УДК 004.056
doi: 10.26583/bit.2024.1.04

Сергей В. Запечников[1], Анатолий Ю. Конкин[2]
Национальный исследовательский ядерный университет «МИФИ»,
Каширское ш., 31, Москва, 115409, Россия
[1]e-mail: svzapechnikov@mephi.ru https://orcid.org/0000-0002-7975-6040
[2]e-mail: aykonkin@gmail.com, https://orcid.org/0000-0002-1300-6466

ОБЕСПЕЧЕНИЕ КОНФИДЕНЦИАЛЬНОСТИ ИНФОРМАЦИИ В СИСТЕМАХ РАСПРЕДЕЛЕННОГО РЕЕСТРА ПОСРЕДСТВОМ ДОКАЗАТЕЛЬСТВ С НУЛЕВЫМ РАЗГЛАШЕНИЕМ

Аннотация Статья посвящена актуальной проблеме обеспечения конфиденциальности в системах распределенного реестра. Рассматривается прикладная задача обеспечения конфиденциальности данных при операциях с цифровыми финансовыми активами. Представлено сравнение различных методов обеспечения конфиденциальности, включая перемешивающие сети, кольцевые подписи и оффчейн-протоколы. Отмечено, что эти методы не достигают достаточного уровня децентрализации, что является важным аспектом для систем распределенного реестра. Для одновременного обеспечения свойств децентрализации и конфиденциальности информации используются методы доказательства с нулевым разглашением, включая методы компактных неинтерактивных доказательств знания (SNARK). В статье приводится математическая модель систем доказательства SNARK, а также описаны подходы к их программной реализации. Приведены результаты экспериментов, направленные на сравнение производительности методов SNARK для решения прикладной задачи проведения операций с цифровыми финансовыми активами. Результаты эксперимента позволяют выделить дальнейшие возможности снижения времени генерации доказательства и сокращения его объема посредством использования пакетной верификации. Полученные результаты имеют практическую значимость для разработки систем распределенного реестра, требующих высокого уровня конфиденциальности и децентрализации.
Ключевые слова: системы распределенного реестра, блокчейн, конфиденциальность, доказательство с нулевым разглашением.
Для цитирования: ЗАПЕЧНИКОВ, Сергей В.; КОНКИН, Анатолий Ю. ОБЕСПЕЧЕНИЕ КОНФИДЕНЦИАЛЬНОСТИ ИНФОРМАЦИИ В СИСТЕМАХ РАСПРЕДЕЛЕННОГО РЕЕСТРА ПОСРЕДСТВОМ ДОКАЗАТЕЛЬСТВ С НУЛЕВЫМ РАЗГЛАШЕНИЕМ. Безопасность информационных технологий, [S.l.], т. 31, № 1, с. 75–85, 2024. ISSN 2074-7136. URL: https://bit.spels.ru/index.php/bit/article/view/1612. DOI: http://dx.doi.org/10.26583/bit.2024.1.04.

Sergey V. Zapechnikov[1], Anatoly Yu. Konkin[2]
National Research Nuclear University MEPhI (Moscow Engineering Physics Institute),
Kashirskoe sh., 31, Moscow, 115409, Russia
[1]e-mail: svzapechnikov@mephi.ru, https://orcid.org/0000-0002-7975-6040
[2]e-mail: aykonkin@gmail.com, https://orcid.org/0000-0002-1300-6466

Ensuring the privacy of information in distributed ledger systems with zero-knowledge proofs

Abstract. This article addresses the privacy issue in distributed ledger technology (DLT) and blockchain. The research focuses on the applied use case of private transactions involving digital assets. The article provides a comparison of various privacy methods, including mixing networks, ring signatures, and off-chain protocols. It is noted that these methods do not achieve the target level of decentralization, which is a crucial property for blockchain. To address these limitations, the article introduces zero-knowledge proof methods, with a particular emphasis on SNARK. Mathematical descriptions of SNARK methods are

presented, along with software implementation approaches. The research includes experiments conducted to compare the performance and information security properties of SNARK methods in the use case of private transactions involving digital assets. The experimental results highlight approaches to improve performance using batch verification methods. The findings of this research have practical implications for the development of systems requiring high levels of privacy and decentralization in blockchain-based systems.

Keywords: distributed ledger, blockchain, confidentiality, zero knowledge proofs.
For citation: ZAPECHNIKOV, Sergey V.; KONKIN, Anatoly Yu. Ensuring the privacy of information in distributed ledger systems with zero-knowledge proofs. IT Security (Russia), [S.l.], v. 31, no. 1, p. 75–85, 2024. ISSN 2074-7136. URL: https://bit.spels.ru/index.php/bit/article/view/1612. DOI: http://dx.doi.org/10.26583/bit.2024.1.04.

