РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 719 311**(13) **C1**

(51) МПК
G06F 21/00 (2013.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК
G06F 21/10 (2020.02); G06F 21/554 (2020.02); H04L 9/3218 (2020.02)

(21)(22) Заявка: **2019111923, 27.11.2018**

(24) Дата начала отсчета срока действия патента:
27.11.2018

Дата регистрации:
17.04.2020

Приоритет(ы):
(22) Дата подачи заявки: **27.11.2018**

(45) Опубликовано: **17.04.2020** Бюл. № 11

(85) Дата начала рассмотрения заявки PCT на национальной фазе: **19.04.2019**

(86) Заявка PCT:
CN 2018/117548 (27.11.2018)

(87) Публикация заявки PCT:
WO 2019/072275 (18.04.2019)

Адрес для переписки:
129090, Москва, ул. Большая Спасская, д. 25, строение 3, ООО "Юридическая фирма Городисский и Партнеры"

(72) Автор(ы):
МА, Баоли (CN),
ЧЖАН, Вэньбинь (CN),
ЛИ, Личунь (CN),
ЛЮ, Чжэн (CN),
ИНЬ, Шань (CN)

(73) Патентообладатель(и):
Алибаба Груп Холдинг Лимитед (CN)

(56) Список документов, цитированных в отчете о поиске: **CN 108683669 A, 19.10.2018. WO 2018/185724 A1, 11.10.2018. CN 107451175 A, 08.12.2017. CN 108282459 A, 13.07.2018. RU 2636105 C1, 20.11.2017.**

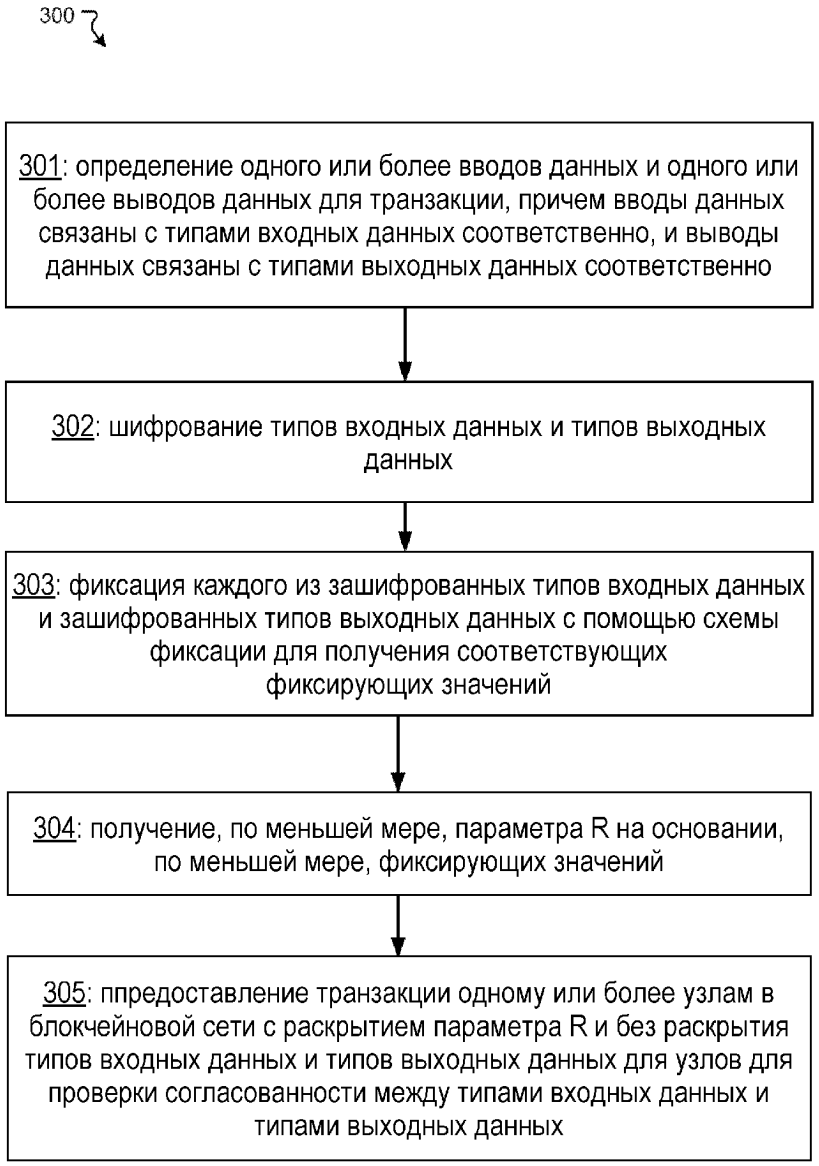
(54) СИСТЕМА И СПОСОБ ЗАЩИТЫ ИНФОРМАЦИИ

(57) Реферат:

Изобретение относится к устройствам для защиты информации. Технический результат изобретения заключается в защите онлайновой информации. Компьютерно-реализуемый способ защиты информации содержит: определение одного или более вводов данных и одного или более выводов данных для транзакции, причем вводы данных связаны с типами входных данных соответственно, и выводы данных связаны с типами выходных данных соответственно; шифрование типов входных данных и типов выходных данных; заверение каждого из зашифрованных типов входных данных и зашифрованных типов выходных данных с помощью схемы обязательств для получения соответствующих заверяющих значений; получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений; и предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных. 6 н. и 16 з.п. ф-лы, 5 ил.

RU 2719311 C1

RU 2719311 C1



ФИГ.3

RU 2719311 C1

RU 2719311 C1

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(19) **RU** (11) **2 719 311** (13) **C1**

(51) Int. Cl.
G06F 21/00 (2013.01)

(12) ABSTRACT OF INVENTION

(52) CPC
G06F 21/10 (2020.02); *G06F 21/554* (2020.02); *H04L 9/3218* (2020.02)

(21)(22) Application: **2019111923, 27.11.2018**

(24) Effective date for property rights:
27.11.2018

Registration date:
17.04.2020

Priority:
(22) Date of filing: **27.11.2018**

(45) Date of publication: **17.04.2020** Bull. № **11**

(85) Commencement of national phase: **19.04.2019**

(86) PCT application:
CN 2018/117548 (27.11.2018)

(87) PCT publication:
WO 2019/072275 (18.04.2019)

Mail address:
129090, Moskva, ul. Bolshaya Spasskaya, d. 25, stroenie 3, OOO "Yuridicheskaya firma Gorodisskij i Partnery"

(72) Inventor(s):
MA, Baoli (CN),
CHZHAN, Venbin (CN),
LI, Lichun (CN),
LYU, Chzhen (CN),
IN, Shan (CN)

(73) Proprietor(s):
Alibaba Grup Kholding Limited (CN)

RU 2719311 C1

(54) INFORMATION PROTECTION SYSTEM AND METHOD

(57) Abstract:

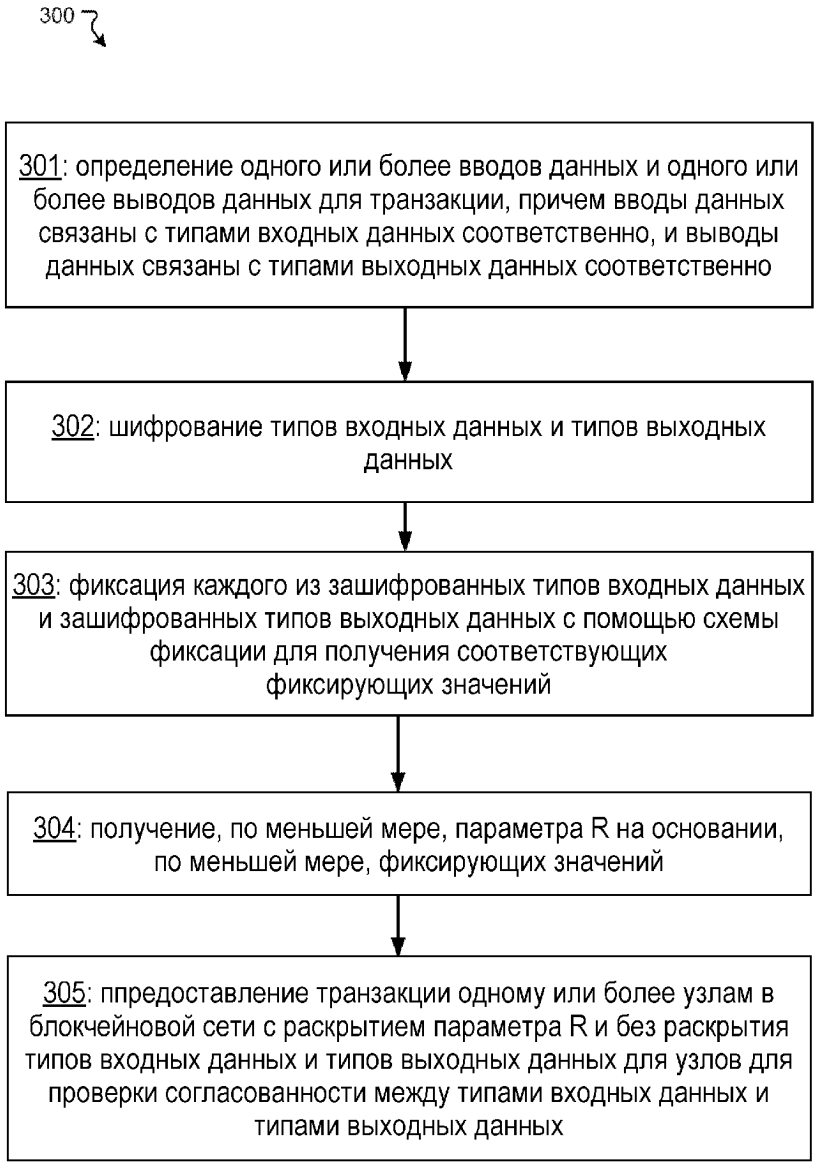
FIELD: information technology.

SUBSTANCE: invention relates to information security devices. Computer-implemented information protection method comprises: determining one or more data inputs and one or more data outputs for a transaction, wherein the data inputs are associated with the input data types, respectively, and the data outputs are associated with the output data types, respectively; encryption of types of input data and types of output data; assurance of each of encoded types of input data and encrypted types of output data by means of a commitment circuit for obtaining corresponding certifying values; obtaining, at least, parameter R on the basis of at least certifying values; and providing a transaction to one or more nodes in a blockchain network with parameter R disclosure and without disclosing types of input data and output data types for nodes to check consistency between types of input data and types of output data.

EFFECT: technical result of the invention is to protect online information.

