Розсоховатский Владимир Владимирович, Корниловска Наталья Владимировна, Лурье Ирина Анатольевна

Херсонский национальный технический институт, г. Херсон, Украина, 2018 год

Основы технологии Blockchain и концепты ее практического применения.

В статье рассматриваются основные принципы работы технологии Blockchain и предложения по внедрению ее в различные структуры.

Введение

31 октября 2008 года несколько сотен энтузиастов и специалистов по криптографии, включенных в закрытый список e-mail рассылки (The Cryptography Mailing list (Криптографическая рассылка) на сайте metzdowd.com, получили письмо, подписанное неким Satoshi Nakamoto(Сатоши Накамото). В нём он сообщил, что работает над созданием новой электронной системы денежных расчетов, в которой операции производятся непосредственно между участниками без привлечения третьей доверенной стороны.

В письме содержалась ссылка на короткий текст доклада под названием Bitcoin: A Peer-to-Peer Electronic Cash System («Биткоин: Одноранговая электронная денежная система»), в котором кратко описывалась технология новой денежной системы, названная автором Bitcoin (Биткоин).

На сегодняшний день, сложно найти того, кто еще не слышал ничего про эту нашумевшую криптовалюту. Однако явление Bitcoin - только

верхушка айсберга. Ее концепт основывается на такой структуре данных как Blockchain (Цепочка блоков). Впервые, упоминание криптоустойчивых цепочек блоков было описанно еще в 1991 году в статье «How To Time-Stamp a Digital Document» by Stuart Haber and W. Scott Stornetta («Как подписывать документы временной меткой» Стюарта Хабера и У. Скотт Сторнетта), а затем в 1992 Bayer, Haber and Stornetta (Байер, Хабер и Сторнетта) в своей статье «Improving the Efficiency and Reliability of Digital Time-Stamping» (Совершенствование эффективности и надежности цифровой временной метки) внедрили в блокчейн дерево Меркла, так же известное как древовидное хеширование, которое основывается на криптографических хеш функциях.

Как известно - хеш функция (До начала 1990-х годов в русскоязычной литературе в качестве эквивалента термину «хеширование», благодаря работам Андрея Петровича Ершова, использовалось слово «расстановка»), это любая функция, которая отображает данные произвольного размера в данные фиксированной длины.

Криптографически стойкой хеш функцией называют такую хеш функцию, которая обладает 3 основными критериями:

Необратимость или стойкость к восстановлению прообраза: для заданного значения хеш-функции m не должен быть вычислен блок данных X, для которого: H(X) = m

Другими словами, невозможно (кроме полного перебора) подобрать исходное сообщение, зная его хеш.

Стойкость к коллизиям первого рода или восстановлению вторых прообразов: для заданного сообщения M должно быть вычислительно невозможно подобрать другое сообщение N, для которого H(M) = H(N)

Другими словами, невозможно (кроме полного перебора) подобрать сообщение, отлшичное от исходного, хеш которого будет равен хешу исходного.

Стойкость к коллизиям второго рода: должно быть вычислительно невозможно подобрать пару сообщений (M, M') имеющих одинаковый хеш H(M) = H(M')

Другими словами, невозможно (кроме полного перебора) подобрать 2 сообщения у которых будет одинаковый хеш.

Полный перебор, в случае для наиболее используемой хеш функции SHA-256 потребует 2^256 итераций. Самые сильные майнеры, в сети биткоин обладают вычислительными мощностями порядка Терахешей (1 000 000 000 000) в секунду . Даже при такой вычислительной мощности понадобится порядка 10^59 лет, что несоизмеримо больше нынешнего возраста вселенной (10^10)

Дерево Меркла – структура данных, в которой хеши всех включенных элементов помещаются в листья бинарного дерева, а затем считаются хеши каждой пары хешей предыдущего уровня, пока не образуется один корневой хеш, зависимый от всех включенных элементов (см. Рисунок 1).

