Блокчейн как криптографический примитив

Н. П. Варновский

***Аннотация*—В работе блокчейн рассматривается как новый криптографический примитив, представляющий собой особую базу данных, записи в которой линейно упорядочены, а допустимые запросы к ней могут быть лишь двух видов — запросы на чтение записей (любым желающим) и на добавление записи в конец базы (пользователем, выполнившим определённые условия). При этом блокчейн должен удовлетворять требованиям живучести и незыблемости, гарантирующим добавление любой записи, коль скоро она снабжена необходимым подтверждением права на это действие, и «вечную сохранность» раз добавленной записи.**

Одним из основных является вопрос о принципиальной возможности построения такого криптографического примитива. В статье обсуждается проблема обоснования существования блокчейна в различных моделях, позволяющих использовать (гипотетически) этот примитив для защиты информации. Известные положительные результаты не дают удовлетворительного ответа на данный вопрос. В частности, все они получены в модели со случайным оракулом.

В настоящей работе основное внимание уделено блокчейнам, отвечающим парадигме Proof-of-Work, в конструкции которых существенным образом используются криптографические хэш-функции. При сравнительно сильных предположениях доказан следующий отрицательный результат: при условии существования семейств хэш-функций с труднообнаружимыми коллизиями всегда найдутся такие семейства, на которых нельзя строить PoW-блокчейны. Как следствие, не существует blackbox-конструкции PoW-блокчейна на основе семейства хэш-функций с труднообнаружимыми коллизиями.

***Ключевые слова*—блокчейн, защита информации, криптографические хэш-функции, криптографический примитив, модель со случайным оракулом, blackbox-конструкции, Proof-of-Work**

I. Введение

Посвящённая блокчейнам литература весьма обширна, и практически всегда в ней используется словосочетание «технология блокчейн». Однако, строго говоря, данный термин некорректен. Очевидно, что ни конкретные криптографические конструкции, ни способы их реализации в конкретных моделях не относятся к категории технологий. При этом понятие криптографического примитива здесь более чем уместно.

По своей структуре блокчейн — это особая база данных, состоящая из линейно упорядоченных записей и допускающая только два вида запросов:

- запрос на чтение любой записи — доступен любому пользователю;
- запрос на добавление записи в конец базы данных — право пользователей на это действие определяется способом реализации блокчейна и конкретным приложением.

Статья получена 17 ноября 2020.

Николай Павлович Варновский, Институт проблем информационной безопасности МГУ им. М. В. Ломоносова, (email: math-dep@iisi.msu.ru).

Собственно криптографическим примитивом, называемым блокчейном, такая база данных является при выполнении следующих двух основных требований.

Первое условие (требование) связано с обеспечением свойства *незыблемости* (persistency). Содержательно оно означает, что никакая запись базы данных не может быть изменена или удалена. Более того, порядок записей зафиксирован раз и навсегда.

Замечание 1. Сто́ит отметить, что указанное свойство имеет и негативный эффект в плане практического применения соответствующих систем. Специалистам по базам данных хорошо известно, что контроль входных данных представляет собой серьёзную научно-техническую проблему. Исправление ошибок в данных требует существенных затрат. В случае же блокчейна эта проблема и вовсе неразрешима: ошибки во внесённых записях исправить невозможно по определению.

Второе условие — обеспечение свойства *живучести* (liveness): если подан запрос на добавление (в конец базы данных) записи, удовлетворяющей всем необходимым требованиям, то эта запись будет непременно добавлена в базу с задержкой, не превосходящей определённую величину. Требования к записи и определение допустимой задержки задаются спецификациями конкретных реализации и приложения блокчейна.

Модели, в которых анализируются блокчейны, делятся на две больших категории — с контролем доступа (permissioned) и без контроля доступа (permissionless). В последнем случае добавлять записи в базу данных может любой пользователь, выполнивший все необходимые действия, предписанные протоколом. В первом же случае добавлять записи могут только те пользователи, которые получили на это соответствующее разрешение (это можно отнести к тем требованиям к записи, которые упоминались выше) от некоторого центрального органа управления и контроля, который, следовательно, должен быть предусмотрен в системе.