22 cl, 5 dwg

RU 2719311 C1



ФИГ.3

RU 2719311 C1

RU 2719311 C1

ОБЛАСТЬ ТЕХНИКИ, К КОТОРОЙ ОТНОСИТСЯ ИЗОБРЕТЕНИЕ

[1] Это изобретение относится, в целом, к способам и устройствам для защиты информации.

УРОВЕНЬ ТЕХНИКИ

[2] Персональная информация играет важную роль при осуществлении связи и переносе данных между различными пользователями. В отсутствие защиты пользователи подвергаются опасности хищения личных данных, незаконного перевода или других возможных потерь. Опасность дополнительно увеличивается, когда передачи и переводы реализуются онлайн, вследствие свободного доступа к онлайновой информации.

СУЩНОСТЬ ИЗОБРЕТЕНИЯ

[3] Различные варианты осуществления настоящего изобретения включают в себя системы, способы и долговременные машиночитаемые носители для защиты информации.

[4] Согласно одному аспекту, компьютерно-реализуемый способ защиты информации содержит: определение одного или более вводов данных и одного или более выводов данных для транзакции, причем вводы данных связаны с типами входных данных соответственно, и выводы данных связаны с типами выходных данных соответственно; шифрование типов входных данных и типов выходных данных; заверение каждого из зашифрованных типов входных данных и зашифрованных типов выходных данных с помощью схемы обязательств для получения соответствующих заверяющих значений; получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений; и предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных.

[5] В некоторых вариантах осуществления, шифрование типов входных данных и типов выходных данных содержит шифрование типов входных данных и типов выходных данных с помощью хеш-функции.

[6] В некоторых вариантах осуществления, схема обязательств содержит схему обязательств Педерсена.

[7] В некоторых вариантах осуществления, схема обязательств содержит, по меньшей мере, коэффициент ослепления; и коэффициент ослепления изменяется в зависимости от времени заверения зашифрованных типов входных данных и зашифрованных типов выходных данных.

[8] В некоторых вариантах осуществления, узлам предписывается проверять согласованность между типами входных данных и типами выходных данных без известности типов входных данных и типы выходных данных.

[9] В некоторых вариантах осуществления, транзакция основана, по меньшей мере, на модели неизрасходованных выходов транзакции (UTXO); и вводы данных и выводы данных содержат типы одного или более активов, подвергающихся транзакции.

[10] В некоторых вариантах осуществления, схема обязательств содержит множество коэффициентов ослепления, соответственно соответствующих типам входных данных и типам выходных данных; и получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений содержит: получение разностей между парами заверяющих значений; конкатенацию полученных разностей; шифрование конкатенированных разностей с помощью хеш-функции для получения шифрованного значения x; и получение параметра R на основе, по меньшей мере, шифрованного значения x и разностей между парами коэффициентов ослепления.

[11] В некоторых вариантах осуществления, предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных содержит предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для предписания узлам: получать параметр R и базисную точку G; получать разности между парами заверяющих значений; конкатенировать полученные разности; шифровать конкатенированные разности с помощью хеш-функции для получения шифрованного значения x; получать сумму C многочленов на основе, по меньшей мере, полученных разностей и шифрованного значения x; в ответ на определение, что сумма C равна произведению параметра R и базисной точки G, определять, что типы входных данных и типы выходных данных согласованы; и в ответ на определение, что сумма C не равна произведению параметра R и базисной точки G, определять, что типы входных данных и типы выходных данных не согласованы.

[12] Согласно другому аспекту, на долговременном машиночитаемом носителе данных хранятся инструкции, подлежащие выполнению процессором, для предписания процессору осуществлять операции, содержащие: определение одного или более вводов данных и одного или более выводов данных для транзакции, причем вводы данных связаны с типами входных данных соответственно, и выводы данных связаны с типами выходных данных соответственно; шифрование типов входных данных и типов выходных данных; заверение каждого из зашифрованных типов входных данных и зашифрованных типов выходных данных с помощью схемы обязательств для получения соответствующих заверяющих значений; получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений; и предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных.

[13] Согласно другому аспекту, система для защиты информации содержит процессор и долговременный машиночитаемый носитель данных, подключенный к процессору, причем на носителе данных хранятся инструкции, подлежащие выполнению процессором, для предписания системе осуществлять операции, содержащие: определение одного или более вводов данных и одного или более выводов данных для транзакции, причем вводы данных связаны с типами входных данных соответственно, и выводы данных связаны с типами выходных данных соответственно; шифрование типов входных данных и типов выходных данных; заверение каждого из зашифрованных типов входных данных и зашифрованных типов выходных данных с помощью схемы обязательств для получения соответствующих заверяющих значений; получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений; и предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных.

[14] Согласно другому аспекту, компьютерно-реализуемый способ защиты информации содержит: получение, одним или более узлами в блокчейновой сети, транзакции, инициированной узлом-инициатором. Транзакция связана с одним или более вводами данных и одним или более выводами данных. Вводы данных, соответственно, связаны с типами входных данных, и выводы данных, соответственно, связаны с типами выходных данных соответственно. Типы входных данных и типы

выходных данных шифруются и заверяются согласно схеме обязательств для получения соответствующих заверяющих значений. Типы входных данных и типы выходных данных не раскрываются одному или более узлам. Способ защиты информации дополнительно содержит: проверку, одним или более узлами, согласованности между типами входных данных и типами выходных данных; в ответ на определение, что типы входных данных и типы выходных данных согласованы, добавление, одним или более узлами, транзакции в блокчейновую сеть; и в ответ на определение, что типы входных данных и типы выходных данных не согласованы, отказ, одним или более узлами, от добавления транзакции к блокчейновой сети.

[15] В некоторых вариантах осуществления, проверка согласованности между типами входных данных и типами выходных данных содержит: получение параметра R и базисной точки G; получение разностей между парами заверяющих значений; конкатенацию полученных разностей; шифрование конкатенированных разностей с помощью хеш-функции для получения шифрованного значения x; получение суммы C многочленов на основе, по меньшей мере, полученных разностей и шифрованного значения x; и определение, равна ли сумма C произведению параметра R и базисной точки G.

[16] В некоторых вариантах осуществления, способ дополнительно содержит: в ответ на определение, что сумма C равна произведению параметра R и базисной точки G, определение, что типы входных данных и типы выходных данных согласованы; и в ответ на определение, что сумма C не равна произведению параметра R и базисной точки G, определение, что типы входных данных и типы выходных данных не согласованы.

[17] В некоторых вариантах осуществления, один или более узлов содержат консенсусный узел.

[18] Согласно другому аспекту, на долговременном машиночитаемом носителе данных хранятся инструкции, подлежащие выполнению процессором, для предписания процессору осуществлять операции, содержащие: получение, одним или более узлами в блокчейновой сети, транзакции, инициированной узлом-инициатором. Транзакция связана с одним или более вводами данных и одним или более выводами данных. Вводы данных, соответственно, связаны с типами входных данных, и выводы данных, соответственно, связаны с типами выходных данных соответственно. Типы входных данных и типы выходных данных шифруются и заверяются согласно схеме обязательств для получения соответствующих заверяющих значений. Типы входных данных и типы выходных данных не раскрываются одному или более узлам. Операции дополнительно содержат: проверку, одним или более узлами, согласованности между типами входных данных и типами выходных данных; в ответ на определение, что типы входных данных и типы выходных данных согласованы, добавление, одним или более узлами, транзакции в блокчейновую сеть; и в ответ на определение, что типы входных данных и типы выходных данных не согласованы, отказ, одним или более узлами, от добавления транзакции к блокчейновой сети.

[19] Согласно другому аспекту, система для защиты информации содержит процессор и долговременный машиночитаемый носитель данных, подключенный к процессору, причем на носителе данных хранятся инструкции, подлежащие выполнению процессором, для предписания системе осуществлять операции, содержащие: получение, одним или более узлами в блокчейновой сети, транзакции, инициированной узлом-инициатором. Транзакция связана с одним или более вводами данных и одним или более выводами данных. Вводы данных, соответственно, связаны с типами входных

данных, и выводы данных, соответственно, связаны с типами выходных данных соответственно. Типы входных данных и типы выходных данных шифруются и заверяются согласно схеме обязательств для получения соответствующих заверяющих значений. Типы входных данных и типы выходных данных не раскрываются одному или более узлам. Операции дополнительно содержат: проверку, одним или более узлами, согласованности между типами входных данных и типами выходных данных; в ответ на определение, что типы входных данных и типы выходных данных согласованы, добавление, одним или более узлами, транзакции в блокчейновую сеть; и в ответ на определение, что типы входных данных и типы выходных данных не согласованы, отказ, одним или более узлами, от добавления транзакции к блокчейновой сети.

[20] Эти и другие признаки раскрытых здесь систем, способов и долговременных машиночитаемых носителей, а также способы работы и функции соответствующих элементов конструкции и комбинация частей и экономий производства, явствуют из нижеследующего описания и нижеследующей формулы изобретения, приведенных со ссылкой на прилагаемые чертежи, которые все образуют часть этого описания изобретения, где аналогичные ссылочные позиции обозначают соответствующие части на различных фигурах. Однако очевидно, что чертежи служат только для иллюстрации и описания, но не призваны ограничивать изобретение.

КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

[21] Некоторые признаки различных вариантов осуществления настоящей технологии подробно изложены в нижеследующей формуле изобретения. Чтобы лучше понять признаки и преимущества технологии, следует обратиться к нижеследующему подробному описанию, где изложены иллюстративные варианты осуществления, где используются принципы изобретения, и прилагаемым чертежам, в которых:

[22] фиг. 1 демонстрирует иллюстративную систему для защиты информации, в соответствии с различными вариантами осуществления;

[23] фиг. 2 демонстрирует иллюстративные этапы для инициирования и проверки транзакции, в соответствии с различными вариантами осуществления;

[24] фиг. 3 демонстрирует блок-схему операций иллюстративного способа защиты информации, в соответствии с различными вариантами осуществления;

[25] фиг. 4 демонстрирует блок-схему операций иллюстративного способа защиты информации, в соответствии с различными вариантами осуществления;

[26] фиг. 5 демонстрирует блок-схему иллюстративной компьютерной системы, в которой может быть реализован любой из описанных здесь вариантов осуществления.

ПОДРОБНОЕ ОПИСАНИЕ

[27] Блокчейн можно рассматривать как децентрализованную базу данных, в целом именуемую распределенной бухгалтерской книгой, поскольку операция осуществляется различными узлами (например, вычислительными устройствами) в сети. Любая информация может записываться в блокчейн и сохраняться в нем или считываться из него. Можно создать сервер и подключиться к блокчейновой сети с образованием узла. Любой узел может вносить вклад в вычислительную мощность для поддержания блокчейна путем осуществления сложных вычислений, например, вычисления хеша для добавления блока к текущему блокчейну, и добавленный блок может содержать различные типы данных или информации. Узел, который вносит вклад в вычислительную мощность для добавленного блока, может вознаграждаться жетоном (например, единицей цифровой валюты). Поскольку блокчейн не имеет центрального узла, все узлы одинаковы и в каждом из них содержится вся база данных блокчейна.

