Yuamble Blockchain protocol

February 5, 2022

by Ivanov Alexandr (@iaa2005)

yuamble.org

Abstract. This document is devoted to the description of a new type of blockchain — Yuamble. Blockchain is a distributed registry that stores transactions, data on smart contracts (to be disclosed in detail later), as well as other data of users and participants of the blockchain network. This document will talk about algorithms that will reduce the size of fees on the network, as well as make the blockchain post-quantum.

Introduction. The Yuamble blockchain is very similar to the Bitcoin and Ethereum blockchain. It performs the same main functions, such as storing data about users and their transactions. But Ethereum has expanded the possibilities of using blockchain – this is the use of smart contracts.

Smart contracts are a computer program that monitors and ensures the fulfillment of obligations. The parties prescribe in it the terms of the transaction and sanctions for their non-fulfillment, put digital signatures. A smart contract independently determines whether everything has been executed and makes a decision: to complete the transaction and issue the required (money, shares, real estate), impose a fine or penalty on the participants, close access to assets. More about Smart Contracts on ethereum.org.

But often the blockchain network consisting of these smart contracts is very heavily loaded. Network participants who send signed transactions to the pool must pay huge fees to reduce the processing time of this transaction, and so that the transaction gets into the block — the main part of the distributed registry as soon as possible. To solve this problem, we are making some changes to the protocol — the blockchain algorithm.

Proof-of-stake (PoS). This consensus protocol will be used in the Yuamble blockchain. This method of protection in cryptocurrencies, in which the probability of the formation of the next block in the blockchain by the participant is proportional to the share that the account units of this cryptocurrency belong to this participant from their total number. This method is an alternative to the proof of work (PoW) method, in which the probability of creating the next block is higher for the owner of more powerful equipment.

When using this method, the block formation algorithm does not depend on the capacity of the equipment, but the block is more likely to be formed by the account with the largest current balance. For example, a participant who owns 1% of the total amount will, on average, generate 1% of new blocks. More about PoS on en.wikipedia.org.

Using the digital signature of Winternitz. The Winternitz signature (W-OTS, Winternitz One Time Signature) is an improved algorithm for the digital signature of a Lamport. This algorithm signs transactions quickly. It is post-quantum, i.e., quantum computers and their algorithms will not be able to find a private key to a public one, forge a transaction signature and pass it off as the real account owner. More about W-OTS you can find in this file or this.

Duescript — smart contract programming language. Duescript is an object-oriented, high-level language for implementing smart contracts. Smart contracts are programs which govern the behaviour of accounts within the Ethereum state. Duescript is statically typed, supports inheritance, libraries and complex user-defined types among other features. With Duescript you can create contracts for uses such as voting, crowdfunding, blind auctions, and multi-signature wallets. The programming language is very similar to Javascript and C++. Therefore, the syntax will be well understood by programmers who know these languages.

Make processing of smart contracts without fees. Most blockchains process transactions only with the payment of a commission. Due to the payment of the fee, the number of participants in the network who sign transactions and send them to the pool decreases over time, because

the size of the fee increases with the demand for transaction processing. We made the main decision — to remove fees on transactions and calls of smart contracts under one condition: if the coefficient k is the probability coefficient of looping, it will be approximately zero.
This coefficient k shows how secure the code (or part of the code, function, method) written by the creator of the smart contract is for nodes participating in the Yuamble network. Since in the presence of malicious code, Yuamble virtual machines (YVM, an analogue of the Ethereum Virtual Machine EVM) can go into a resource-draining endless cycle. There are several ways to provide a way to terminate the contract from the outside and avoid entering into a resource-draining infinite loop:

1. *Turing incompleteness:* Limited functionality will not allow jumping and/or loops. Therefore, the smart contract will not be able to enter an infinite loop.

2. *Step and Cost Meter:* The program can simply track the number of commands executed, and then shut down after completing a certain count of steps. Another method is a counter. Here contracts are executed with prepayment. A certain amount is required to complete each instruction. If the fee paid exceeds the prepaid fee, the contract is terminated.

3. *Timer:* If the execution of the contract does not meet a certain deadline, then it is terminated forcibly.

We will use *Turing Incompleteness* and a *Timer*, because the *Step and Cost Meter*, as in Ethereum, will not work if we have transaction processing without fees and gas.
If the k coefficient is greater than 0.0, the Yuamble network participant will need to sign the transaction and pay the commission, as in the Ethereum network. To avoid such payments, the network participant needs to carefully read the smart contract code and its k coefficient. In order for smart contracts to be processed without commission, developers will try to write the contract code as concisely as possible and use the while, do while and for loops as little as possible. Otherwise, the k coefficient of the smart contract will increase after each processing of the transaction that triggers this contract. And this leads to a decrease in contract customers.