В теоретических и прикладных исследованиях систем, связанных с защитой информации, рассматриваются центральные органы следующих трёх основных типов.

Первый тип представляет выделенный участник (пользователь), выполняющий особые функции. Вопреки распространённому мнению, такой участник необходим и для блокчейнов без контроля доступа: кто-то должен, как минимум, инициировать процесс и задать начальное состояние блокчейна. С точки зрения криптографии, нельзя не учитывать возможность коррумпированности этого участника, однако сценарий с таким противником в литературе не рассматривается.

Ко второму типу относится центр доверия, который представляет собой участника, наделённого доверием всех остальных пользователей. Стойкость соответствую-

щей системы защиты информации анализируется только в предположении, что центр доверия — честный. Если же центр доверия отклоняется от предписанного поведения, должна иметься возможность доказать этот факт некоему арбитру.

Третий тип центрального органа — большой брат (Big Brother): это центр доверия, действия которого не подвергаются контролю со стороны других пользователей, его нечестность недоказуема.

Следует отметить, что в предполагаемых приложениях практически всегда присутствует какой-либо центральный орган, которому пользователи вынуждены доверять выполнение тех или иных функций. В связи с этим важно для конкретного приложения оценить степень доверия участников этому органу и проанализировать возможность реализации блокчейна в модели с центром доверия соответствующего вида. Естественно предположить, что в таком случае (когда, в отличие от первоначальной задумки, не имеет места полная децентрализация сети) для построения блокчейнов было бы достаточно использовать уже известные — и хорошо изученные — криптографические средства. Однако эта проблематика остаётся до сих пор неисследованной.

По-видимому, только одно разумное приложение, реализуемое в модели без контроля доступа, можно выделить в имеющейся научной литературе по блокчейнам. Речь идёт о решении проблемы приоритета: представляется целесообразным использовать блокчейн для фиксации информации о научных открытиях, изобретениях и т. п.

Первоначальная и наиболее популярная идея создания блокчейна без контроля доступа основана на майнинге и протоколе консенсуса. Всякий желающий может добавить новую запись в базу данных, если решит так называемую умеренно трудную вычислительную задачу, которая определяется некоторой фиксированной криптографической хэш-функцией H. Запись r помещается в блок, содержащий также h_{-1} — хэш-значение предыдущего блока — и некоторое специальное значение η. Умеренно трудная вычислительная задача состоит в поиске такого значения η, что выполнено неравенство $H(h_{-1}, r, \eta) < D_p$, где D_p — параметр, определяющий трудность этой задачи. Процесс поиска такого значения η называется майнингом. На основе общих теоретических предположений о свойствах хэш-функций считается, без всякого обоснования, что параметр D_p можно задать таким образом, что при случайном выборе η указанное неравенство выполняется с вероятностью p.

Благодаря включению в каждый добавляемый блок ссылки на предыдущий (h_{-1}) весь массив блоков оказывается объединённым в цепочку, что, собственно, и породило (в английском варианте) термин «блокчейн».

Каждый участник хранит у себя текущее состояние цепочки блоков и в случае успеха процедуры майнинга (положительного завершения протокола консенсуса), добавляет новый блок в конец цепочки и рассылает её всем участникам сети с помощью специального протокола (broadcast).

Этот подход к созданию блокчейнов основан на так называемой парадигме проверки выполнения вычислительной работы (proof of work). Он предполагает, что честные участники контролируют бóльшую часть вычислительных ресурсов сети.

Исследования PoW-блокчейнов находятся пока на своём начальном этапе. Имеющиеся результаты, касающиеся вопросов существования такого криптографического примитива, можно трактовать лишь как частичное обоснование PoW-блокчейнов. Подробнее об этом будет сказано ниже, в разделе III.

В литературе нередко встречаются утверждения, смысл которых сводится к следующему: поскольку так называемые криптовалюты используют PoW-блокчейны и существуют годами, такие блокчейны достаточно надёжны. Но следует заметить, что в некоторых теоретических работах, посвящённым криптовалютам, используются математические методы теории игр и исследования операций, а не математической криптографии. Поскольку в случае криптовалюты единственной целью участников является прибыль, вполне разумным представляется предположение о том, что подавляющее большинство участников использует рациональные стратегии. Однако оно неприменимо к криптографическому примитиву блокчейн, предназначенному для защиты информации (в определённом смысле).