Введение

Технология распределенного реестра реализует такой способ хранения и передачи информации, который отличается отсутствием централизованного управления, возможностью автоматизации передачи сообщений между участниками, поддержкой свойства неизменяемости данных. Системы распределенного реестра широко применяются в разных сферах деловой деятельности, включая финансовую отрасль и услуги страхования [1]. В настоящее время проекты на основе технологии распределенного реестра активно запускаются в промышленную эксплуатацию, а на российском и зарубежном рынках присутствуют десятки зрелых решений.

Системы распределенного реестра характеризуются множественной репликацией данных на узлах участников. Такой подход обеспечивает децентрализацию системы и высокий уровень отказоустойчивости, исключает риски подмены данных, а также обеспечивает неизменность и сохранность информации. В то же время репликация может приводить к снижению уровня конфиденциальности, что особенно значимо в финансовом секторе и других областях, где требуется высокий уровень безопасности информации.

В статье анализируются известные методы обеспечения конфиденциальности информации в системах распределенного реестра: перемешивающие сети, кольцевые подписи, оффчейн-протоколы и методы доказательства с нулевым разглашением. Отмечается, что методы доказательства с нулевым разглашением наилучшим образом подходят для применения в прикладных задачах финансовой отрасли. Особое значение для систем распределенного реестра приобрели методы компактных неинтерактивных доказательств знания (SNARK – Succinct Non-interactive ARguments of Knowledge). В связи с этим далее проводится более подробный анализ методов SNARK, а также исследуются возможности повышения производительности этих методов. Представлены результаты программных экспериментов с использованием систем доказательства SNARK.

Статья организована следующим образом. В п. 1 представлен обзор методов обеспечения конфиденциальности в системах распределенного реестра, включая их сравнительный анализ. В п. 2 систематизированы методы доказательства с нулевым разглашением, а п. 3 посвящен математическим моделям систем доказательства SNARK. Далее, в п. 4 приводятся результаты эксперимента по применению систем доказательства SNARK в прикладной задаче, связанной с выполнением операций с цифровыми финансовыми активами (ЦФА). В заключение рассмотрены возможности дальнейших исследований для повышения производительности систем доказательства SNARK.

1. Методы обеспечения конфиденциальности транзакций в системах распределенного реестра

Для систем распределенного реестра можно выделить несколько типов задач обеспечения конфиденциальности информации, связанных с размещением в реестре данных о транзакциях.

Конфиденциальность адресов отправителя и получателя транзакций. Первая задача заключается в обеспечении невозможности анализа соответствия между пользователями и отправляемыми ими транзакциями. Наиболее простые подходы к достижению конфиденциальности реализуются с использованием обезличивания и случайных (псевдослучайных) адресов. Однако такие подходы оставляют риски извлечения информации путем использования специальных средств анализа на стороне других участников системы распределенного реестра.

Другим способом обеспечения конфиденциальности адресов отправителя и получателя транзакций является применение перемешивающих сетей – «миксеров». Протоколы, основанные на данном методе, реализуются с помощью доверенной стороны, которая «перемешивает» транзакции с целью сокрытия информации [2]. Базовая перемешивающая сеть, также известная как миксер, действует в качестве маршрутизатора, принимая сообщения от нескольких отправителей, перемешивая их в случайном порядке и отправляя получателям.

Кольцевые подписи представляют собой протокол последовательной цифровой подписи и отправки транзакции некоторой группой участников системы таким образом, что получатель имеет возможность проверить целостность сообщения, но не имеет возможности определить, кто именно из участников группы был инициатором транзакции [3]. Кольцевые подписи позволяют отказаться от доверенной третьей стороны, однако они не обеспечивают высокого уровня конфиденциальности, поскольку инициатор является одним из пользователей группы. Кроме этого, использование кольцевых подписей подразумевает дополнительные вычислительные издержки из-за необходимости генерации цифровой подписи множеством пользователей.

Конфиденциальность содержания транзакций. Вторая задача заключается в том, чтобы сведения о транзакциях были доступны только авторизованным участникам системы распределенного реестра (как правило, только отправителям и получателям каждой из транзакций).

Наиболее распространённый подход к решению такой задачи – это оффчейн-протоколы. Они основаны на использовании дополнительного сервиса, который обрабатывает транзакции и записывает их результаты, например, контрольные суммы (хэш-суммы) сообщений, в распределённый реестр. В то же время конфиденциальные данные передаются напрямую между отправителем и получателем, минуя распределенный реестр. Недостаток такого подхода – увеличение риска потери информации из-за наличия дополнительного хранилища вне распределённого реестра, который становится единственным местом хранения этой информации.