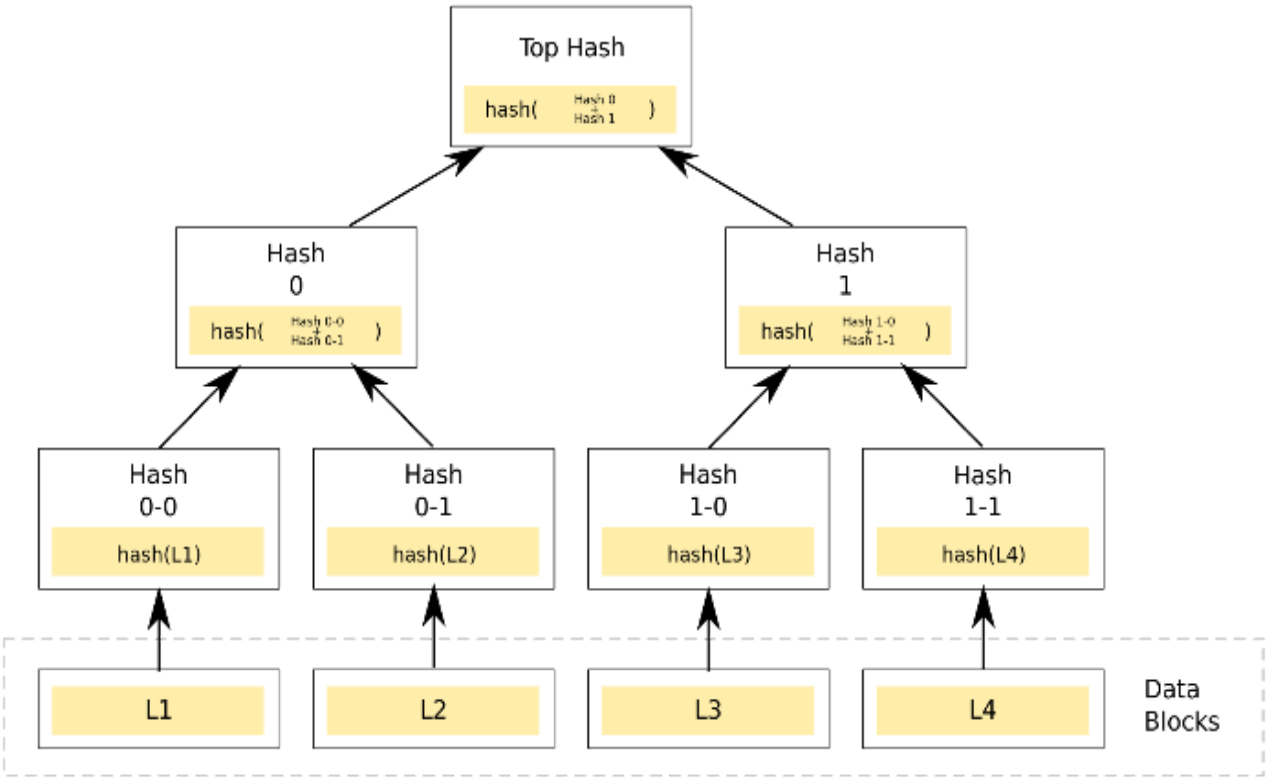


Рисунок 1. Дерево Меркла и демонстрация алгоритма его постраения на примере 4 элементов

Преимущество такого подхода дает возможность проверить включен ли определенный элемент в дерево с логарифмической сложностью:

например, имея 16 элементов (см. рисунок 2), нам необходимо знать всего 4 промежуточных хеша и сделать всего 4 итерации

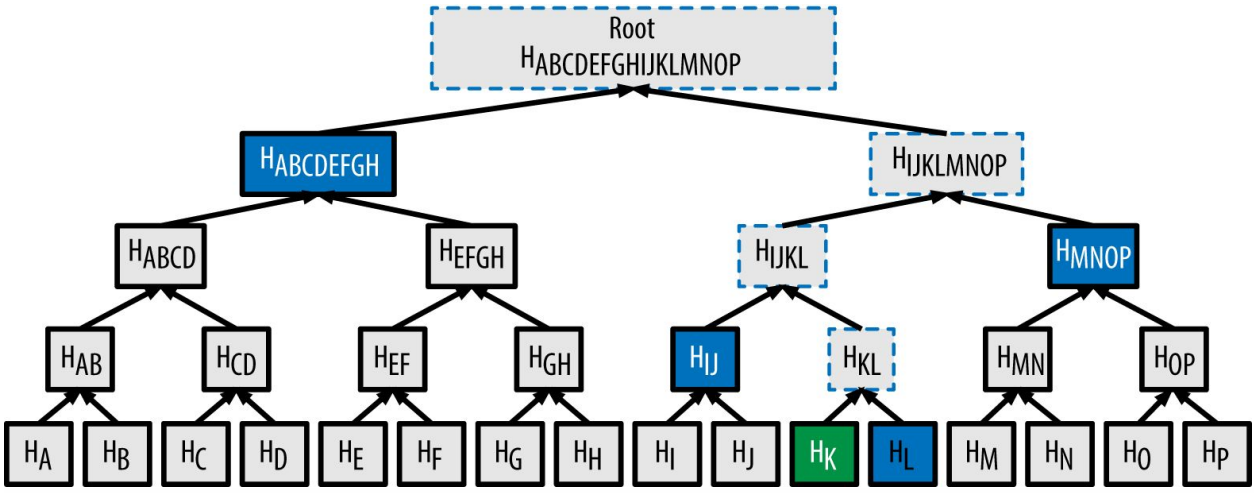


Рисунок 2. Дерево Меркла для 16 элементов и демонстрирация количества необходимой информации и количество итераций для проверки включения одного из элементов.