[28] Узлы являются, например, вычислительными устройствами или большими

компьютерными системами, которые поддерживают блокчейновую сеть и ее плавное выполнение. Узлы могут использоваться индивидами или группами людей, которые вносят денежный вклад в покупку мощных компьютерных систем, известные как майнинг-фермы. Существует два типа узлов: полные узлы и облегченные узлы. На полных узлах хранится полная копия блокчейна. Полные узлы в блокчейновой сети подтверждают транзакции и блоки, которые они принимают, и ретранслируют их на подключенные равноправные устройства для обеспечения проверки консенсуса транзакций. С другой стороны, в облегченные узлы загружается только часть блокчейна. Например, облегченные узлы используются для транзакций в цифровой валюте. Облегченный узел будет осуществлять связь с полным узлом для осуществления транзакции.

[29] Это свойство децентрализации позволяет препятствовать появлению центра управления в управляемой позиции. Например, обслуживание биткойнового блокчейна осуществляется сетью узлов связи биткойнового программного обеспечения в области выполнения. Таким образом, вместо банков, учреждений или администраторов в традиционном смысле, множественные посредники существуют в форме компьютеров-серверов, выполняющих биткойновое программное обеспечение. Эти компьютеры-серверы образуют сеть, соединенную через интернет, и потенциально любой может подключиться к сети. Транзакции, производимые сетью, могут иметь форму: "пользователь A хочет отправить Z биткойнов пользователю B," причем транзакции рассылаются в сеть с использованием легкодоступных прикладных программ. Компьютеры-серверы функционируют как биткойновые серверы, которые способны подтверждать эти финансовые транзакции, добавлять запись о них в свою копию бухгалтерской книги, и затем рассылать эти добавления в бухгалтерскую книгу на другие серверы сети.

[30] Поддержание блокчейна именуется "майнингом", и те, которые осуществляют такое поддержание, вознаграждаются вновь созданными биткойнами и вышеупомянутыми транзакционными взносами. Например, узлы могут определять, действительны ли транзакции, на основе набора правил, которым подчиняется блокчейновая сеть. Майнеры могут располагаться на любом континенте и обрабатывать платежи, проверяя действительность каждой транзакции и добавляя ее к блокчейну. Такая проверка достигается через консенсус, обеспеченный множеством майнеров, и предполагает отсутствие систематического сговора. В конце, все данные будут согласованы, поскольку вычисление должно удовлетворять некоторым требованиям, чтобы приводить к правильному результату, и все узлы будут синхронизироваться чтобы гарантировать согласованность блокчейна.

[31] В процессе майнинга, транзакции, например, переводы активов, проверяются и добавляются к растущей цепи блоков блокчейна сетевыми узлами. При обходе всего блокчейна, проверка может включать в себя, например, имеет ли оплачивающая сторона доступ к активу, подлежащему переводу, был ли актив потрачен до этого, верна ли сумма перевода, и т.д. Например, в гипотетической транзакции (например, транзакции биткойнов согласно модели UTXO (неизрасходованного выхода транзакции)), подписанной отправителем, предложенная транзакция может рассылаться в блокчейновую сеть для майнинга. Майнеру нужно проверять, подлежит ли транзакция выполнению, согласно истории блокчейна. Если на балансе кошелька отправителя достаточно средств согласно существующей истории блокчейна, транзакция считается действительной и может добавляться к блоку. После проверки, переводы активов могут быть включены в следующий блок, подлежащий добавлению в блокчейн.

[32] Блок во многом подобен записи базы данных. Каждый раз при записи данных создается блок. Эти блоки связаны и защищены с использованием криптографии, образуя взаимосвязанные сети. Каждый блок соединен с предыдущим блоком, откуда происходит название "блокчейн". Каждый блок обычно содержит криптографический хеш предыдущего блока, время генерации и фактические данные. Например, каждый блок содержит две части: заголовок блока для записи характерного значения текущего блока и тело для записи фактических данных (например, данных транзакции). Блоки цепи связаны заголовками блоков. Каждый заголовок блока может содержать множественные характерные значения, например, версию, хеш предыдущего блока, корень Меркла, метку времени, целевой уровень сложности и нонс (nonce). Хеш предыдущего блока содержит не только адрес предыдущего блока, но и хеш данных внутри предыдущего блока, что обеспечивает неизменность блокчейнов. Нонс представляет собой число, которое, будучи включено, дает хеш с заданным количеством ведущих нулевых битов.

[33] Для майнинга, узел берет хеш содержимого нового блока. Нонс (например, случайная строка) присоединяется к хешу для получения новой строки. Новая строка снова хешируется. Затем окончательный хеш сравнивается с целевым уровнем сложности (например, уровнем), и производится определение, меньше ли фактически окончательный хеш целевого уровня сложности. Если нет, то нонс изменяется, и процесс повторяется снова. Если да, то блок добавляется в цепь, и открытая бухгалтерская книга обновляется и получает объявление о добавлении. Узел, отвечающий за успешное добавление, вознаграждается биткойнами, например, путем добавления транзакции своего вознаграждения в новый блок (известный как генерация coinbase).

[34] Таким образом, для каждого выхода "Y", если k выбирается из распределения с высокой минимальной энтропией, невозможно найти вход x таким образом, что H(k|x) =Y, где K - нонс, x - хеш блока, Y - целевой уровень сложности, и "|" обозначает конкатенацию. Поскольку криптографические хеши, по существу, случайны, в том смысле, что их выход нельзя прогнозировать на основе их входов, найти нонс можно только одним способом: проверять целые числа одно за другим, например, 1, затем 2, затем 3 и т.д., то есть "в лоб". Чем больше количество ведущих нулей, тем больше времени в среднем потребуется, чтобы найти необходимый нонс Y. В одном примере, система биткойна постоянно регулирует количество ведущих нулей, таким образом, что среднее время нахождения нонса составляет около десяти минут. Таким образом, поскольку возможности обработки вычислительного оборудования возрастают со временем, в течение лет, протокол биткойна всегда будет просто требовать больше ведущих нулевых битов, чтобы осуществление майнинга занимало около десяти минут.

[35] Как описано, хеширование является краеугольным камнем блокчейна. Алгоритм хеширования можно понимать как функцию, которая сжимает сообщения произвольной длины в сборник сообщений заверенной длины. Более широко используются MD5 и SHA. В некоторых вариантах осуществления, хеш блокчейна имеет длину 256 битов, и это означает, что независимо от первоначального содержимого, в конце концов вычисляется 256-битовое двоичное число. Можно гарантировать, что соответствующий хеш уникален при условии, что первоначальное содержимое отличается. Например, хеш строки "123" равен a8fdc205a9f19cc1c7507a60c4f01b13d11d7fd0 (в шестнадцатеричном формате), что составляет 256 битов будучи преобразован в двоичный формат, и только "123" имеет этот хеш. Алгоритм хеширования в блокчейне необходим, то есть прямое вычисление осуществляется легко (из "123" в a8fdc205a9f19cc1c7507a60c4f01b1c7507a60c4f01b13d11d7fd0), и обратное вычисление

неосуществимо даже если израсходовать все вычислительные ресурсы. Таким образом, хеш каждого блока блокчейна уникален.

[36] Дополнительно, если содержимое блока изменяется, его хеш также изменяется. Существует взаимно-однозначное соответствие между блоком и хешем, и хеш каждого блока вычисляется в зависимости от заголовка блока. Таким образом, характерные значения заголовков блоков соединяются для формирования длинной строки, и затем вычисляется хеш строки. Например, "Хеш=SHA256 (заголовок блока)" - формула вычисления хеша блока, SHA256 - алгоритм хеширования блокчейна, применяемый к заголовку блока. Хеш уникально определяется заголовком блока, но не телом блока. Как упомянуто выше, заголовок блока содержит много содержимого, включая хеш текущего блока и хеш предыдущего блока. Это означает, что, если содержимое текущего блока изменяется, или если хеш предыдущего блока изменяется, это приведет к изменению хеша в текущем блоке. Если взломщик изменяет блок, хеш этого блока изменяется. Для соединения более позднего блока с измененным блоком, взломщик должен последовательно изменить все последующие блоки, поскольку следующий блок должен содержать хеш предыдущего блока. В противном случае измененный блок будет отсоединяться от блокчейна. По соображениям конструкции, вычисления хеша требуют длительного времени, и почти невозможно изменить множественные блоки за короткий период времени, если только взломщик не завладеет более 51% вычислительной мощности всей сети. Таким образом, блокчейн гарантирует свою собственную надежность, и как только данные записаны, с ним нельзя ничего сделать.

[37] Как только майнер находит хеш (то есть пригодную подпись или решение) для нового блока, майнер рассылает эту подпись всем остальным майнерам (узлам блокчейна). Теперь другие майнеры проверяют, в свою очередь, соответствует ли это решение задаче блока отправителя (то есть, определяют, приводит ли фактически ввод хеша к этой подписи). Если решение верно, другие майнеры подтвердят решение и согласятся добавить новый блок в блокчейн. Таким образом, достигается консенсус нового блока. Это также называется "доказательством выполнения работы". Теперь блок, для которого достигнут консенсус, можно добавить в блокчейн и разослать на все узлы в сети совместно с его подписью. Узлы будут принимать блок и сохранять его в своих данных транзакции при условии, что транзакции внутри блока точно соответствуют текущим балансам кошелька (истории транзакций) в этот момент времени. Каждый раз при добавлении нового блока поверх этого блока, добавление также рассматривается как другое "подтверждение" для блоков до него. Например, если транзакция включена в блок 502, и длина блокчейна составляет 507 блоков, это означает, что транзакция имеет пять подтверждений (соответствующих блокам 507-502). Чем больше подтверждений имеет транзакция, тем труднее злоумышленникам изменить ее.

[38] В некоторых вариантах осуществления, иллюстративная система активов блокчейна использует криптографию открытого ключа, в которой генерируются два криптографических ключа, один открытый ключ и один личный ключ. Открытый ключ можно рассматривать как номер счета, и личный ключ можно рассматривать как права собственности. Например, биткойновый кошелек является совокупностью открытых и личных ключей. Владение активом (например, цифровой валютой, наличным активом, акциями, собственными средствами, облигации) связанным с некоторым адресом актива можно продемонстрировать знанием личного ключа, принадлежащего адресу. Например, программное обеспечение биткойнового кошелька, иногда именуемое "клиентским программным обеспечением биткойна", позволяет данному пользователю

совершать транзакции в биткойнах. Программа кошелька генерирует и сохраняет личные ключи и осуществляет связь с равноправными устройствами в биткойновой сети.