Decentralized data from Off-chain and centralized Internet. It is a technology for creating a global decentralized network of oracles running on countless computers to provide reliable and real data for smart contracts running on the Yuamble blockchain.
Oracles are real data points that can connect to blockchain-based smart contracts. Each oracle in the Yuamble decentralized network has an incentive to provide accurate data, while each of the oracles has its own reputation (or rating). When oracles follow the rules of the software and provide useful (accurate) data, they receive as a reward YMB — their own Yuamble coin.
An important advantage of the technology is that the system is completely decentralized, so the data is approved by all nodes of the network before it gets to the end user. More about blockchain oracles.

What will the blockchain protocol be written on? The entire protocol will be written in Rust, C++. Rust provides great advantages. It is convenient to write code on it and connect important libraries and modules for working with the network, digital signatures and smart contract processing.

And so, we believe that this project will provide huge opportunities in the field of decentralized finance, banking, programming and cryptography. This project will overestimate the possibilities of the blockchain and make the world of censorship stable and private for the participants of the blockchain network.

Yuamble Blockchain protocol

February 5, 2022

by Ivanov Alexandr (@iaa2005)

yuamble.org

Краткое изложение. Этот документ посвящен описанию нового типа блокчейна — Yuamble. Блокчейн — это распределенный реестр, в котором хранятся транзакции, данные о смарт-контрактах (будут подробно раскрыты позже), а также другие данные пользователей и участников блокчейн-сети. В этом документе будет рассказано об алгоритмах, которые позволят снизить размер комиссий в сети, а также сделают блокчейн постквантовым.

Введение. Блокчейн Yuamble очень похож на блокчейн Биткоина и Эфириума. Он выполняет те же основные функции, такие как хранение данных о пользователях и их транзакциях. Но Ethereum расширил возможности использования блокчейна — это использование смарт-контрактов.

Смарт-контракты — это компьютерная программа, которая контролирует и обеспечивает выполнение обязательств. Стороны прописывают в нем условия сделки и санкции за их невыполнение, ставят цифровые подписи. Смарт-контракт самостоятельно определяет, все ли было выполнено, и принимает решение: завершить транзакцию и выдать требуемое (деньги, акции, недвижимость), наложить штраф или неустойку на участников, закрыть доступ к активам. Подробнее о смарт-контрактах на ethereum.org.

Но часто блокчейн-сеть, состоящая из этих смарт-контрактов, очень сильно загружена. Участники сети, отправляющие подписанные транзакции в пул, должны платить огромные комиссионные, чтобы сократить время обработки этой транзакции и чтобы транзакция как можно скорее попала в блок — основную часть распределенной регистрации. Чтобы решить эту проблему, мы вносим некоторые изменения в протокол — алгоритм блокчейна.

Доказательство доли владения, Proof-of-stake (PoS). Этот консенсусный протокол будет использоваться в блокчейне Yuamble. Это метод защиты в криптовалютах, при котором вероятность формирования участником следующего блока в блокчейне пропорциональна доле, которую составляют учетные единицы этой криптовалюты, принадлежащие данному участнику от их общего количества. Этот метод является альтернативой методу доказательства работы (PoW), при котором вероятность создания следующего блока выше для владельца более мощного оборудования.

При использовании этого метода алгоритм формирования блока не зависит от мощности оборудования, но блок с большей вероятностью будет сформирован счетом с наибольшим текущим балансом. Например, участник, которому принадлежит 1% от общей суммы, в среднем будет генерировать 1% новых блоков. Подробнее о PoS на en.wikipedia.org.

Использование цифровой подписи Винтерница. Подпись Winternitz (W-OTS, Одноразовая подпись Винтерница) представляет собой улучшенный алгоритм цифровой подписи Лэмпорта (ссылка). Этот алгоритм быстро подписывает транзакции. Это постквантовый, то есть квантовые компьютеры и их алгоритмы не смогут найти закрытый ключ к открытому, подделать подпись транзакции и выдать ее за реального владельца учетной записи. Подробнее о W-OTS вы можете найти в этом файле или здесь.