Это позволило эффективно собирать несколько документов в один блок. Это и стало началом того блокчейна, который сейчас набирает попюлярность с невероятной скоростью.

«Блокчейн – это неподкупная цифровая учетная книга экономических транзакций, которая может быть запрограммирована на запись не только финансовых транзакций, но и практически любых ценностей» [Don & Alex Tapscott, authors Blockchain Revolution (2016)]

Другими словами - это последовательная цепочка блоков, содержащих информацию, с жестким правилом: что каждый блок хранит хеш предыдущего, таким образом изменение блока внутри цепи затрагивает каскадное изменение всех последующих блоков. Таким образом, с помощью Blockchain формируются журналы, которые невозможно модифицировать (запись, попавшую в такой журнал уже нельзя ни изменить, ни удалить).

Благодаря этому появляется возможность создания децентрализованных структур хранения немодифицируемых журналов транзакций. При этом под транзакцией может пониматься что угодно:

финансовая транзакция (перевод между счетами), аудит событий аутентификации и авторизации, записи о выполненных ТО и ТУ автомобилей. В такой структуре событие считается случившимся, если запись о нём включена в журнал.

В такой системе существует 3 роли:

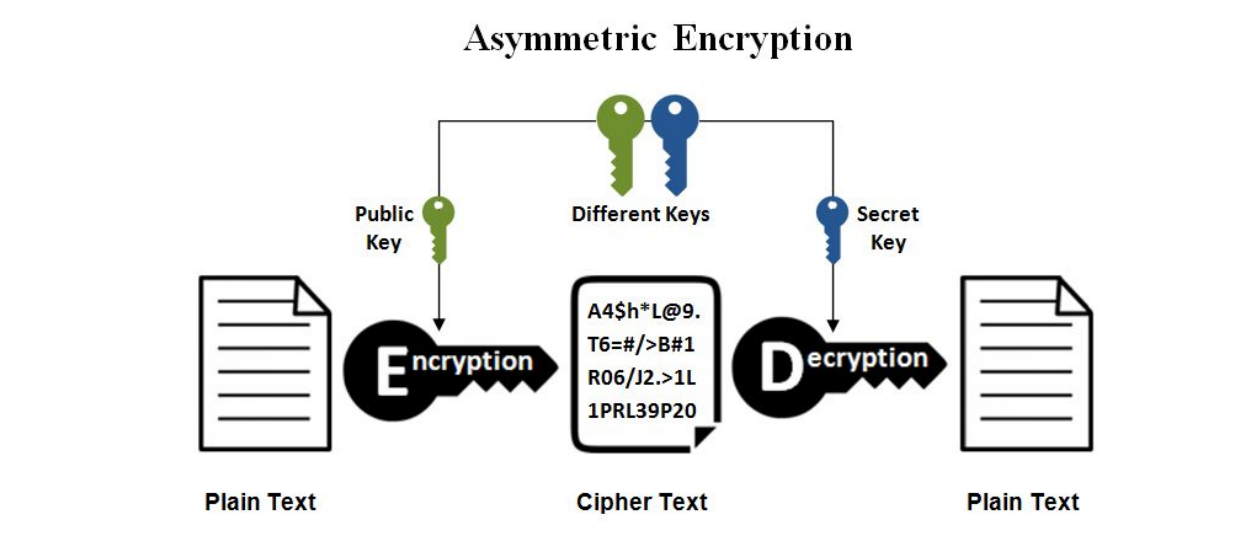
Источник транзакций – участник сети который добавляет новую информацию в сеть (привилегированный пользователь)

Источник блоков - участник сети, поддерживающей ее работоспособность за некое вознаграждение (их так же называют майнерами)

Наблюдатель – участник сети который только получает существующие блоки и транзакции (обычный пользователь)

В зависимости от реализации Blockchain, эти роли могут пересекаться.

Так же в блокчейне используются системы шифрования с открытым ключем – это такие системы, которые используют пару ключей: публичный (он же - открытый ключ, тот который предоставляется пользователем системы всем желающим) и приватный (он же – закрытый ключ, тот который пользователь системы хранит только у себя), и основываются на односторонних функциях. Один из ключей может только зашифровывать текст, а другой расшифровывать. В зависимости от того где и как используется система шифрования, публичным ключем может быть и зашифровующий (для передачи зашифрованных документов), так и разшифровующий (для проверки заверения электронной подписью).