Другим способом решения этой задачи является применение доказательств с нулевым разглашением (ZKP – zero knowledge proofs), которые позволяют обеспечить конфиденциальность содержания транзакций без доверенной третьей стороны или создания дополнительных сервисов вне распределенного реестра. Эти протоколы представляют наибольший интерес и поэтому подробнее рассматриваются в статье далее.

Конфиденциальность вычислений. Обеспечение конфиденциальности вычислений представляет собой отдельную задачу, которая заключается в невозможности воспроизведения или исполнения исходного кода смарт-контрактов в системе

распределенного реестра. Далее этот класс задач не рассматривается, поскольку он наименее распространен, характеризуется весьма сложными и специфичными решениями.

В табл. 1 показано сравнение методов обеспечения конфиденциальности транзакций в системах распределенного реестра.

Таблица 1. Сравнение методов обеспечения конфиденциальности в системах распределенного реестра

Тип	Группа методов	Преимущества	Недостатки
Конфиденциальность адресов отправителя и получателя транзакций	Одноразовые адреса	Наиболее простые способы реализации, не требуют дополнительных разработок.	Риски раскрытия информации при связывании адресов источника и получателя средств.
	Перемешивающие сети	Достаточно простые способы реализации, совместимость с разными платформами.	Наличие (доверенной) третьей стороны для выполнения функций «смешивания».
	Кольцевые подписи	Высокий уровень конфиденциальности.	Увеличение объема транзакции за счет наличия набора публичных ключей группы подписывающих.
Конфиденциальность содержания транзакций	Оффчейн-протоколы	Достаточно простые способы реализации и доказательства свойств безопасности.	Необходимость поддержки дополнительных сервисов помимо распределенного реестра.
	Методы доказательства с нулевым разглашением	Высокий уровень безопасности информации.	Малое число реализаций в промышленной эксплуатации, сложность строгого (формального) доказательства свойств безопасности.

2. Доказательства с нулевым разглашением

Любая система доказательства включает в себя три обязательных элемента: доказывающего, проверяющего и доказываемое утверждение. На практике доказательство представляют собой некоторый текст, передаваемый от доказывающего к проверяющему, который служит для того, чтобы убедить проверяющего в истинности утверждения, заранее сформулированного и известного обоим участникам. Такие доказательства являются неинтерактивными, т.е. не требуют взаимодействия доказывающего и проверяющего в процессе проверки доказательства. Текст доказательства содержит определенные аргументы в пользу доказываемого утверждения, которые проверяющий прочитывает, понимает и подтверждает истинность исходного утверждения (или, наоборот, опровергает его). Такие утверждения могут быть сгенерированы для любых задач класса сложности NP, т.е. таких, которые решаются не обязательно за полиномиальное время, но чьи решения могут быть проверены за полиномиальное время.

Однако для того, чтобы сгенерировать доказательство решения задач, выходящих за рамки класса NP, необходима другая модель доказательства. Такая модель получила название вероятностного интерактивного доказательства [4]. Вероятностным интерактивным доказательством называется такая система доказательства, где доказывающий не ограничен в своих вычислительных ресурсах, проверяющий имеет полиномиально ограниченные вычислительные ресурсы, а оба участника могут выполнять вероятностные алгоритмы и взаимодействовать при проверке доказательства, т.е.

выполнять протокол интерактивного доказательства. Любой интерактивный протокол доказательства стандартным образом, а именно, с помощью преобразования Фиата-Шамира, может быть преобразован в неинтерактивную систему доказательства.

Любые доказательства должны обладать, как минимум, следующими двумя свойствами:

- *полнота* (completeness): если утверждение истинно, и стороны следуют протоколу, то утверждение будет принято проверяющей стороной.
- *состоятельность* (soundness): если утверждение ложно, то никакой доказывающий не сможет убедить проверяющего в истинности такого утверждения (за исключением ошибок, допускаемых с пренебрежимо малой вероятностью).

Наличие этих двух свойств с высокой вероятностью гарантирует защиту проверяющего от нечестного доказывающего, однако не гарантирует защиту доказывающего от попыток нечестного проверяющего получить какую-либо дополнительную информацию о том, почему доказываемое утверждение истинно, которую он не хотел бы разглашать.

Доказательства с нулевым разглашением – криптографические протоколы, позволяющие одному участнику доказать некоторое утверждение другому участнику без раскрытия конфиденциальных данных или информации о способе его подтверждения [4]. Такие доказательства должны обладать следующим дополнительным свойством:

- *нулевое разглашение*: при истинности утверждения и соблюдении протокола, проверяющая сторона не получит никакой конфиденциальной информации, кроме самого факта истинности утверждения.

При доказательстве сложных утверждений существенное значение приобретает такое свойство доказательств как *компактность* (succinctness), т.е. достаточно малый размер набора данных, представляющих доказательство, и малое время его проверки.