[39] В блокчейновых транзакциях, отправители и получатели платежа идентифицируются в блокчейне своими открытыми криптографическими ключами. Например, наиболее современные переводы биткойнов совершаются от одного открытого ключа к другому открытому ключу. На практике хеши этих ключей используются в блокчейне и называются "адресами биткойнов". В принципе, гипотетическое злонамеренное лицо S может похитить денежные средства у лица A, просто добавив в блокчейновую бухгалтерскую книгу транзакцию наподобие "лицо A уплачивает лицу S 100 биткойнов" с использованием адресов биткойнов пользователей вместо их имен. Протокол биткойна препятствует хищению такого рода, требуя, чтобы каждый перевод был снабжен цифровой подписью в виде личного ключа плательщика, и чтобы в блокчейновую бухгалтерскую книгу добавлялись только подписанные переводы. Поскольку лицо S не может подделать подпись лица A, лицо S не может обмануть лицо A путем добавления в блокчейн элемент, эквивалентный "лицо A уплачивает лицу S 200 биткойнов". В то же время, любой может проверить с использованием своего открытого ключа подпись лица A и, таким образом, уполномочено совершать любую транзакцию в блокчейне, где оно является плательщиком.

[40] В контексте биткойновой транзакции, для перевода некоторой суммы в биткойнах пользователю B, пользователь A может построить запись, содержащую информацию о транзакции через узел. Запись может быть подписана ключом подписания (личным ключом) пользователя A и содержать открытый проверочный ключ пользователя A и открытый проверочный ключ пользователя B. Подпись используется для подтверждения, что транзакция пришла от пользователя, и также препятствует изменению транзакции после ее выдачи. Запись, связанная с другой записью, которая возникала в том же временном окне в новом блоке, может рассылаться на полные узлы. Приняв записи, полные узлы могут включать в бухгалтерскую книгу записи всех транзакций, когда-либо произошедших в блокчейновой системе, добавлять новый блок в ранее принятый блокчейн через вышеописанный процесс майнинга, и подтверждать добавленный блок согласно правилам консенсуса сети.

[41] Актив пользователя A, подлежащий переводу, может иметь форму UTXO (неизрасходованного выхода транзакции). UTXO является объектной моделью блокчейна. Согласно UTXO, активы представлены выходами блокчейновых транзакций, которые не были потрачены, которые могут использоваться как входы в новых транзакциях. Чтобы потратить (перевести) актив, пользователь должен подписаться личным ключом. Биткойн является примером цифровой валюты, которая использует модель UTXO. В случае действительной блокчейновой транзакции, неизрасходованные выходы могут использоваться для осуществления дополнительных транзакций. В некоторых вариантах осуществления, только неизрасходованные выходы могут использоваться в дополнительных транзакциях для предотвращения двойного расходования и мошенничества. По этой причине, входы на блокчейне удаляются, когда происходит транзакция, хотя в то же время, выходы создаются в форме UTXO. Эти неизрасходованные выходы транзакции могут использоваться (держателями личных ключей, например, лицами, имеющими кошельки цифровой валюты) в целях будущих транзакций.

[42] Поскольку блокчейн и другие аналогичные бухгалтерские книги полностью

открыты, блокчейн сам по себе не имеет защиты персональной информации. Открытый характер сети P2P означает, что, хотя ее пользователи не идентифицируются по имени, транзакции легко связывать с индивидами и компаниями. Например, в заграничных переводах или в логистических цепочках, типы активов имеют чрезвычайно высокий уровень защиты персональной информации, поскольку на основе информации о типе актива, можно вывести конкретное местоположение и личности сторон транзакции. Тип актива может содержать, например, денежные средства, цифровую валюту, контракт, акт, медицинскую карту, детали потребителя, акции, облигации, собственные средства или тип любого другого актива, который можно описать в цифровой форме. Хотя модель UTXO скрывает личности и объемы транзакций и применялась к Monero и Zcash, тип актива транзакции остается незащищенным. Таким образом, техническая проблема, решаемая настоящим изобретением, состоит в защите онлайновой информации, например, персональной информации типа актива в транзакциях. Раскрытые системы и способы могут быть интегрированы в модель UTXO для обеспечения защиты персональной информации для различного содержимого транзакции.

[43] В ходе транзакций, защита информации важна для защиты персональной информации пользователя, и тип актива транзакции является одним типом информации, которому не хватает защиты. На фиг. 1 показана иллюстративная система 100 для защиты информации, в соответствии с различными вариантами осуществления. Как показано, блокчейновая сеть может содержать множество узлов (например, полные узлы, реализованные в серверах, компьютерах и т.д.). Для некоторой платформы блокчейна (например, NEO), полные узлы с некоторым уровнем права голоса могут именоваться консенсусными узлами, которые предполагают ответственность за проверку транзакции. В этом изобретении, полные узлы, консенсусный узел или другие эквивалентные узлы могут проверять транзакцию.

[44] Также, как показано на фиг. 1, пользователь A и пользователь B могут использовать соответствующие устройства, например, портативные компьютеры и мобильные телефоны, выступающие в роли облегченных узлов, для осуществления транзакций. Например, пользователь A может пожелать совершать транзакции с пользователем B путем перевода некоторого актива со счета пользователя A на счет пользователя B. Пользователь A и пользователь B могут использовать соответствующие устройства с установленным на них надлежащим программным обеспечением блокчейна для транзакции. Устройство пользователя A может именоваться узлом-инициатором A, который инициирует транзакцию с устройством пользователя B, именуемым узлом-получателем B. Узел A может осуществлять доступ к блокчейну путем связи с узлом 1, и узел B может осуществлять доступ к блокчейну путем связи с узлом 2. Например, узел A и узел B могут предоставлять транзакции блокчейну через узел 1 и узел 2 для запрашивания добавления транзакций в блокчейн. Вне блокчейна, узел A и узел B могут иметь другие каналы связи. Например, узел A и узел B могут обмениваться открытыми ключами посредством регулярной связи через интернет.

[45] Каждый из узлов на фиг. 1 может содержать процессор и долговременный машиночитаемый носитель данных, на котором сохранены инструкции, подлежащие исполнению процессором, для предписания узлу (например, процессору узла) осуществлять различные описанные здесь этапы для защиты информации. На каждом узле может быть установлено программное обеспечение (например, программа транзакций) и/или оборудование (например, провода, беспроводные соединения) для осуществления связи с другими узлами и/или другими устройствами. Дополнительные

детали оборудования и программного обеспечения узла описаны ниже со ссылкой на фиг. 5.

[46] Фиг. 2 демонстрирует иллюстративные этапы для инициирования и проверки транзакции, в соответствии с различными вариантами осуществления.

[47] Инициирование транзакции может осуществляться узлом-инициатором. В некоторых вариантах осуществления, каждый тип типа актива может отображаться в уникальную идентификацию или назначаться ему. Например, уникальной идентификацией может быть порядковый номер sn, вычисленный следующим образом:

этап 1.2 sn=Hash(тип актива)

[48] где Hash() - хеш-функция. Дополнительно, тип актива может шифроваться схемой обязательств (например, схемой обязательств Педерсена) следующим образом:

этап 1.3 C(sn)=r×G+sn×H

[49] где r - случайный коэффициент ослепления (альтернативно именуемый коэффициентом связывания), который обеспечивает сокрытие, G и H - публично согласованные генераторы/базисные точки эллиптической кривой, которые могут выбираться произвольно, sn - значение заверения, C(sn) - точка кривой, используемая как заверение и передаваемая контрагенту, и H - другая точка кривой. Таким образом, G и H могут быть параметрами, известными узлам. Генерация H "ничего в рукаве" может генерироваться путем хеширования базисной точки G с помощью хеш-функции, отображающей одну точку в другую с H= Hash(G). H и G - открытые параметры данной системы (например, произвольно сгенерированные точки на эллиптической кривой). Узел-отправитель может открывать H и G всем узлам. Хотя выше приведен пример схемы обязательств Педерсена в форме эллиптической кривой, альтернативно можно использовать различные другие формы схемы обязательств Педерсена или другие схемы обязательств.

[50] Схема обязательств поддерживает секретность данных, но заверяет данные таким образом, чтобы отправитель данных не мог изменять их позже. Если сторона знает только заверяющее значение (например, C(sn)), она не может определить, какие значения основополагающих данных (например, sn) были заверены. Данные (например, sn) и коэффициент ослепления (например, r) могут раскрываться позже (например, узлом-инициатором), и получатель (например, консенсусный узел) заверения может выполнять заверение и удостоверяться в том, что заверенные данные совпадают с раскрытыми данными. Коэффициент ослепления присутствует, поскольку без него, кто-то может попытаться угадать данные.

[51] Схемы обязательств позволяют отправителю (заверяющей стороне) заверять значение (например, sn) таким образом, что заверенное значение остается личным, но может раскрываться в более позднее время, когда заверяющая сторона разглашает необходимый параметр процесса заверения. Сильные схемы обязательств могут предусматривать как сокрытие информации, так и вычислительное связывание. Сокрытие означает указание, что данное значение sn и заверение этого значения C(sn) не подлежат связыванию. Таким образом, C(sn) не должно раскрывать информацию о sn. Зная C(sn), G и H, почти невозможно узнать sn, поскольку r является случайным числом. Схема обязательств является связывающей в отсутствие правдоподобной возможности, что два разные значения могут приводить к одной и той же заверения. Схема обязательств Педерсена является хорошо скрывающей и вычислительно связывающей при предположении дискретного логарифма.

[52] Схема обязательств Педерсена обладает дополнительным свойством: заверения можно добавлять, и сумма набора заверений идентична заверения в отношении суммы

данных (где ключ ослепления установлен как сумма ключей ослепления): C(BF1, data1) +C(BF2, data2) == C(BF1+BF2, data1+data2); C(BF1, data1) - C(BF1, data1) == 0. Другими словами, заверение сохраняет сложение и применяет свойство коммутативности, т.е. схема обязательств Педерсена является аддитивно гомоморфной, т.е. основополагающими данными можно математически манипулировать, как если бы они не были зашифрованы.

[53] В одном варианте осуществления, схему обязательств Педерсена, используемую для шифрования входного значения, можно построить с использованием точек эллиптической кривой. Традиционно, открытый ключ криптографии на основе эллиптических кривых (ECC) создается путем умножения генератора для группы (G) на секретный ключ (r): Pub=rG. Результат может быть преобразован к последовательному виду в качестве 33-байтового массива. Открытые ключи ECC могут обладать свойством аддитивного гомоморфизма, упомянутым выше в отношении схем обязательств Педерсена. То есть: Pub1+Pub2=(r1+r2(mod n))G.