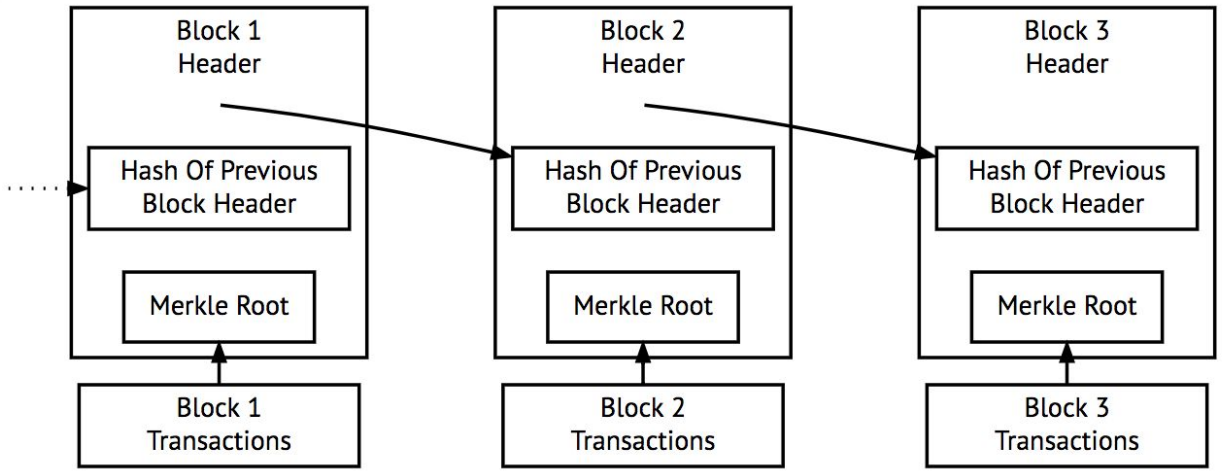
При обмене зашифрованными сообщениями – публичный ключ помещается в открытый источник, и любой желающий написать с помощью него шифрует свое сообщение. Получатель дешифрует это сообщения своим приватным ключем, тем самым получает исходное сообщение. Такой вид шифрования защищает от атаки Man-in-the-middle (так же известная как Атака посредника – вид атаки, когда в общение 2 пользователей через открытый канал связи может вмешаться третий участник и читать и/или изменять проходящие через него сообщения)

Электронная подпись – это схема демонстрации подлинности цифрового сообщения. Для ее формирования формируется хеш документа, этот хеш шифруется приватным ключом пользователя, полученная подпись прикладывается к документу вместе с открытым дешифрующим ключом. Таким образом любой желающий может взять публичный ключ, расшифровать подпись, сформировать хеш исходного документа, и сравнить

их. Если они идентичны – то значит и документ не мог подвергнуться модификации.

Ассиметричные алгоритмы используются, как уже упоминалось выше (вместе с хешированием) – для формирования электронной подписи транзакций для подтверждения права их регистрации.

Так же, важным фактором системы, работающей на цепочке блоков - является ее децентрализованность, что значит что версия блокчейна хранится у каждого пользователя сети. Пользователи – источники транзакций формируют новые транзакции и отправляют их в некий пул транзкций ожидающих подтверждения. Майнеры собирают эти транзакции и