Системы доказательства, обладающие всем набором вышеперечисленных свойств, получили в литературе название SNARK и являются самыми распространёнными. В дальнейшем рассматриваются именно такие системы доказательства.

3. Математическое описание методов SNARK

Доказательства типа SNARK могут использоваться в случае, когда доказываемое утверждение выражается в виде арифметической схемы C. Функция C является арифметической схемой тогда и только тогда, когда все её выходные переменные могут быть определены через входные переменные путем переходов по направленному ациклическому графу, вершины которого соответствуют арифметическим операциям сложения и умножения в некотором выбранном конечном поле, а рёбра – передаче данных между операциями. Из теории алгоритмов известно, что любую вычислимую по Тьюрингу функцию можно представить в виде арифметической схемы [5].

Пусть имеются доказывающий P и проверяющий V. Доказывающий P имеет открытую часть доказываемого утверждения x и его конфиденциальную часть w (так называемое свидетельство). Система доказательства SNARK состоит из следующих алгоритмов [6]:

1) алгоритма генерации открытых параметров $S(C) = S(S_p, S_v)$, который создаёт публичные параметры, используемые для генерации и проверки доказательства;

2) алгоритма генерации доказательства $P(S_p, x, w) \to \pi$, который формирует доказательство, передаваемое от доказывающего P к проверяющему V;

3) алгоритма проверки доказательства $V(S_v\, x, \pi) \to accept\ /\ reject$, который подтверждает достоверность представленного доказательства либо отклоняет его.

Схема функционирования системы доказательства типа SNARK показана на рис. 1.

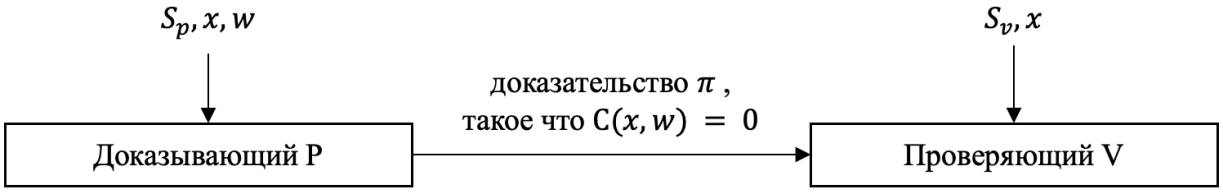

Рис. 1. Система доказательства SNARK

Известны две основные конструкции практичных систем доказательства SNARK.

Первая основана на линейных вероятностно проверяемых доказательствах (PCP – Probabilistically Checkable Proof). Известная теорема о PCP [4] определяет возможность представления доказательств в компактной форме: проверяющий, обладая доступом только к ограниченному количеству битов доказательства π, способен с высокой вероятностью проверить истинность доказываемого утверждения. В современной литературе такая конструкция представлена в реализации Groth16 [7].

Основное достоинство системы доказательства Groth16 состоит в том, что оно позволяет получать самые компактные доказательства: независимо от размера исходных данных и сложности вычисляемой функции длина доказательства всегда равна размеру представления трёх элементов алгебраической группы, выбранной при генерации доказательства, – на практике это обычно несколько сотен байтов.

Недостаток системы Groth16 и её модификаций заключается в необходимости выполнения алгоритма генерации открытых параметров доверенным лицом и повторении этой процедуры для каждой новой арифметической схемы. Другими словами, система Groth16 создается для определенных операций и не способна без перенастройки генерировать доказательства для различных арифметических схем.

Вторая конструкция SNARK – это комбинация двух компонентов: схемы коммитмента функций и интерактивного доказательства с оракулом (IOP - interactive oracle proof).

Схема коммитмента может быть представлена в виде пары алгоритмов $Commit$ и $Verify$, где:

- $c \leftarrow Commit\ (x, r)$ – алгоритм генерации коммитмента, где x – сообщение, r – случайная величина, c – вырабатываемое значение «свёртки» сообщения x, также называемой коммитментом;
- $Verify\ (c, x, r) = \{accept,\ reject\}$ – алгоритм проверки коммитмента c, где x – сообщение, r – случайная величина, результатом которого является либо подтверждение достоверности коммитмента, либо его отклонение и отказ от дальнейшего взаимодействия с создателем коммитмента.

Функции $Commit$ и $Verify$ должны обладать следующими свойствами [8]:

- *сокрытием* (hiding): для любого фиксированного значения x и при случайном выборе r, вычисленное значение c не раскрывает информации об x, благодаря случайности r и однонаправленности хеш-функции.
- *связыванием* (binding): для любого коммитмента c вычислительно сложно подобрать два различных значения x_1 и x_2 (и соответствующих им r_1и r_2), так чтобы их коммитменты совпали $Commit\ (x_1, r_1) = Commit\ (x_2, r_2)$.