Duescript — язык программирования смарт-контрактов. Duescript — это объектно-ориентированный язык высокого уровня для реализации смарт-контрактов. Смарт-контракты — это программы, которые управляют поведением учетных записей в государстве Ethereum. Duescript статически типизирован, поддерживает наследование, библиотеки и сложные пользовательские типы, среди прочих особенностей. С помощью Duescript можно создавать контракты для таких целей, как голосование, краудфандинг, слепые аукционы и кошельки с несколькими подписями. Язык программирова-

ния очень похож на Javascript и C++. Поэтому синтаксис будет хорошо понятен программистам, знающим эти языки.

Сделать обработку смарт-контрактов без комиссий. Большинство блокчейнов обрабатывают транзакции только с выплатой комиссии. Из-за уплаты комиссии количество участников сети, которые подписывают транзакции и отправляют их в пул, со временем уменьшается, поскольку размер комиссии увеличивается вместе со спросом на обработку транзакций. Мы приняли главное решение — убрать комиссии за транзакции и вызовы смарт-контрактов при одном условии: если коэффициент k является коэффициентом вероятности зацикливания, он будет примерно равен нулю.
Этот коэффициент k показывает, насколько безопасен код (или часть кода, функция, метод), написанный создателем смарт-контракта, для узлов, участвующих в сети Yuamble. Так как при наличии вредоносного кода виртуальные машины Yuamble (YVM, аналог виртуальной машины Ethereum EVM) может перейти в бесконечный цикл истощения ресурсов. Существует несколько способов обеспечить возможность расторгнуть контракт извне и избежать попадания в бесконечный цикл, истощающий ресурсы:

1. *Неполнота Тьюринга:* Ограниченная функциональность не позволит совершать прыжки и/или циклы. Следовательно, смарт-контракт не сможет войти в бесконечный цикл.

2. *Счетчик шагов и затрат:* Программа может просто отслеживать количество выполненных команд, а затем завершать работу после выполнения определенного количества шагов. Другой метод — это счетчик. Здесь контракты заключаются с предоплатой. Для выполнения каждой инструкции требуется определенная сумма. Если уплаченная сумма превышает предоплаченную, договор расторгается.

3. *Счетчик времени:* Если исполнение контракта не укладывается в определенный срок, то он расторгается в принудительном порядке.

Мы будем использовать *Неполноту Тьюринга* и *Счетчик времени*, потому что *Счетчик шагов и затрат*, как в Ethereum, не будет работать, если у нас есть обработка транзакций без комиссий (сборов) и газа (ссылка).
Если коэффициент k больше 0, участнику сети Yuamble необходимо будет подписать транзакцию и оплатить комиссию, как в сети Ethereum. Чтобы избежать подобных выплат, участнику сети нужно внимательно ознакомиться с кодом смарт-контракта и его коэффициентом k. Чтобы смарт-контракты были обработаны без комиссии, разработчики будут стараться писать код контракта как можно лаконичнее и как можно меньше использовать циклы while, do while и for. В противном случае коэффициент k смарт-контракта будет повышаться после каждой обработки транзакции, которая вызывает этот контракт. А это приводит к уменьшению клиентов контракта.

Децентрализованные данные из автономного и централизованного Интернета. Это технология создания глобальной децентрализованной сети оракулов, работающих на бесчисленных компьютерах, для предоставления надежных и реальных данных для смарт-контрактов, работающих на блокчейне Yuamble. Оракулы — это реальные точки данных, которые могут подключаться к смарт-контрактам на основе блокчейна. У каждого оракула в децентрализованной сети Yuamble есть стимул предоставлять точные данные, в то время как у каждого из оракулов есть своя собственная репутация (или рейтинг). Когда оракулы следуют правилам программного обеспечения и предоставляют полезные (точные) данные, они получают в качестве награды YMB — свою собственную монету Yuamble.
Важным преимуществом технологии является то, что система полностью децентрализована, поэтому данные утверждаются всеми узлами сети до того, как они попадут к конечному пользователю. Подробнее об оракулах блокчейна.

На чем будет написан протокол блокчейна? Весь протокол будет написан на Rust, C++. Rust дает большие преимущества. На нем удобно писать код и подключать важные библиотеки и модули для работы с сетью, цифровых подписей и обработки смарт-контрактов.

Подведем итог. Мы считаем, что этот проект предоставит огромные возможности в области децентрализованных финансов, банковского дела, программирования и криптографии. Этот проект переоценит возможности блокчейна и сделает мир цензура устойчивым и приватным для участников блокчейн-сети.