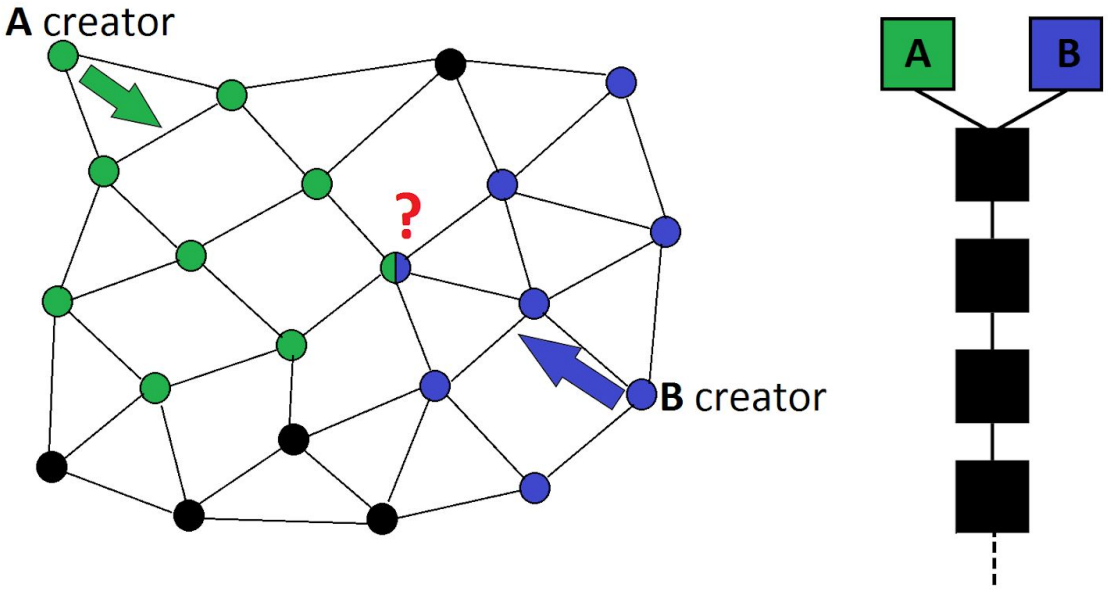


формируют новый блок, который состоит из хеша предыдущего блока, списка включенных транзакций, и корневого хеша дерева Меркла для всех включенных транзакций.

После того как блок сформирован, сформировавший его майнер разсылает его всем участникам сети к которым у него есть непосредственный доступ. Каждый участник сети, который получил этот новый блок сперва проверяет его валидность (вычисляет хеш предыдущего блока, и сравнивает

его с указанным в новом, рассчитывает дерево меркла и сравнивает его корень с полеченным в новом блоке), и если оно валидно – то разсылает его всем участникам сети.

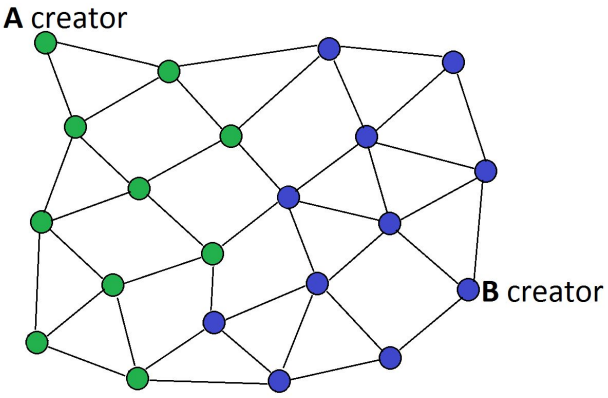
Однако может случиться так, что одновременно (или условно одновременно, по сравнению со временем передачи блока) два, или более майнеров соберут новый блок и начнут из разных сторон сети разсылать эти блоки

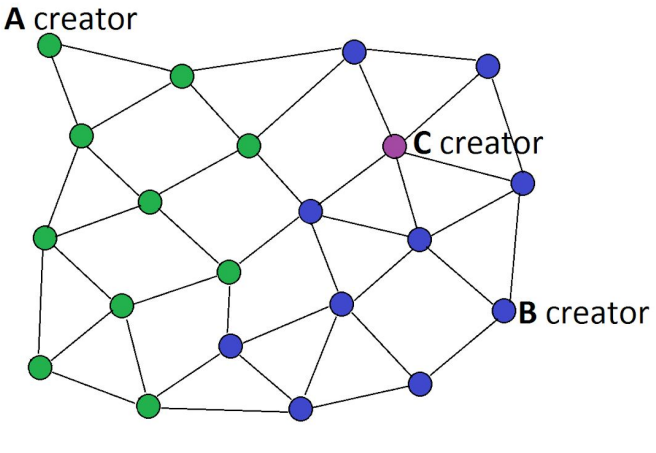


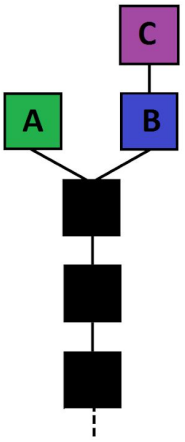
Возникнет «неразбериха» какой блок использовать за истинный, ведь в разные блоки могут быть включены разные транзакции.

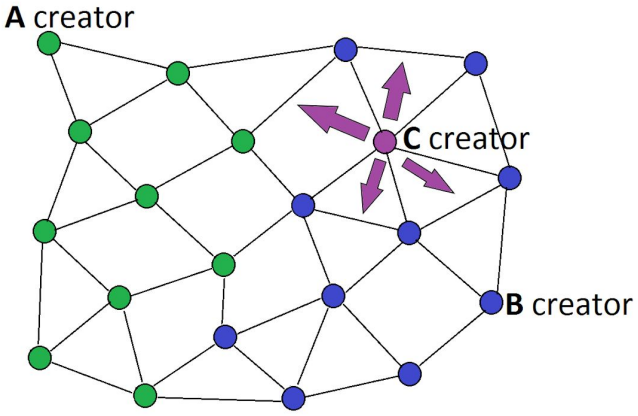
Разрешается эта проблема таким правилом: Актуальной версией цепи считается та, которая длинее, при одинаковой длине – та которая пришла раньше. Таким образом в этом случае сеть делится на части с разным верхним блоком.