Для использования коммитментов в системах доказательства SNARK требуется определить схему коммитмента для функций. Пусть имеется семейство

функций $F = \{f: X \to Y\}$. Тогда доказывающий P направляет проверяющему V коммитмент некоторой функции f из этого семейства с использованием случайной величины r. Проверяющий V высылает некоторое значение $x \in X$, после чего доказывающий P направляет значение $y \in Y$ и доказательство π, такое что $f(x) = y$. Система доказательства SNARK с использованием коммитментов функций показана на рис. 2.

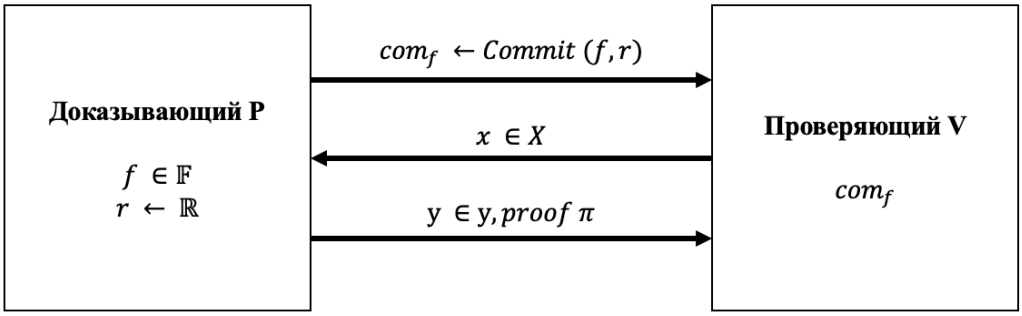

Рис. 2. Система доказательства SNARK с использованием коммитментов

Схема коммитмента функций может быть представлена в виде следующего набора алгоритмов:

- алгоритма генерации общих параметров, доступных для доказывающего и проверяющего $Setup() \to gp$;
- алгоритма коммитмента функции $f \in F$ с некоторой случайной величиной r $Commit\ (gp, f, r) \to com_f$.
- алгоритма генерации доказательства $P(gp, f, x, y) \to \pi$, который принимает на вход функцию *f*, некоторые значения $x \in X$ и $y \in Y$, а далее выдаёт доказательство π того, что одновременно выполняются условия $f(x) = y$ и $Commit\ (gp, f, r) = com_f$.
- алгоритма $V(gp, com_f, x, y, \pi) \to \{accept, reject\}$, который проверяет выполнение условий предыдущего алгоритма без разглашения функции f.

Существуют разные схемы коммитмента функций. Так, полиномиальные коммитменты позволяют фиксировать полином одной переменной $f(x)$ выбранный из семейства полиномов $F(X)$, а затем доказывать некоторые свойства этого полинома без раскрытия всех его коэффициентов. Мультилинейные коммитменты позволяют фиксировать функцию многих переменных $f(x_1, \dots, x_n)$, выбранную из семейства функций $F(X_1, \dots X_n)$. Также можно выделить векторные коммитменты, которые позволяют фиксировать вектор $\vec{u} = (u_1, \dots, u_d)$ так, чтобы при открытии коммитмента можно было доказать, что определенное значение u_i находится на конкретной позиции *i* в векторе $\vec{u}$, т.е. $f_{\vec{u}}(i) = u_i$, при этом не раскрывая других значений вектора [4].

В табл. 2 приводится сравнение схем коммитмента по разным критериям.

Полиномиальные коммитменты особенно удобно применять в системах распределенного реестра для выполнения операций с ЦФА из-за их способности обеспечивать высокий уровень защиты данных, сочетаясь при этом с простотой реализации и использования. Примером таких полиномиальных коммитментов служат схемы KZG [8].

Вторая составляющая конструкции SNARK – это интерактивное доказательство с оракулом (IOP), в котором доказывающий P и проверяющий V обмениваются последовательностью сообщений. В первом раунде доказывающий P предоставляет проверяющему V так называемого оракула, к которому проверяющий впоследствии может обращаться для выборочного запроса определенных битов доказательства. Оракул – это

абстрактный компонент («чёрный ящик»), к которому имеют доступ и доказывающий, и проверяющий для сэмплирования некоторой случайной величины (как правило, из равномерного распределения). Для описания IOP используется понятие системы ограничений первого ранга R1CS (Rank-1 Constraint Systems) [3]. Это система линейных уравнений, предназначенная для кодирования и представления арифметических схем. Система R1CS определяется набором матриц A, B и C размерностью $m \times n$, при этом для каждого $i = \overline{1,m}$ определяется уравнение вида: $a_i \times b_i = c_i$. R1CS используется в IOP для того, чтобы преобразовать задачу доказательства правильности вычисления некоторой арифметической схемы в задачу доказательства существования решения соответствующей системы линейных уравнений без раскрытия самого решения.

