Stake https://eprint.iacr.org/2014/452.pdf
90. LCPoA https://docs.google.com/document/d/1KHeG4iUZFk2fj32-4kVnbx_3KsIQqM_xDMpzlctGRSc/edit
91. LCPoA https://docs.google.com/document/d/1KHeG4iUZFk2fj32-4kVnbx_3KsIQqM_xDMpzlctGRSc/edit
92. IZZZIO https://izzz.io/
93. ICON https://icon.foundation/ICON https://icon.foundation/

Навчальне видання

Кузнецов Олександр Олександрович
Полуяненко Микола Олександрович
Краснобаєв Віктор Анатолійович
Кошман Сергій Олександрович

Довідник термінів і визначень технології блокчейн

Коректор _____
Комп'ютерне верстання _____
Макет обкладинки *І. М. Дончик*

Формат 60 х 84/16. Ум. друк. арк. 3,14. Наклад 50 пр. Зам № __/__.

Видавець і виготовлювач
Харківський національний університет імені В. Н. Каразіна,
61022, м. Харків, майдан Свободи, 4.
Свідоцтво суб'єкта видавничої справи ДК № 3367 від 13.01.2009

Видавництво ХНУ імені В. Н. Каразіна
Тел. 705-24-32