II. Математическая модель блокчейна

Блокчейн как криптографический примитив определяется четвёркой следующих объектов: E — множество возможных (допустимых) записей, представляющих собой битовые строки конечной длины, $E \subseteq \{0,1\}^*$; P — эффективно вычислимый трёхместный предикат; $\mathcal{R}$ — эффективный алгоритм, выполняющий запросы на чтение записей в блоках (этот компонент может отсутствовать: в предлагаемых реализациях, как правило, записи в блокчейне общедоступны); $\mathcal{S}$ — эффективный алгоритм, выполняющий запросы на добавление записей в конец базы данных. Запись r, данная в запросе, добавляется при выполнении условия $P(state, r, proof) = 1$, где $state$ — состояние блокчейна на момент получения запроса (в общем случае под состоянием подразумевается вся цепочка блоков), $proof$ — зависящая от типа блокчейна дополнительная информация, которая играет роль подтверждения права пользователя на добавление данной записи.

Например, в общей схеме PoW-блокчейна необходимым и достаточным условием равенства $P(state, r, \eta) = 1$ (и, соответственно, возможности добавить запись) выступает выполнение неравенства $H(h_{-1}, r, \eta) < D_p$ и отношения $r \in E$ (в данном случае h_{-1} определяется значением $state$). Если рассматривается модель с контролем доступа, следует добавить и условие, проверяющее полномочия данного абонента на добавление записей.

Требование незыблемости блокчейна определяется следующим образом. Положим, $state(t)$ — состояние блокчейна в момент времени t, представляющее собой последовательность имеющихся в нём на данный момент блоков. Тогда, для любых t_1, t_2 с условием $t_1 < t_2$ последовательность $state(t_1)$ должна быть префиксом последовательности $state(t_2)$.

Второе требование — живучесть — означает существование такого числового параметра Δ, что если в момент времени t выдан запрос на добавление записи r и предъявлено доказательство $proof$, удовлетворяющее условию

$P(state, r, proof) = 1$, то в любой момент времени t', $t' > t + \Delta$, блокчейн (его состояние) будет содержать запись r.

Приведённые формулировки — это метаопределения, поскольку для полной формализации понятий незыблемости и живучести необходима модель противника, а она существенно зависит от конкретной реализации.

III. Вопрос существования PoW-блокчейнов

Далее будем рассматривать основной вариант реализации блокчейна, с майнингом и протоколом консенсуса.

После того как введены все необходимые определения, важнейшим теоретическим вопросом становится обоснование существования данного примитива. В работе [1] при определённых предположениях и допущениях доказана теорема существования для PoW-блокчейнов. Подобного рода результаты можно найти и в некоторых других публикациях.

Однако, строго говоря, вопрос о существовании PoW-блокчейнов всё же остаётся открытым: все известные положительные результаты доказаны в модели со случайным оракулом, когда хэш-функция заменяется оракулом, вычисляющим случайную функцию, а это допущение противоречит предположениям о блокчейне как о средстве защиты информации в моделях, где нет никаких центральных органов.

Очевидно, свойства блокчейна тесно связаны со свойствами составляющих его компонентов. И в рассматриваемом случае PoW-блокчейнов основной целью соответствующей теории может ставиться доказательство их существования в предположении о стойкости используемой хэш-функции. Однако здесь возникают определённые проблемы, первая из которых — обоснование существования самой криптографически стойкой хэш-функции. На данный момент в математической криптографии достаточно хорошо обоснованы лишь односторонние семейства хэш-функций.

Рассмотрим подробнее некоторые варианты определения семейств хэш-функций. Для этого предполагаем, что для каждого натурального n задано некое множество функций $H_n = \{h\colon \Sigma^n \to \Sigma^m\}$, где $m = m(n) < n$ и Σ обозначает множество (алфавит) $\{0, 1\}$.