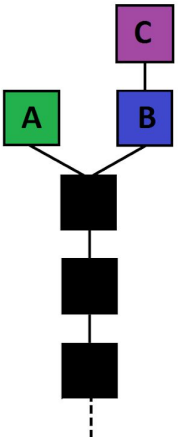
И затем, первый кто смайнит следующий блок и определит какую ветку считать за истинную, так как его версия цепи остальными пользователями примется за истинную по правилу наибольшей длины.

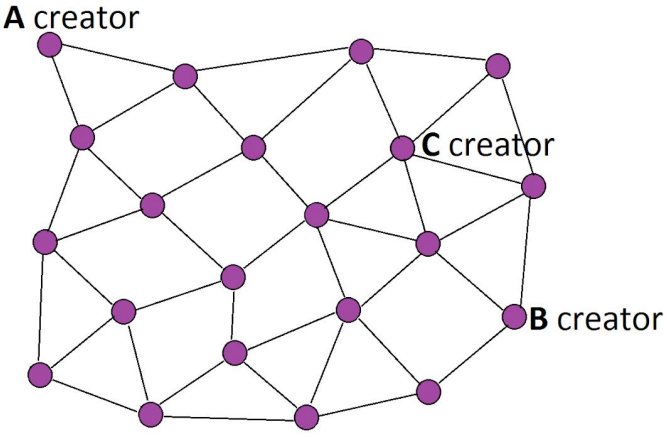

A creator
B creator

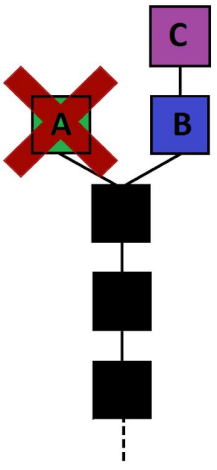

A creator
C creator
B creator


C
A
B


A creator
C creator
B creator


C
A
B


A creator
C creator
B creator


C
A
B

Однако, что мешает злоумышленнику сфальсифицировать несколько новых блоков и разослать их как валидные блоки наибольшей длины? А мешает ему механизмы заверения блоков. Существует 2 основных алгоритма консенсуса:

Proof-of-Work и Proof-of-Stake.

Суть их в том, чтобы создать блок мог не кто-попало, а какой-либо «уважаемый» или «доверенный» участник сети. Однако, как можно доверять кому-либо в анонимной сети?

Для этого анонимному пользователю нужно как-то доказать свою уважаемость, делается это экономическим фактором. Тут и вступают в силу алгоритмы консенсуса: Первый - Proof-of-Work, заключается в том, что майнер создающий блок, должен выполнить какую-либо сложную задачу, дабы доказать, что он обладает дорогостоящим оборудованием. Второй - Proof-of-Stake, заключается в том, что приоритет заверения блока отдается тому, кто обладает большим виртуальным капиталом.

На технологии блокчейна основывается так же еще одна важная технология – Smart contracts

Smart contract (умный контракт) – это алгоритм, предназначенный для заключения и поддержания коммерческих контрактов в технологии блокчейн. Преимущество таких «умных» сделок в том, что их не контролирует человек или организация. После того, как открытый код умной сделки сохранен в блокчейн как транзакция – он начинает быть активным, и его уже практически невозможно деактивировать. Его исполнение является автономным и полностью прозрачным. В частности, это выполнение не может быть отменено или изменено, и слежение за ним является общедоступным. Умный контракт может отправлять, получать и хранить деньги. Он также может взаимодействовать с другими смарт-контрактами или любыми вычислительными системами, подключенными к Интернету.

Существует множество реализаций, однако рассмотрим одну из наиболее популярных реализаций на платформе Etherium (Эфириум). Его главное преимущество – в том, что блокчейн и фреймворк для написания контрактов уже запущен, пользуется успехом и позволяет создавать очень гибкие контракты. В этой системе используется внутренняя криптовалюта – Ether (Эфир).

Ethereum взял идею блокчейна за основу, и применил для решения более широкого класса задач. Гарантировать не только валидность денежных переводов, но и вообще любых условий и сделок. И даже автоматизировать создание таких условий по средством написания кода на JavaScript подобном языке, называемым Solidity.

В Etherium отправка смартконтракта в блокчейн, а также вызовы методов существующих контрактов выполняются за внутресетевую одноименную валюту. Так же, стоимость отправки и исполнения кода контрактов напрямую зависит от количества тактов процессора, которые придётся майнеру потратить на осуществление транзакции и характеризуется величиной Gas (Газ).