[54] Схема обязательств Педерсена для входного значения может создаваться путем взятия дополнительного генератора для группы (H, в нижеприведенных уравнениях) таким образом, чтобы никто не знал дискретный журнал для второго генератора H в отношении первого генератора G (или наоборот), то есть, чтобы никто не знал x, благодаря чему rG=H. Это может осуществляться, например, с использованием криптографического хеша G для взятия H: H=to_point(SHA256(ENCODE(G))).

[55] При наличии двух генераторов G и H, иллюстративную схему обязательств для шифрования входного значения можно определить как: заверение=rG+aH. Здесь, r может быть секретным коэффициентом ослепления, и a может быть входным значением, являющимся целью заверения. Следовательно, если sn заверено, можно получить вышеописанную схему обязательств C(sn)=r×G+sn×H. Схемы обязательств Педерсена связаны с теорией информации: для любой заверения, существует некоторый коэффициент ослепления, величина которого совпадает с заверением. Схемы обязательств Педерсена могут быть вычислительно защищены от ложной заверения, таким образом, что произвольное отображение невозможно вычислить.

[56] Сторона (узел), заверившая значение, может открывать заверение путем раскрытия первоначального значения sn и коэффициента r, который удовлетворяет уравнению заверения. Сторона, желающая открыть значение C(sn), будет снова вычислять заверение для проверки, что совместно используемое первоначальное значение действительно совпадает с первоначально принятым заверением C(sn). Таким образом, информацию о типе актива можно защищать путем ее отображения в уникальный порядковый номер, и затем ее шифрования схемой обязательств Педерсена. Случайное число r, выбранное при генерации заверения, практически не позволяет никому вывести тип типа актива, который заверен согласно заверяющему значению C (sn).

[57] В некоторых вариантах осуществления, при включении способа защиты информации о типе актива согласно модели UTXO, согласованность типа актива входа (sn_in) и типа актива выхода (sn_out) транзакции можно проверять для определения действительности транзакции. Например, блокчейновые узлы могут отвергать транзакции или блоки, которые не проходят проверку согласованности, то есть sn_in == sn_out. Поскольку тип актива sn зашифрован (например, схемой обязательств Педерсена), проверка согласованности состоит в проверке C(sn_in) == C(sn_out).

[58] В некоторых вариантах осуществления, как показано на фиг. 2, этап 1, транзакция типа UTXO может содержать m входов (например, доступных активы) и n выходов

(например, перенесенные активы и оставшиеся активы). Входы могут обозначаться sn_in_k, где 1≤k≤m и выходы могут обозначаться sn_out_k, где 1≤k≤n. Некоторые из выходов могут переноситься на узел-получатель B, тогда как остальные выходы могут возвращаться на узел-инициатор A. Например, в гипотетической транзакции, пользователь A может иметь в своем кошельке всего 5 биткойнов и 10 облигаций, и для входов транзакции, sn_in_1=Hash(bitcoin) и sn_in_2=Hash(stock). Если пользователь A хочет перевести 3 биткойна пользователю B, для выходов транзакции, sn_out_1=Hash (bitcoin), sn_out_2=Hash(bitcoin), и sn_out_3=Hash(stock), благодаря чему, один из выходов биткойнов (3 биткойна) адресуется пользователю B, и другой выход биткойнов (2 биткойна) и выход акций адресуются обратно пользователю A.

[59] Поэтому, в некоторых вариантах осуществления, тип актива, соответствующий входу, может шифроваться в форме:

C_in_k=r_in_k×G+sn_in_k×H, где 1≤k≤m

[60] Выходной тип актива соответствует форме шифрования:

C_out_k=r_out_k×G+sn_out_k×H, где 1≤k≤n

[61] Когда типы активов скрыты, инициатору транзакции необходимо доказывать узлам (например, полным узлам, консенсусному узлу), что типы входных активов транзакции, соответственно, согласуются с типами выходных активов. Соответственно, полные узлы могут проверять, является ли транзакция действительной.

[62] В некоторых вариантах осуществления, для инициирования транзакции типа UTXO, когда тип актива скрыт схемой обязательств Педерсена, инициатор транзакции может выбирать надлежащие входы и выходы для осуществления нижеследующих этапов 2.1-2.5 (соответствующих фиг. 2, этап 2):

[63] Этап 2.1 вычислять

C_1=C_in_1 - C_in_2,

C_2=C_in_2 - C_in_3,

…

C_(m-1)=C_in_(m-1) -C_in_m,

C_m=C_out_1 - C_out_2,

C_(m+1)=C_out_2 - C_out_3,

…

C_(m+n-2)=C_out_(n-1) -C_out_n,

C_(m+n-1)=C_in_1 - C_out_1;

[64] Этап 2.2 вычислять x=Hash (C_1 || C_2 || C_3 || … || C_(m+n-1)), где "||" представляет конкатенацию;

[65] Этап 2.3 вычислять $C=C\_1+x\times C\_2+x^2\times C\_3+\ldots+x^{(m+n-2)}\times C\_(m+n-1)$. Заметим, что члены многочлена могут соответствовать членам на этапе 2.1;

[66] Этап 2.4 вычислять $R=(r\_in\_1 - r\_in\_2)+x \times (r\_in\_2 - r\_in\_3)+x^2 \times (r\_in\_3 - r\_in\_4)+\ldots+x^{(m+n-2)} \times (r\_in\_1 - r\_out\_1)$. Заметим, что члены многочлена могут соответствовать членам на этапе 2.1, например, (r_in_1 - r_in_2) соответствует C_in_1 - C_in_2;

[67] Этап 2.5 открыть R узлам, например, в рассылке информации о транзакции.

[68] В некоторых вариантах осуществления, для проверки согласованности типов входных активов и типов выходных активов, должно быть C=R × G. Например, в ходе проверки транзакции, узлы осуществляют нижеследующие этапы 3.1-3.3 (соответствующие фиг. 2, этапы 3.1-3.3) для проверки согласованности типа актива транзакции:

[69] Этап 3.1 вычислять x=Hash (C_1 || C_2 || C_3 || … || C_(m+n-1));

[70] Этап 3.2 вычислять $C=C\_1+x \times C\_2+x^2 \times C\_3+\ldots+x^{(m+n-2)} \times C\_(m+n-1)$;

[71] Этап 3.3 проверять $C=R \times G$. Если $C=R \times G$, тип актива является согласованным; в противном случае, тип актива является несогласованным, и транзакция отвергается. В некоторых вариантах осуществления, C(sn) может раскрываться узлам, и алгоритмы этапов 2.1-2.3 известны узлам (например, включающим в себя узел предоставления транзакции и узлы проверки транзакции). Таким образом, узлы проверки транзакции могут осуществлять этапы 3.1-3.3, соответственно, для осуществления проверки. Таким образом, отвергнутая транзакция не будет добавляться в блокчейн. Согласно этапу 4 на фиг. 2, на основе определения согласованности, узлы могут определять, добавлять ли транзакцию в блокчейн или отвергать добавление транзакции.

[72] По этой причине, инициатор транзакции может предоставлять информацию для блокчейновых узлов для проверки транзакции на основе согласованности типов активов, вводимых и выводимых из транзакции без раскрытия фактических типов активов и без возможности изменения предоставленной информации. Выделение порядковых номеров (например, хешей) для каждого типа актива увеличивает и рандомизирует представление каждого типа актива, что затрудняет инициатору транзакции фальсификацию типа актива для прохождения проверки. Дополнительно, вследствие существования случайного числа r, активы одного и того же типа, зашифрованные в разные моменты времени, не одинаковы. Применение схемы обязательств Педерсена для шифрования хеша типа актива повышает уровень защиты персональной информации типа актива. Таким образом, на этапах 2.1-2.5, инициатор транзакции может доказывать другим узлам, что типы активов транзакции действительны, не раскрывая типы активов. Например, получаются разности между типами входных и выходных активов, на основе которых строятся многочлены, что позволят инициатору транзакции передавать преобразованные типы активов другим узлам для доказательства согласованности типов активов и действительности транзакции. В то же время, вероятностью того, что инициатор транзакции или другой узел способен фальсифицировать тип актива, можно пренебречь, поскольку x вычисляется путем хеширования, чтобы служить основанием различных показательных функций в многочленах. Кроме того, раскрытие R позволяет другим узлам проверять согласованность типов активов в транзакции без известности типов активов на этапах 3.1-3.3. Поэтому раскрытые системы и способы позволяют третьим сторонам проверять информацию данных, поддерживая при этом исключительную защиту персональной информации.

[73] Фиг. 3 демонстрирует блок-схему операций иллюстративного способа 300 для защиты информации, согласно различным вариантам осуществления настоящего изобретения. Способ 300 может осуществляться одним или более компонентами (например, узлом A) системы 100, показанной на фиг. 1. Способ 300 может осуществляться системой или устройством (например, компьютером), содержащим процессор и долговременный машиночитаемый носитель данных (например, память), на котором сохранены инструкции, подлежащие выполнению процессором, для предписания системе или устройству (например, процессору) осуществлять способ 300. Операции способа 300, представленного ниже, предназначены для иллюстрации. В зависимости от реализации, иллюстративный способ 300 может включать в себя больше, меньше или альтернативные этапы, осуществляемые в различном порядке или параллельно.

[74] Блок 301 содержит: определение одного или более вводов данных и одного или более выводов данных для транзакции, причем вводы данных связаны с типами входных данных соответственно, и выводы данных связаны с типами выходных данных

соответственно. См., например, этап 1 на фиг. 2. В некоторых вариантах осуществления, транзакция основана, по меньшей мере, на модели неизрасходованных выходов транзакции (UTXO); и вводы данных и выводы данных содержат типы одного или более активов, подвергающихся транзакции между отправителем (узлом-инициатором) и получателем (узлом-получателем). Тип актива может содержать, например, денежные средства, цифровую валюту, контракт, акт, медицинскую карту, детали потребителя, акции, облигации, собственные средства, или тип любого другого актива, который можно описать в цифровой форме.

[75] Блок 302 содержит: шифрование типов входных данных и типов выходных данных. См., например, вышеописанный этап 1.2. В некоторых вариантах осуществления, шифрование типов входных данных и типов выходных данных содержит шифрование каждый из типы входных данных и типы выходных данных с помощью хеш-функции или другой односторонней функцией.