Определение 1. Пусть для каждого натурального n задано $H_n = \{h\colon \Sigma^n \to \Sigma^m\}$, где $m < n$. Семейство $\{H_n\}$ называется *односторонним семейством хэш-функций*, если выполнены следующие условия.

1) Существует полиномиальная вероятностная машина Тьюринга G, такая, что $G(1^n) = \bar{h}$, где $\bar{h}$ — элемент множества $\bar{H}_n$ описаний функций из H_n.
2) Существует полиномиальная машина Тьюринга C, такая, что $C(\bar{h}, x) = h(x)$ для любого $\bar{h} \in \bar{H}_n$ и любого $x \in \Sigma^n$.
3) Для любой полиномиальной вероятностной машины Тьюринга A, которая, получив на первом шаге на вход 1^n, выдаёт начальное значение $z \in \Sigma^n$, справедливо следующее: пусть $\bar{h}$ — описание хэш-функции, сгенерированное машиной G на входе 1^n, тогда для любого полинома p и всех достаточно больших n

$$\Pr[A(\bar{h}) = y\colon y \neq z,\ h(z) = h(y)] \leq \frac{1}{p(n)}.$$

Известно, что такое семейство можно построить из произвольной односторонней функции. Более проблематичным является семейство хэш-функций с труднообнаружимыми коллизиями. В определении этого примитива первые два пункта такие же, как и для одностороннего семейства хэш-функций, поэтому мы их опустим, о приведём только третий.

Определение 2. Семейство $\{H_n\}$ называется *семейством хэш-функций с труднообнаружимыми коллизиями*, если для любой полиномиальной вероятностной машины Тьюринга A, для любого полинома p и всех достаточно больших n справедливо неравенство

$$\Pr[A(\bar{h}) = (z, y),\ y \neq z,\ h(z) = h(y)] \leq \frac{1}{p(n)},$$

где $\bar{h}$ — описание хэш-функции, сгенерированное машиной G на входе 1^n.

Уточним некоторые особенности используемой далее модели. Во-первых, оговорим, каким образом генерируются описания конкретных хэш-функций из семейства. Для этих целей мы дополняем модель специальным оракулом, который выбирает описания хэш-функций. Поскольку результат (теорема 1 ниже) отрицательный, такое расширение модели его только усиливает.

Во-вторых, если описание хэш-функции выбирается уже после того, как зафиксирована подготовленная для добавления в блокчейн запись r (например, в модели, где r служит запросом к оракулу), то требуются семейства, занимающие промежуточное положение между двумя определёнными выше семействами хэш-функций. Такие семейства были введены в работе [2] и, независимо, в работе [3]. Как и в предыдущем случае, мы опускаем в определении пункты 1 и 2.

Определение 3. Пусть $\alpha : \mathbb{N} \to \mathbb{N}$ — функция, вместе со своей обратной вычислимая за полиномиальное время и удовлетворяющая условию $\alpha(n) \leq n$ для любого n. Семейство $\{H_n\}$ называется CR_α*-семейством хэш-функций*, если для любой полиномиальной вероятностной машины Тьюринга A, которая, получив на первом шаге на вход 1^n, выдаёт начальное значение $z \in \Sigma^{\alpha(n)}$, справедливо следующее: пусть $\bar{h}$ — описание хэш-функции, сгенерированное машиной G на входе 1^n, тогда для любого полинома p и всех достаточно больших n

$$\Pr[A(\bar{h}, z) = (x, y),\ x = zz',\ x \neq y,\ h(x) = h(y)] \leq \frac{1}{p(n)}.$$

Для дальнейшего в дополнение к соглашению о специальном оракуле мы принимаем самое сильное из приведённых предположение — о существовании семейств хэш-функций с труднообнаружимыми коллизиями. В этих условиях справедливо следующее утверждение, которое показывает, что нельзя строить blackbox-конструкции PoW-блокчейна исходя из произвольного семейства хэш-функций такого типа.