Таблица 2. Сравнение схем коммитмента

Схема	Описание	Преимущества	Недостатки
Полиномиальные коммитменты	Коммитменты, использующиеся для защиты дискретных логарифмов и других криптографических задач.	Высокая степень защиты данных, простота реализации и использования.	Ограничение на размер многочлена. Удобно в блокчейн, т.к. данные в транзакции одномерны.
Мультилинейные коммитменты	Коммитменты, использующиеся для защиты мультилинейных отображений и других задач, связанных с многомерными данными.	Высокая степень защиты данных, эффективность работы с многомерными данными.	Ограничения на количество переменных.
Векторные коммитменты	Коммитменты, использующиеся для защиты векторных данных и других задач, связанных с множествами данных.	Высокая степень защиты данных, простота работы с векторными данными.	Ограничение на размер вектора.

Для преобразования IOP в неинтерактивное доказательство можно использовать преобразование Фиата-Шамира [4]. Вместо ожидания запросов проверяющего доказывающий самостоятельно вычисляет случайные запросы с помощью криптографической хеш-функции. Таким образом, все запросы и ответы могут быть вычислены последовательно доказывающим без взаимодействия с проверяющим.

В табл. 3 приводится характеристика основных протоколов, использующих комбинацию схемы коммитмента с IOP.

Таблица 3. Сравнение протоколов IOP

Протокол	Описание
FRI (Fast Reed-Solomon Interactive Oracle Proofs)	Предназначен для эффективной проверки принадлежности к кодам Рида-Соломона. Служит в качестве основного блока для многих последующих протоколов [9].
Ligero	Один из первых протоколов IOP с уменьшенными размерами доказательства. Использует FRI как один из своих компонентов [10].
Dory	Использует рекурсивные вызовы для уменьшения размера доказательства. Также опирается на FRI [11].
Plonk	Введен для упрощения универсальных схем доказательства. Имеет универсальный набор доверенных параметров для всех арифметических схем одинакового размера. [12]

4. Эксперимент для прикладной задачи обеспечения конфиденциальности при операциях с ЦФА

Системы доказательства SNARK могут быть использованы для обеспечения конфиденциальности при операциях по выпуску и передаче ЦФА, соответствующих положениям Федерального закона № 259-ФЗ «О цифровых финансовых активах».

На практике корпоративные системы распределенного реестра реализуют процесс аутентификации пользователей путем проверки цифровых подписей для транзакций, хранящихся в распределенном реестре. Однако этот подход имеет некоторые риски информационной безопасности из-за разглашения данных о содержании транзакции.

Процесс выпуска ЦФА с применением SNARK может выглядеть следующим образом.

1. После регистрации в информационной системе Продавец (доказывающий P) получает идентификаторы ЦФА от Покупателя (проверяющий V).
2. Доказывающий P генерирует секретную соль – набор данных для свидетельства w, а также код аутентификации на основе идентификатора ЦФА – набор данных для сообщения x.
3. Информационная система регистрирует идентификатор ЦФА в смарт-контракте.
4. Доказывающий P создает доказательство знания секретной соли – набор данных π.
5. При аутентификации доказывающий P использует секретную соль, которая проверяется смарт-контрактом с помощью функции $Prove(w, x)$.
6. При выполнении условия $Verify(x, \pi) = accept$ смарт-контракт записывает информацию по передаче ЦФА от доказывающего P к проверяющему V.

В целях реализации систем доказательства SNARK могут быть использованы специальные языки описания предметной области (DSL - Domain specific language). Такие языки позволяют описать систему ограничений R1CS, выполнить её дальнейшее преобразование в байт-код для виртуальных машин EVM (Ethereum virtual machine), сгенерировать доказательство.

В экспериментальной части работы реализация SNARK осуществлялась с использованием набора библиотек и языка ZoKrates, совместимого с Ethereum [13]. ZoKrates имеет возможности использования разных систем доказательства, набор библиотек ZoKrates поддерживает широкий спектр криптографических примитивов и операций, таких как арифметические операции, реализаций условий, формирование циклов, вычисление хэш-кодов и пр.

В рамках эксперимента реализован сценарий проведения операций с ЦФА. В эксперименте использовались различные хэш-функции: SHA-256 и функция Педерсена [14]. При этом получены и измерены следующие параметры протокола: скорость проверки доказательства, размер транзакций и объём используемой оперативной памяти. Результаты эксперимента представлены в табл. 4.