[76] Блок 303 содержит: заверение каждого из зашифрованных типов входных данных и зашифрованных типов выходных данных с помощью схемы обязательств для получения соответствующих заверяющих значений. См., например, вышеописанный этап 1.3. В некоторых вариантах осуществления, схема обязательств содержит схему обязательств Педерсена. В некоторых вариантах осуществления, схема обязательств содержит, по меньшей мере, коэффициент ослепления; и коэффициент ослепления изменяется в зависимости от времени заверения зашифрованных типов входных данных и зашифрованных типов выходных данных. Таким образом, даже одни и те же данные (например, данные одного и того же типа) заверенные в разные моменты времени будут отличаться заверяющими значениями вследствие изменения коэффициента ослепления.

[77] Блок 304 содержит: получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений. См., например, вышеописанные этапы 2.1-2.4. В некоторых вариантах осуществления, схема обязательств содержит множество коэффициентов ослепления, соответственно соответствующих типам входных данных и типам выходных данных (см., например, r_in_k и r_out_k); и получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений содержит: получение разностей между парами заверяющих значений (см., например, этап 2.1 для различных пар заверяющих значений между типами входных активов и типами выходных активов, для которых можно получить разности); конкатенацию полученных разностей (см., например, этап 2.2); шифрование конкатенированных разностей с помощью хеш-функции для получения шифрованного значения x (см., например, этап 2.2); и получение параметра R на основе, по меньшей мере, шифрованного значения x и разностей между парами коэффициентов ослепления (см., например, этап 2.4).

[78] Блок 305 содержит: предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных. В некоторых вариантах осуществления, узлам предписывается проверять согласованность между типами входных данных и типами выходных данных без известности типов входных данных и типы выходных данных.

[79] В некоторых вариантах осуществления, предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных содержит предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и

без раскрытия типов входных данных и типов выходных данных для предписания узлам: получать параметр R и базисную точку G (см., например, G на этапе 3.1. H и G могут быть открытыми параметрами, доступными всем узлам); получать разности между парами заверяющих значений типов входных активов и типы выходных активов (см., например, этап, аналогичный этапу 2.1); конкатенировать полученные разности (см., например, этап 3.1); шифровать конкатенированные разности с помощью хеш-функции для получения шифрованного значения x (см., например, этап 3.1); получать сумму C многочленов на основе, по меньшей мере, полученных разностей и шифрованного значения x (см., например, этап 3.2); в ответ на определение, что сумма C равна произведению параметра R и базисной точки G, определять, что типы входных данных и типы выходных данных согласованы и добавлять транзакцию в блокчейн (см., например, этап 3.3); и в ответ на определение, что сумма C не равна произведению параметра R и базисной точки G, определять, что типы входных данных и типы выходных данных не согласованы и отвергать добавление транзакции в блокчейн (см., например, этап 3.3).

[80] Фиг. 4 демонстрирует блок-схему операций иллюстративного способа 400 для защиты информации, согласно различным вариантам осуществления настоящего изобретения. Способ 400 может осуществляться одним или более компонентами (например, узлом i) системы 100, показанной на фиг. 1. Узел i может содержать полный узел, реализованный на сервере. Способ 400 может осуществляться системой или устройством (например, компьютером), содержащим процессор и долговременный машиночитаемый носитель данных (например, память), на котором сохранены инструкции, подлежащие выполнению процессором, для предписания системе или устройству (например, процессору) осуществлять способ 400. Операции способа 400, представленного ниже, предназначены для иллюстрации. В зависимости от реализации, иллюстративный способ 400 может включать в себя больше, меньше или альтернативные этапы, осуществляемые в различном порядке или параллельно.

[81] Блок 401 содержит: получение, одним или более узлами (например, консенсусным узлом) в блокчейновой сети, транзакции, инициированной узлом-инициатором. Транзакция связана с одним или более вводами данных и одним или более выводами данных. Вводы данных, соответственно, связаны с типами входных данных, и выводы данных, соответственно, связаны с типами выходных данных соответственно. Типы входных данных и типы выходных данных шифруются и заверяются согласно схеме обязательств для получения соответствующих заверяющих значений. Типы входных данных и типы выходных данных не раскрываются одному или более узлам.

[82] Блок 402 содержит: проверку, одним или более узлами, согласованности между типами входных данных и типами выходных данных. В некоторых вариантах осуществления, проверка согласованности между типами входных данных и типами выходных данных содержит: получение параметра R и базисной точки G (см., например, R на этапах 2.4 и 2.5, G на этапе 3.1); получение разностей между парами заверяющих значений типов входных активов и типы выходных активов (см., например, этап, аналогичный этапу 2.1); конкатенацию полученных разностей (см., например, этап 3.1); шифрование конкатенированных разностей с помощью хеш-функции для получения шифрованного значения x (см., например, этап 3.1); получение суммы C многочленов на основе, по меньшей мере, полученных разностей и шифрованного значения x (см., например, этап 3.2); и определение, равна ли сумма C произведению параметра R и базисной точки G (см., например, этап 3.3).

[83] Блок 403 содержит: в ответ на определение, что типы входных данных и типы

выходных данных согласованы, добавление, одним или более узлами, транзакции в блокчейновую сеть.

[84] Блок 404 содержит: в ответ на определение, что типы входных данных и типы выходных данных не согласованы, отказ, одним или более узлами, от добавления транзакции к блокчейновой сети.

[85] В некоторых вариантах осуществления, способ дополнительно содержит: в ответ на определение, что сумма C равна произведению параметра R и базисной точки G, определение, что типы входных данных и типы выходных данных согласованы; и в ответ на определение, что сумма C не равна произведению параметра R и базисной точки G, определение, что типы входных данных и типы выходных данных не согласованы.

[86] По этой причине, инициатор транзакции может предоставлять информацию для блокчейновых узлов для проверки транзакции на основе согласованности типов активов, вводимых и выводимых из транзакции без раскрытия фактических типов активов и без возможности изменения предоставленной информации. Выделение порядковых номеров (например, хешей) для каждого типа актива увеличивает и рандомизирует представление каждого типа актива, что затрудняет инициатору транзакции фальсификацию типа актива для прохождения проверки. Дополнительно, вследствие существования случайного числа r, активы одного и того же типа, зашифрованные в разные моменты времени, не одинаковы. Применение схемы обязательств Педерсена для шифрования хеша типа актива повышает уровень защиты персональной информации типа актива. Таким образом, на этапах 2.1-2.5, инициатор транзакции может доказывать другим узлам, что типы активов транзакции действительны, не раскрывая типы активов. Например, получаются разности между типами входных и выходных активов, на основе которых строятся многочлены, что позволят инициатору транзакции передавать преобразованные типы активов другим узлам для доказательства согласованности типов активов и действительности транзакции. В то же время, вероятностью того, что инициатор транзакции или другой узел способен фальсифицировать тип актива, можно пренебречь, поскольку x вычисляется путем хеширования, чтобы служить основанием различных показательных функций в многочленах. Кроме того, раскрытие R позволяет другим узлам проверять согласованность типов активов в транзакции без известности типов активов, на этапах 3.1-3.3. Поэтому раскрытые системы и способы позволяют третьим сторонам проверять информацию данных, поддерживая при этом исключительную защиту персональной информации.

[87] Описанные здесь методы реализуются одним или более вычислительными устройствами специального назначения. Вычислительные устройства специального назначения могут быть настольными компьютерными системами, серверными компьютерными системами, портативными компьютерными системами, карманными устройствами, сетевыми устройствами или любым другим устройством или комбинацией устройств, которые включают в себя аппаратную и/или программную логику для осуществления методов. Вычислительное(ые) устройство(а), в целом, управляются и координируются программным обеспечением операционной системы. Традиционные операционные системы управляют выполнением компьютерных процессов и планируют его, осуществляют управление памятью, обеспечивают файловую систему, сетевую связь, службы I/O и обеспечивают функциональные возможности пользовательского интерфейса, например, графического пользовательского интерфейса ("GUI"), помимо прочего.

[88] На фиг. 5 показана блок-схема, которая демонстрирует компьютерную систему

500, в которой можно реализовать любой из описанных здесь вариантов осуществления. Система 500 может быть реализована в любом из описанных здесь узлов и выполнена с возможностью осуществлять соответствующие этапы способов защиты информации. Компьютерная система 500 включает в себя шину 502 или другой механизм связи для передачи информации, один или более аппаратных процессоров 504, соединенных с шиной 502 для обработки информации. Аппаратные процессоры 504 могут представлять собой, например, один или более микропроцессоров общего назначения.

[89] Компьютерная система 500 также включает в себя основную память 506, например, оперативную память (RAM), кэш-память и/или другие динамические запоминающие устройства, подключенные к шине 502 для сохранения информации и инструкций, подлежащих выполнению процессором(ами) 504. Основная память 506 также может использоваться для сохранения временных переменных или другой промежуточной информации в ходе выполнения инструкций, подлежащих выполнению процессором(ами) 504. Такие инструкции, хранящиеся на носителях данных, доступных процессору(ам) 504, превращают компьютерную систему 500 в машину специального назначения, которая специализирована для осуществления операций, заданных в инструкциях. Компьютерная система 500 дополнительно включает в себя постоянную память (ROM) 508 или другое статическое запоминающее устройство, подключенное к шине 502, для хранения статической информации и инструкций для процессора(ов) 504. Запоминающее устройство 510, например, магнитный диск, оптический диск или карта флэш-памяти (флэш-носитель) на USB и т.д., обеспечивается и подключается к шине 502 для сохранения информации и инструкций.

[90] Компьютерная система 500 может осуществлять описанные здесь методы с использованием специализированной аппаратной логики, один или более ASIC или FPGA, программно-аппаратное обеспечение и/или программная логика, которая, совместно с компьютерной системой, предписывает или программирует компьютерную систему 500 быть машиной специального назначения. Согласно одному варианту осуществления, описанные здесь операции, способы и процессы осуществляются компьютерной системой 500, когда процессор(ы) 504 выполняют одну или более последовательностей из одной или более инструкций, содержащихся в основной памяти 506. Такие инструкции могут считываться в основную память 506 с другого носителя данных, например, запоминающего устройства 510. Выполнение последовательностей инструкций, содержащихся в основной памяти 506, предписывает процессору(ам) 504 осуществлять процесс описанные здесь этапы. В альтернативных вариантах осуществления, аппаратная схема может использоваться вместо программных инструкций или совместно с ними.