Теорема 1. *Существуют семейства хэш-функций с труднообнаружимыми коллизиями, не позволяющие строить PoW-блокчейны.*

Доказательство. Пусть $\{H_n\}$ — семейство хэш-функций с труднообнаружимыми коллизиями, где $H_n = \{h\colon \Sigma^n \to$

$\Sigma^m\}$, $m < n$. Построим ещё одно семейство функций $\{H'_n\}$, полагая, что множества описаний функций из обоих семейств совпадают, $\bar{H}'_n = \bar{H}_n$. Пусть D_p — параметр из определения предиката P. D_p очевидным образом определяет параметр l — количество начальных нулей в «целевом» для майнинга хэш-значении, то есть l — минимальная длина состоящего из нулей префикса в хэш-значении, которое успешно пройдёт проверку, определяемую предикатом P. Для всякой функции $h \in H_n$ строим функцию $h' \in H'_n$, которая отличается от h только в случае, когда для данного аргумента x в хэш-значении $h(x)$ есть l начальных нулей, — тогда h' записывает в начальный (старший) бит единицу.

Очевидно, что $\{H'_n\}$ — семейство хэш-функций с труднообнаружимыми коллизиями. Так же очевидно, что семейство $\{H'_n\}$ не позволяет строить блокчейны, поскольку майнинг никогда не приводит к успеху. □

IV. Заключение

Понятие блокчейна с математической точки зрения наиболее адекватно вписывается в ряд криптографических примитивов. Подтверждение выполненной работы (proof of work) остаётся основным подходом к построению блокчейнов в модели без контроля доступа. Однако и для этого случая до сих пор не найдено надёжного обоснования.

Вместе с тем, как показано в данной работе, для доказательства существования PoW-блокчейна недостаточно даже сравнительно сильного предположения о наличии семейств хэш-функций с трудообнаружимыми коллизиями при условии выбора функций из семейства специальным оракулом.

Библиография

[1] Pass R., Seeman L., Shelat A. Analysis of the blockchain protocol in asynchronous networks // Advances in Cryptology—EUROCRYPT '17. — Vol. 10210 of Lecture Notes in Computer Science. — Berlin, Heidelberg : Springer, 2017. — P. 643–673.

[2] Варновский Н. П. Об определениях криптографически стойких хэш-функций. — 1998. — Рукопись.

[3] Mironov I. Hash functions from Merkle—Damgård to Shoup // Advances in Cryptology—EUROCRYPT '01. — Vol. 2045 of Lecture Notes in Computer Science. — Berlin, Heidelberg : Springer, 2001. — P. 166–181.

Blockchain as a cryptographic primitive

N. P. Varnovsky

***Abstract*—We consider blockchain as a new cryptographic primitive. This primitive is defined as an ordered database that allows only the following two types of queries: (a) read the data (by any user) and (b) add a new record to the end of the database (by a user who complied with certain requirements). A blockchain must satisfy the liveness and persistency conditions. The former condition guarantees that after a query to add a correct record, this record will eventually appear in the database. The latter one means that a record once added cannot be removed or modified.**

One of the main problems concerning blockchain is whether this primitive exists. The paper discusses this problem in various models. Known positive results do not provide a satisfactory answer to this problem. In particular, they are all proved in the random oracle model.

In this paper, we focus on Proof-of-Work (PoW) blockchains that are based on cryptographic hash functions. Our main result is that if collision-resistant hash function families exist, then there exists such a family that cannot be used in a PoW-blockchain. Therefore there is no black box construction of a PoW-blockchain from a collision-resistant hash function family.

***Keywords*—Black box construction, blockchain, cryptographic hash functions, cryptographic primitive, information security, Proof-of-Work, random oracle model**

References

[1] Pass R., Seeman L., Shelat A. Analysis of the blockchain protocol in asynchronous networks // Advances in Cryptology—EUROCRYPT '17. Vol. 10210 of Lecture Notes in Computer Science. Berlin, Heidelberg : Springer, 2017. P. 643–673.

[2] Varnovsky N. P. Ob opredeleniyakh kriptograficheski stoykikh kheshfunktsiy. 1998. Manuscript [in Russian].

[3] Mironov I. Hash functions from Merkle—Damgård to Shoup // Advances in Cryptology—EUROCRYPT '01. Vol. 2045 of Lecture Notes in Computer Science. Berlin, Heidelberg : Springer, 2001. P. 166–181.