Таблица 4. Результаты эксперимента с использованием ZoKrates

Показатель	SHA-256	Pedersen
Скорость проверки доказательства	65 ± 15 с	10 ± 5 с
Размер транзакций	44 МБ	5 МБ
Объём оперативной памяти	1100 МБ	20 МБ

В результате проведенного эксперимента выявлена необходимость ускорения проверки доказательств. Способы решения этой задачи неочевидны. В частности, в некоторых источниках предлагается использовать пакетную верификацию для массовой проверки доказательств, сгенерированных для множества однородных утверждений [15, 16]. Такой подход потенциально может быть применим в приложениях, связанных с

обработкой ЦФА, где операции, как правило, обладают аддитивными свойствами. Однако для реализации такого подхода требуются дальнейшие исследования.

Заключение

В статье проанализированы методы обеспечения конфиденциальности в системах распределенного реестра. Проведен анализ различных аспектов конфиденциальности, таких как конфиденциальность адресов отправителя и получателя транзакций, содержания транзакций и вычисляемых функций. Особо выделены системы доказательства SNARK для применения в системах распределенного реестра.

Выявлено, что системы доказательства SNARK содержат такие обязательные компоненты как схемы коммитмента функций и интерактивные доказательства с оракулом (IOP). Показано, что различные IOP развиваются либо с целью повышения скорости генерации и проверки доказательства, а также снижения объема доказательства и требуемых при этом вычислительных ресурсов.

В экспериментальной части исследования реализован сценарий аутентификации пользователей с использованием доказательств SNARK при операциях с ЦФА. Полученные результаты указывают на необходимость повышения скорости выполнения операций и снижения размера доказательства. Предполагается, что в операциях с ЦФА может быть применена пакетная верификация. В дальнейшем планируется исследовать свойства безопасности методов пакетной верификации для систем доказательства SNARK.

СПИСОК ЛИТЕРАТУРЫ:

1. Равал С. Децентрализованные приложения. Технология Blockchain в действии. СПб: Питер, 2017. – 240 с.
2. Giannikou I. Cryptocurrencies and Mixing Services. 2021. URL: https://pergamos.lib.uoa.gr/uoa/dl/object/2948074/file.pdf (дата обращения: 16.01.2024).
3. Konkin A., Zapechnikov S. Zero knowledge proof and SNARK for private blockchains. Journal of Computer Virology and Hacking Techniques. V. 19, p. 443–449, 2023. DOI: 10.1007/s11416-023-00466-1.
4. Thaler J. Proofs, Arguments, and Zero-Knowledge, Foundations and Trends in Privacy and Security. Foundations and Trends in Privacy and Security. V. 4, no. 2–4, p. 117–660, 2023. DOI: 10.1561/3300000030.
5. Запечников С.В. Доказательства с нулевым разглашением и их применения при обработке информации в недоверенных средах. Вестник современных цифровых. 2021, № 6, с.11–22. – EDN: OHFIYI.
6. Yang X., Wenjie L. A zero-knowledge-proof-based digital identity management scheme in blockchain, Computers & Security. V. 99, 102050 p., 2020. DOI: 10.1016/j.cose.2020.102050.
7. Groth J. (2016). On the Size of Pairing-Based Non-interactive Arguments. In: Fischlin, M., Coron, JS. (eds) Advances in Cryptology – EUROCRYPT 2016. EUROCRYPT 2016. Lecture Notes in Computer Science, v. 9666. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-662-49896-5_11.
8. Kate A., Zaverucha G.M., Goldberg I. (2010). Constant-Size Commitments to Polynomials and Their Applications. In: Abe, M. (eds) Advances in Cryptology - ASIACRYPT 2010. ASIACRYPT 2010. Lecture Notes in Computer Science, v. 6477. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-17373-8_11.
9. Ben-Sasson E., Bentov I., Horesh Y., Riabzev M. STARKs: Scalable Transparent ARguments of Knowledge. URL: https://eprint.iacr.org/2018/046 (дата обращения: 16.01.2024).
10. Ames S., Hazay C., Ishai Y., Venkitasubramaniam, M. Ligero: Lightweight Sublinear Arguments Without a Trusted Setup. Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. 2018. DOI: 10.1145/3133956.3134104.
11. Lee J. Dory: Efficient, Transparent arguments for Generalised Inner Products and Polynomial Commitments. Microsoft Research. 2020. URL: https://eprint.iacr.org/2020/1274.pdf (дата обращения: 16.01.2024).
12. Gabizon A., Williamson Z., Ciobotaru O. PLONK: Permutations over Lagrange-bases for Oecumenical Noninteractive arguments of Knowledge. IACR Cryptology. 2019. URL: https://eprint.iacr.org/2019/953 (дата обращения: 16.01.2024).
13. Eberhardt J. and Tai S. ZoKrates - Scalable Privacy-Preserving Off-Chain Computations. IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada. 2018, p. 1084–1091. DOI: 10.1109/Cybermatics_2018.2018.00199.