[91] Основная память 506, ROM 508 и/или хранилище 510 может включать в себя долговременные носители данных. Используемый здесь термин "долговременные носители" и аналогичные термины, означает носители, где хранятся данные и/или инструкции, которые предписывают машине действовать тем или иным образом, носители исключают переходные (кратковременные) сигналы. Такие долговременные носители могут содержать энергонезависимые носители и/или энергозависимые носители. Энергонезависимые носители включает в себя, например, оптические или магнитные диски, например, запоминающее устройство 510. Энергозависимые носители включают в себя динамическую память, например, основную память 506. Общие формы долговременных носителей включают в себя, например, флоппи-диск, гибкий диск, жесткий диск, твердотельный привод, магнитную ленту или любой другой магнитный носитель данных, CD-ROM, любой другой оптический носитель данных, любой

физический носитель с шаблонами отверстий, RAM, PROM, EPROM, FLASH-EPROM, NVRAM, любую другую микросхему памяти или картриджа и их сетевые версии.

[92] Компьютерная система 500 также включает в себя сетевой интерфейс 518, подключенный к шине 502. Сетевой интерфейс 518 обеспечивает двустороннюю передачу данных, подключающуюся к одной или более сетевым линиям связи, которые подключены к одной или более локальным сетям. Например, сетевой интерфейс 518 может представлять собой карту цифровой сети связи с комплексными услугами (ISDN), кабельный модем, спутниковый модем или модем для обеспечения соединения для передачи данных для соответствующего типа телефонной линии. В порядке другого примера, сетевой интерфейс 518 может представлять собой карту локальной сети (LAN) для обеспечения соединения для передачи данных для совместимой LAN (или компонента WAN, соединенного с WAN). Также можно реализовать беспроводные линии связи. В любой подобной реализации, сетевой интерфейс 518 отправляет и принимает электрические, электромагнитные или оптические сигналы, которые несут потоки цифровых данных, представляющие различные типы информации.

[93] Компьютерная система 500 может отправлять сообщения и принимать данные, включающий в себя программный код, через сеть(и), сетевую линию связи и сетевой интерфейс 518. В примере интернета, сервер может передавать запрошенный код для прикладной программы через интернет, ISP, локальную сеть и сетевой интерфейс 518.

[94] Принятый код может выполняться процессором(ами) 504, когда он принимается и/или сохраняется в запоминающем устройстве 510 или другом энергонезависимом хранилище для выполнения в дальнейшем.

[95] Каждый из процессов, способов и алгоритмов, описанных в предыдущих разделах, может быть реализован и полностью или частично автоматизироваться в модулях кода, исполняемых одной или более компьютерными системами или компьютерными процессорами, содержащими компьютерное оборудование. Процессы и алгоритмы могут осуществляться частично или полностью в схеме специального назначения.

[96] Различные вышеописанные признаки и процессы можно использовать независимо друг от друга или можно комбинировать по-разному. Все возможные комбинации и подкомбинации подлежат включению в объем этого изобретения. Кроме того, в некоторых реализациях некоторые блоки способа или процесса могут быть исключены. Описанные здесь способы и процессы также не ограничиваются никакой конкретной последовательностью, и блоки или связанные с ними состояния могут осуществляться в других надлежащих последовательностях. Например, описанные блоки или состояния могут осуществляться в порядке, отличном от конкретно раскрытого, или множественные блоки или состояния можно комбинировать в единый блок или состояние. Иллюстративные блоки или состояния могут осуществляться последовательно, параллельно, или каким-либо другим способом. Блоки или состояния могут добавляться в раскрытые иллюстративные варианты осуществления или удаляться из них. Иллюстративные системы и компоненты, описанный здесь, могут быть сконфигурированы иначе, чем описаны. Например, элементы могут добавляться, удаляться или переставляться по сравнению с раскрытыми иллюстративными вариантами осуществления.

[97] Различные операции описанных здесь иллюстративных способов могут осуществляться, по меньшей мере, частично, посредством алгоритма. Алгоритм может содержаться в программных кодах или инструкциях, хранящихся в памяти (например, вышеописанного долговременного машиночитаемого носителя данных). Такой алгоритм может содержать алгоритм машинного обучения. В некоторых вариантах

осуществления, алгоритм машинного обучения может в явном виде не программировать компьютеры для осуществления функции, но может осуществляться из обучающих данных для создания прогностической модели, которая осуществляет функцию.

[98] Различные операции описанных здесь иллюстративных способов может осуществляться, по меньшей мере, частично, одним или более процессорами, которые временно конфигурируются (например, программным обеспечением) или постоянно конфигурируются для осуществления соответствующих операций. Будучи сконфигурированы на временной или постоянной основе, такие процессоры могут образовывать процессорно-реализованные машины, которые действуют для осуществления описанных здесь одной или более операций или функций.

[99] Аналогично, описанные здесь способы могут быть, по меньшей мере, частично процессорно-реализованы, причем конкретный процессор или процессоры являются примером оборудования. Например, по меньшей мере, некоторые из операций способа могут осуществляться одним или более процессорами или процессорно-реализованными машинами. Кроме того, один или более процессоров также могут действовать для поддержки эксплуатационных показателей соответствующих операций в среде "облачных вычислений" или в качестве "программного обеспечения как услуги" (SaaS). Например, по меньшей мере, некоторые из операций могут осуществляться группой компьютеров (в качестве примеров машин, включающих в себя процессоры), причем эти операции доступны через сеть (например, интернет) и через один или более надлежащих интерфейсов (например, интерфейс прикладных программ (API)).

[100] Эксплуатационные показатели некоторых из операций могут распределяться между процессорами, не только присутствующими в единой машине, но установленными на нескольких машинах. В некоторых иллюстративных вариантах осуществления, процессоры или процессорно-реализованные машины могут располагаться в одном географическом положении (например, в домашней среде, офисной среде или серверной ферме). В других иллюстративных вариантах осуществления, процессоры или процессорно-реализованные машины могут распределяться в нескольких географических положениях.

[101] В этом описании изобретения, множественные экземпляры могут осуществлять компоненты, операции или структуры, описанные как единственный экземпляр. Хотя отдельные операции одного или более способов проиллюстрированы и описаны как отдельные операции, одна или более из отдельных операций могут осуществляться одновременно, и не требуется, чтобы операции осуществлялись в проиллюстрированном порядке. Структуры и функциональные возможности, представленные как отдельные компоненты в иллюстративных конфигурациях, могут осуществляться как комбинированная конструкция или компонент. Аналогично, структуры и функциональные возможности, представленные в виде единого компонента, могут осуществляться как отдельные компоненты. Эти и другие вариации, модификации, добавления и усовершенствования подлежат включению в объем настоящего изобретения.

[102] Хотя настоящее изобретение описано со ссылкой на конкретные иллюстративные варианты осуществления, различные модификации и изменения можно вносить в эти варианты осуществления, не выходя за рамки более широкого объема вариантов осуществления настоящего изобретения. Такие варианты осуществления изобретения может именоваться здесь, по отдельности или совместно, термином "изобретение" лишь для удобства и без намерения добровольно ограничивать объем данной заявки никаким отдельным раскрытием или понятием, если фактически раскрыто

более одного.

(57) Формула изобретения

1. Компьютерно-реализуемый способ защиты информации, содержащий этапы, на которых:

определяют один или более вводов данных и один или более выводов данных для транзакции, причем вводы данных связаны с типами входных данных соответственно, и выводы данных связаны с типами выходных данных соответственно;

шифруют типы входных данных и типы выходных данных;

заверяют каждый из зашифрованных типов входных данных и зашифрованных типов выходных данных с помощью схемы обязательств для получения соответствующих заверяющих значений;

получают, по меньшей мере, параметр R на основе, по меньшей мере, заверяющих значений; и

предоставляют транзакцию одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных.

2. Способ по п.1, в котором шифрование типов входных данных и типов выходных данных содержит этап, на котором шифруют типы входных данных и типы выходных данных с помощью хеш-функции.

3. Способ по п.1, в котором схема обязательств содержит схему обязательств Педерсена.

4. Способ по п.1, в котором:

схема обязательств содержит, по меньшей мере, коэффициент ослепления; и

коэффициент ослепления изменяется в зависимости от времени заверения зашифрованных типов входных данных и зашифрованных типов выходных данных.

5. Способ по п.1, в котором узлам предписывают проверять согласованность между типами входных данных и типами выходных данных без известности типов входных данных и типов выходных данных.

6. Способ по п.1, в котором:

транзакция основана, по меньшей мере, на модели неизрасходованных выходов транзакции (UTXO); и

вводы данных и выводы данных содержат типы одного или более активов, подвергающихся транзакции.

7. Способ по п.1, в котором:

схема обязательств содержит множество коэффициентов ослепления, соответственным образом соответствующих типам входных данных и типам выходных данных; и

получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений содержит этапы, на которых:

получают разности между парами заверяющих значений;

конкатенируют полученные разности;

шифруют конкатенированные разности с помощью хеш-функции для получения шифрованного значения x; и

получают параметр R на основе, по меньшей мере, шифрованного значения x и разностей между парами коэффициентов ослепления.

8. Способ по п.1, в котором предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных

и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных содержит этап, на котором предоставляют транзакцию одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для предписания узлам:

получать параметр R и базисную точку G;

получать разности между парами заверяющих значений;

конкатенировать полученные разности;

шифровать конкатенированные разности с помощью хеш-функции для получения шифрованного значения x;

получать сумму C многочленов на основе, по меньшей мере, полученных разностей и шифрованного значения x;

в ответ на определение того, что сумма C равна произведению параметра R и базисной точки G, определять, что типы входных данных и типы выходных данных согласованы; и

в ответ на определение того, что сумма C не равна произведению параметра R и базисной точки G, определять, что типы входных данных и типы выходных данных не согласованы.

9. Долговременный машиночитаемый носитель данных, на котором сохранены инструкции, которые должны исполняться процессором для предписания процессору осуществлять операции, содержащие:

определение одного или более вводов данных и одного или более выводов данных для транзакции, причем вводы данных связаны с типами входных данных соответственно, и выводы данных связаны с типами выходных данных соответственно;

шифрование типов входных данных и типов выходных данных;

заверение каждого из зашифрованных типов входных данных и зашифрованных типов выходных данных с помощью схемы обязательств для получения соответствующих заверяющих значений;

получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений; и

предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных.

10. Носитель данных по п.9, при этом шифрование типов входных данных и типов выходных данных содержит шифрование типов входных данных и типов выходных данных с помощью хеш-функции.

11. Носитель данных по п.9, при этом схема обязательств содержит схему обязательств Педерсена.

12. Носитель данных по п.9, при этом:

схема обязательств содержит, по меньшей мере, коэффициент ослепления; и

коэффициент ослепления изменяется в зависимости от времени заверения зашифрованных типов входных данных и зашифрованных типов выходных данных.

13. Носитель данных по п.9, при этом узлам предписывается проверять согласованность между типами входных данных и типами выходных данных без известности типов входных данных и типы выходных данных.

14. Носитель данных по п.9, при этом:

транзакция основана, по меньшей мере, на модели неизрасходованных выходов

транзакции (UTXO); и

вводы данных и выводы данных содержат типы одного или более активов, подвергающихся транзакции.