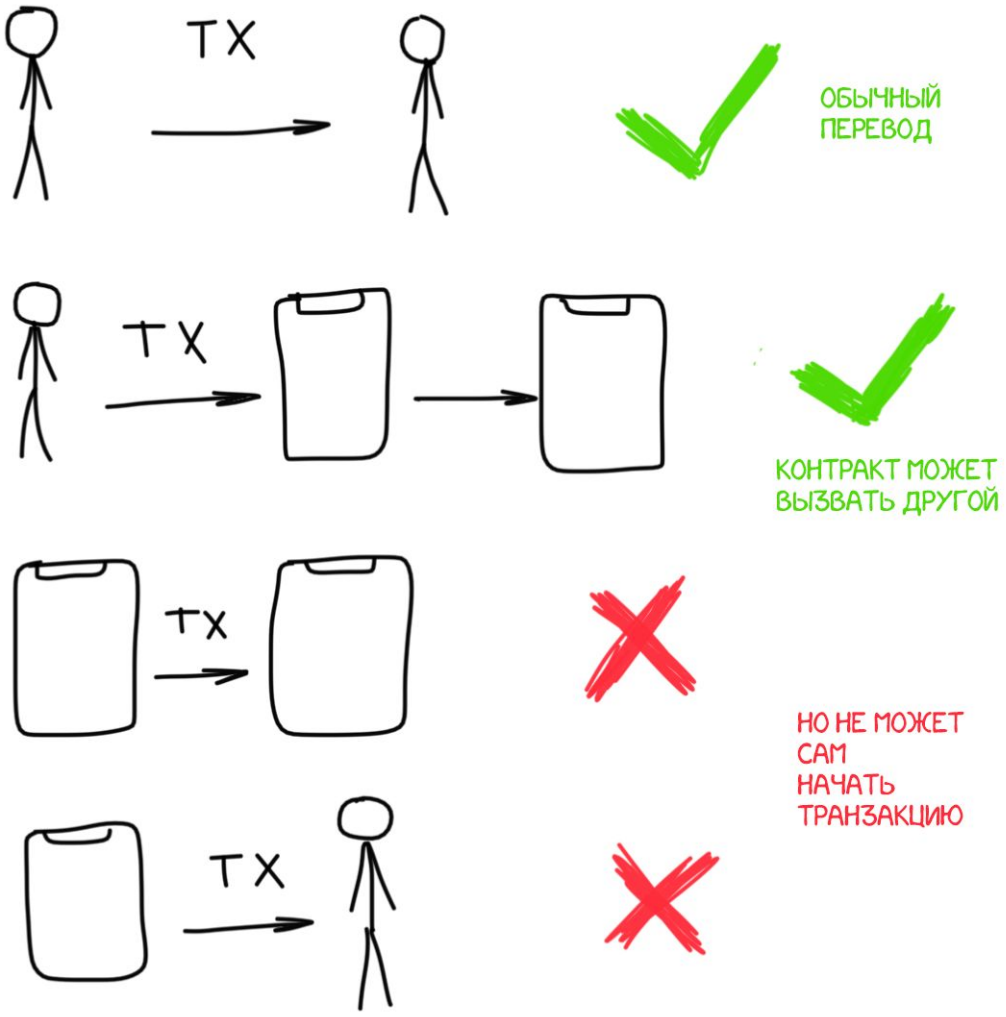
В Etherium существует 2 типа аккаунтов – кошельки и контракты. Кошельки (личные кошельки пользователей) могут управлять приватными ключами, создавать транзакции и хранить финансы на балансе пользователя. Контракты могут управлять собственным кодом, создавать транзакции только в ответ на входящие транзакции и хранить финансы на балансе контракта, которыми распоряжаться может только этот контракт.

Общение с обоими типами аккаунтов возможно только с помощью транзакций:

- Транзакция на кошелек пользователя — это перевод средств. Перевод включает в себя количество перечисляемых финансов и адрес получателя.

- Транзакция на контракт — это вызов его метода, потому её принято называть «сообщением». В неё, кроме количества и адреса контракта, включаются еще и дополнительные параметры вызова и Газ за исполнение кода.

- Транзакция без получателя — это создание смарт-контракта. В такой транзакции обязательно нужно передать скомпилированный байт-



конструктора).

У Etherium важная особенность: в контрактах невозможны таймеры, срабатывающие по истечению какого-то времени. Контракт может быть вызван только транзакцией, а их всегда запускает живой человек. «В фоне» контракт работать не умеет, но, если его вызвали — он вполне может вызвать и другой контракт.