14. Pedersen T. Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing. Advances in Cryptology – CRYPTO 91. LNCS 576. p. 129–140. Springer, Berlin, Heidelberg. 2019. DOI:10.1007/3-540-46766-1_9.
15. Guan Z., Wan Z., Yang Y., Zhou Y. and Huang B. BlockMaze: An Efficient Privacy-Preserving Account-Model Blockchain Based on zk-SNARKs. IEEE Transactions on Dependable and Secure Computing, v. 19, no. 3, p. 1446–1463, 1 May-June 2022. DOI: 10.1109/TDSC.2020.3025129.
16. Libert B. Vector Commitments With Proofs of Smallness: Short Range Proofs and More. URL: https://eprint.iacr.org/2023/800.pdf (дата обращения: 16.01.2024).

REFERENCES:

[1] Raval S. Decentralized applications: Harnessing Bitcoin’s blockchain technology. O’Reilly, 2016. – 240 p. (in Russian).
[2] Giannikou I. Cryptocurrencies and Mixing Services. 2021. URL: https://pergamos.lib.uoa.gr/uoa/dl/object/2948074/file.pdf (accessed: 16.01.2024).
[3] Konkin A., Zapechnikov S. Zero knowledge proof and SNARK for private blockchains. Journal of Computer Virology and Hacking Techniques. V. 19, p. 443–449, 2023. DOI: 10.1007/s11416-023-00466-1.
[4] Thaler J. Proofs, Arguments, and Zero-Knowledge, Foundations and Trends in Privacy and Security. Foundations and Trends in Privacy and Security. V. 4, no. 2–4, p 117–660, 2023. DOI: 10.1561/3300000030.
[5] Zapechnikov S.V. Zero-knowledge proofs and their applications for information processing into untrusted environments. Vestnik sovremennyh tsifrovyh tehnologiy. 2021, no. 6, p. 11–22 (in Russian) – EDN: OHFIYI.
[6] Yang X., Wenjie L. A zero-knowledge-proof-based digital identity management scheme in blockchain, Computers & Security. V. 99, 102050 p., 2020. DOI: 10.1016/j.cose.2020.102050.
[7] Groth J. (2016). On the Size of Pairing-Based Non-interactive Arguments. In: Fischlin, M., Coron, JS. (eds) Advances in Cryptology – EUROCRYPT 2016. EUROCRYPT 2016. Lecture Notes in Computer Science, v. 9666. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-662-49896-5_11.
[8] Kate A., Zaverucha G.M., Goldberg I. (2010). Constant-Size Commitments to Polynomials and Their Applications. In: Abe, M. (eds) Advances in Cryptology - ASIACRYPT 2010. ASIACRYPT 2010. Lecture Notes in Computer Science, v. 6477. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-17373-8_11.
[9] Ben-Sasson E., Bentov, I., Horesh, Y., Riabzev, M. STARKs: Scalable Transparent ARguments of Knowledge. URL: https://eprint.iacr.org/2018/046 (accessed: 16.01.2024)
[10] Ames, S., Hazay, C., Ishai, Y., Venkitasubramaniam, M. Ligero: Lightweight Sublinear Arguments Without a Trusted Setup. Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. 2018. DOI: 10.1145/3133956.3134104.
[11] Lee J. Dory: Efficient, Transparent arguments for Generalised Inner Products and Polynomial Commitments. Microsoft Research. 2020. URL: https://eprint.iacr.org/2020/1274.pdf (accessed: 16.01.2024).
[12] Gabizon A., Williamson Z., Ciobotaru O. PLONK: Permutations over Lagrange-bases for Oecumenical Noninteractive arguments of Knowledge. IACR Cryptology. 2019. URL: https://eprint.iacr.org/2019/953 (accessed: 16.01.2024).
[13] Eberhardt J. and Tai S. ZoKrates - Scalable Privacy-Preserving Off-Chain Computations. IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada. 2018, p. 1084–1091. DOI: 10.1109/Cybermatics_2018.2018.00199.
[14] Pedersen T. Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing. Advances in Cryptology – CRYPTO 91. LNCS 576. p. 129–140. Springer, Berlin, Heidelberg. 2019. DOI:10.1007/3-540-46766-1_9.
[15] Guan Z., Wan Z., Yang Y., Zhou Y. and Huang B. BlockMaze: An Efficient Privacy-Preserving Account-Model Blockchain Based on zk-SNARKs. IEEE Transactions on Dependable and Secure Computing, v. 19, no. 3, p. 1446–1463, 1 May-June 2022. DOI: 10.1109/TDSC.2020.3025129.
[16] Libert B. Vector Commitments with Proofs of Smallness: Short Range Proofs and More. URL: https://eprint.iacr.org/2023/800.pdf (accessed: 16.01.2024).

Поступила в редакцию – 16 января 2024 г. Окончательный вариант – 12 февраля 2024 г.
Received – January 16, 2024. The final version – February 12, 2024.