Так же, из всего вышесказанного следует что смартконтракты, и Etherium в частности подразумевают совершенно другую парадигму разработки, нежели принятая в современном мире, описанной в книге «Move Fast and Break Things: How Facebook, Google, and Amazon Cornered Culture and Undermined Democracy» by Jonathan Taplin (Пошевеливайся и круши: Как Google, Facebook и Amazon загнали культуру в угол и подорвали демократию) где

подразумевается что нужно в начале нового проекта необходимо совершать быструю и нестабильную разработку ПО, а затем, уже после релиза выпускать обновления устраняющие ошибки и уязвимости

Проблема 15 секунд

В биткоине сложность специально установлена довольно большой, чтобы в среднем по сети блоки находились раз в 10 минут. В Ethereum же новые блоки в блокчейн создаются раз в 15 секунд, а распространяются по всей сети примерно за 12 секунд. Приводит это к тому, что блокчейн чаще обычного находится в расщепленном состоянии — никто не может с уверенностью сказать какой из последних блоков верный, пока не найдут следующий.

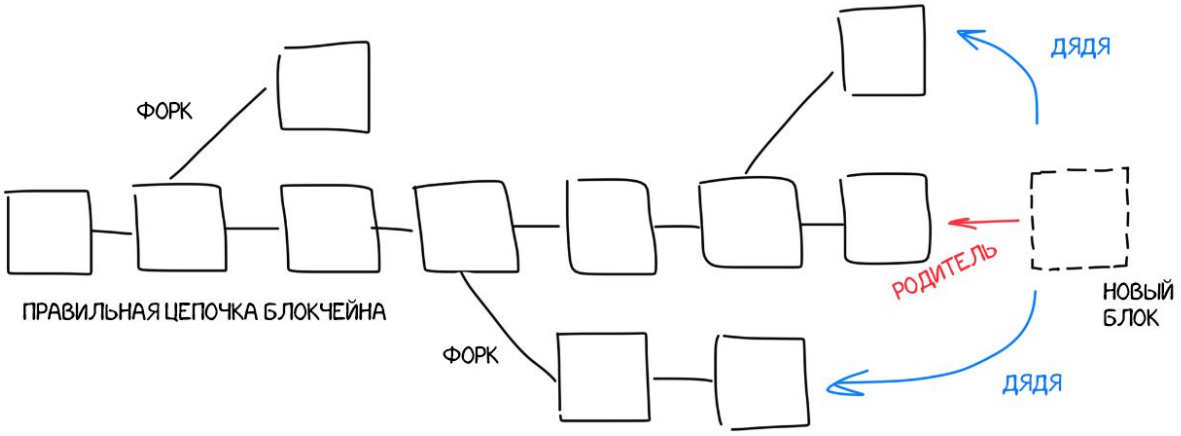
Однако правило длиннейшей цепи действует, и как только одна из цепочек блокчейна становится длиннее остальных, она принимается как единственно верная. Только при 15-секундном майнинге таких конкурирующих цепочек может появиться настолько много, что расщепленный блокчейн может жить часами, и в итоге откатывать и заново майнить большую часть транзакций.

Это не только неудобно (когда вы пытаетесь расплатиться эфирами и ждете подтверждения по пол часа), но и таит серьезную опасность. Если майнеры большую часть времени тратят на то, чтобы майнить блоки в «ненужных» цепочках, то у них появляется мотивация объединиться в пул, где майнить вместе одну определенную цепочку, тем самым повышая шансы на успех и вознаграждение.

Мотивация объединяться в пулы приводит блокчейн к возможности «атаки 51%» — когда больше 50% мощности сосредоточено в руках одного управляющего пулом. Имея такую мощность, он сможет изменить историю блокчейна и откатить транзакции, о чем простые майнеры узнаю только из постов на Реддите.

Поэтому в 2013 году был предложен модифицированный алгоритм GHOST — Greedy Heaviest Observed Subtree (Жадный поиск самой большой цепочки). Кроме понятия предыдущего блока («родителя»), он вводит понятие «дяди» блока (uncle или ommer).

GHOST несёт простой смысл: давать небольшое вознаграждение в том числе тем майнерам, которые нашли «дядю» — логически верный блок, которому просто не повезло оказаться в соседней цепочке. Дяде дают 12.5% от цены полноценного блока — это мотивирует майнеров продолжать майнить самостоятельно, ведь на поиске «дядь» тоже можно неплохо заработать.



РАНЬШЕ БЫЛИ БЫ ПРОСТО ОТБРОШЕНЫ, НО ТЕПЕРЬ ИМЕЮТ ШАНС ПОЛУЧИТЬ 12.5% ВОЗНАГРАЖДЕНИЯ

Предложения по практическому применению.

Прозоро – перенести платформу на смарт контракты и децентрализовать

База данных прав автовладельцев

Кадастровые сделки, хранение архива.