15. Носитель данных по п.9, при этом:

схема обязательств содержит множество коэффициентов ослепления, соответственным образом соответствующих типам входных данных и типам выходных данных; и

получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений содержит:

получение разностей между парами заверяющих значений;

конкатенацию полученных разностей;

шифрование конкатенированных разностей с помощью хеш-функции для получения шифрованного значения x; и

получение параметра R на основе, по меньшей мере, шифрованного значения x и разностей между парами коэффициентов ослепления.

16. Носитель данных по п.9, при этом предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных содержит предоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для предписания узлам:

получать параметр R и базисную точку G;

получать разности между парами заверяющих значений;

конкатенировать полученные разности;

шифровать конкатенированные разности с помощью хеш-функции для получения шифрованного значения x;

получать сумму C многочленов на основе, по меньшей мере, полученных разностей и шифрованного значения x;

в ответ на определение того, что сумма C равна произведению параметра R и базисной точки G, определять, что типы входных данных и типы выходных данных согласованы; и

в ответ на определение того, что сумма C не равна произведению параметра R и базисной точки G, определять, что типы входных данных и типы выходных данных не согласованы.

17. Система для защиты информации, содержащая процессор и долговременный машиночитаемый носитель данных, подключенный к процессору, причем на носителе данных хранятся инструкции, которые должны исполняться процессором для предписания системе осуществлять операции, содержащие:

определение одного или более вводов данных и одного или более выводов данных для транзакции, причем вводы данных связаны с типами входных данных соответственно, и выводы данных связаны с типами выходных данных соответственно;

шифрование типов входных данных и типов выходных данных;

заверение каждого из зашифрованных типов входных данных и зашифрованных типов выходных данных с помощью схемы обязательств для получения соответствующих заверяющих значений;

получение, по меньшей мере, параметра R на основе, по меньшей мере, заверяющих значений; и

предоставление транзакции одному или более узлам в блокчейновой сети с

раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных.

18. Компьютерно-реализуемый способ защиты информации, содержащий этапы, на которых:

получают на одном или более узлах в блокчейновой сети транзакцию, инициированную узлом-инициатором, причем:

транзакция связана с одним или более вводами данных и одним или более выводами данных,

вводы данных соответственно связаны с типами входных данных, и выводы данных соответственно связаны с типами выходных данных, соответственно,

типы входных данных и типы выходных данных шифруются и заверяются согласно схеме обязательств для получения соответствующих заверяющих значений, и

типы входных данных и типы выходных данных не раскрываются одному или более узлам;

проверяют на одном или более узлах согласованность между типами входных данных и типами выходных данных;

в ответ на определение того, что типы входных данных и типы выходных данных согласованы, добавляют на одном или более узлах транзакцию в блокчейновую сеть; и

в ответ на определение того, что типы входных данных и типы выходных данных не согласованы, отказываются на одном или более узлах от добавления транзакции к блокчейновой сети.

19. Способ по п.18, в котором проверка согласованности между типами входных данных и типами выходных данных содержит этапы, на которых:

получают параметр R и базисную точку G;

получают разности между парами заверяющих значений;

конкатенируют полученные разности;

шифруют конкатенированные разности с помощью хеш-функции для получения шифрованного значения x;

получают сумму C многочленов на основе, по меньшей мере, полученных разностей и шифрованного значения x; и

определяют, равна ли сумма C произведению параметра R и базисной точки G.

20. Способ по п.19, дополнительно содержащий этапы, на которых:

в ответ на определение того, что сумма C равна произведению параметра R и базисной точки G, определяют, что типы входных данных и типы выходных данных согласованы; и

в ответ на определение того, что сумма C не равна произведению параметра R и базисной точки G, определяют, что типы входных данных и типы выходных данных не согласованы.

21. Долговременный машиночитаемый носитель данных, на котором сохранены инструкции, которые должны исполняться процессором для предписания процессору осуществлять операции, содержащие:

получение, одним или более узлами в блокчейновой сети, транзакции, инициированной узлом-инициатором, причем:

транзакция связана с одним или более вводами данных и одним или более выводами данных,

вводы данных соответственно связаны с типами входных данных, и выводы данных

соответственно связаны с типами выходных данных, соответственно,

типы входных данных и типы выходных данных шифруются и заверяются согласно схеме обязательств для получения соответствующих заверяющих значений, и

типы входных данных и типы выходных данных не раскрываются одному или более узлам;

проверку, одним или более узлами, согласованности между типами входных данных и типами выходных данных;

в ответ на определение того, что типы входных данных и типы выходных данных согласованы, добавление, одним или более узлами, транзакции в блокчейновую сеть; и

в ответ на определение того, что типы входных данных и типы выходных данных не согласованы, отказ одним или более узлами от добавления транзакции к блокчейновой сети.

22. Система для защиты информации, содержащая процессор и долговременный машиночитаемый носитель данных, подключенный к процессору, причем на носителе данных хранятся инструкции, которые должны исполняться процессором для предписания системе осуществлять операции, содержащие:

получение, одним или более узлами в блокчейновой сети, транзакции, инициированной узлом-инициатором, причем:

транзакция связана с одним или более вводами данных и одним или более выводами данных,

вводы данных соответственно связаны с типами входных данных, и выводы данных соответственно связаны с типами выходных данных, соответственно,

типы входных данных и типы выходных данных шифруются и заверяются согласно схеме обязательств для получения соответствующих заверяющих значений, и

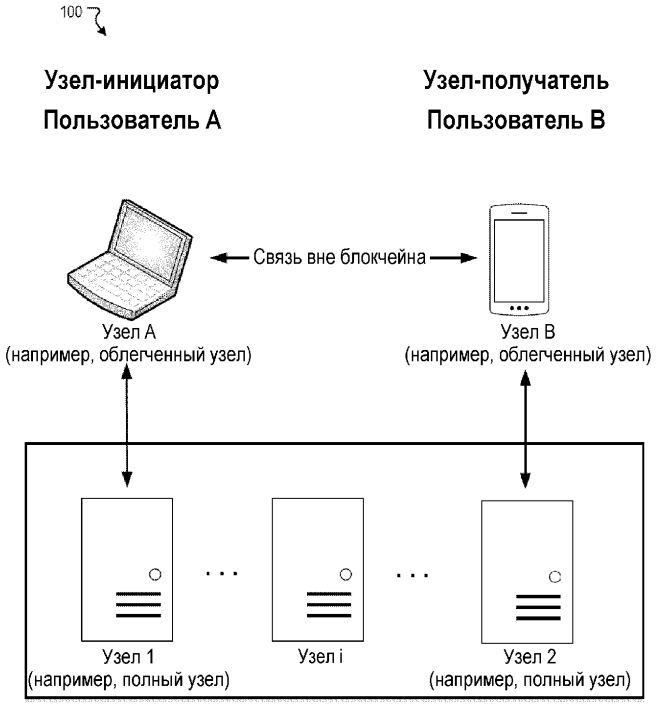
типы входных данных и типы выходных данных не раскрываются одному или более узлам;

проверку, одним или более узлами, согласованности между типами входных данных и типами выходных данных;

в ответ на определение того, что типы входных данных и типы выходных данных согласованы, добавление, одним или более узлами, транзакции в блокчейновую сеть; и

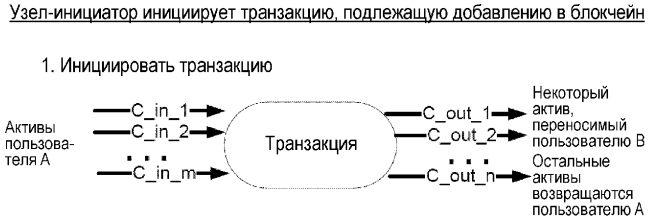
в ответ на определение того, что типы входных данных и типы выходных данных не согласованы, отказ одним или более узлами от добавления транзакции к блокчейновой сети.

1


100
Узел-инициатор
Пользователь A
Узел-получатель
Пользователь B
Связь вне блокчейна
Узел A
(например, облегченный узел)
Узел B
(например, облегченный узел)
Узел 1
(например, полный узел)
Узел i
Узел 2
(например, полный узел)

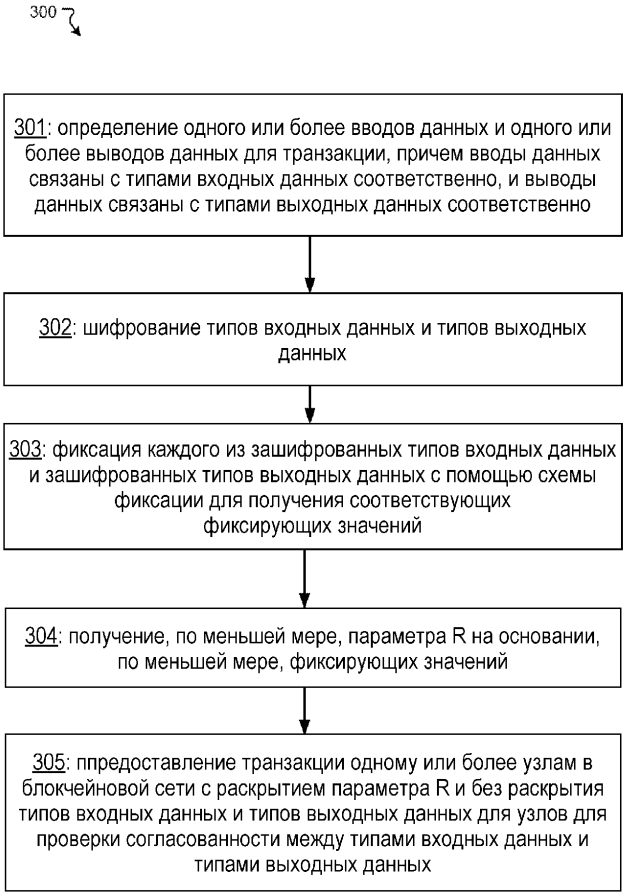
ФИГ.1

2


Узел-инициатор инициирует транзакцию, подлежащую добавлению в блокчейн
1. Инициировать транзакцию
Активы пользователя A
C_in_1
C_in_2
C_in_m
Транзакция
C_out_1
C_out_2
C_out_n
Некоторый актив, переносимый пользователю B
Остальные активы возвращаются пользователю A
2. Вычислять x, C и R, и открыть R узлам
Блок
Блок
Блок
Блок
Консенсусный узел проверяет, является ли транзакция действительной
3.1. Вычислять x
3.2. Вычислять C
3.3. Проверять C=rG для подтверждения транзакции
4. Добавлять транзакции в блокчейн или отвергать на основании действительности

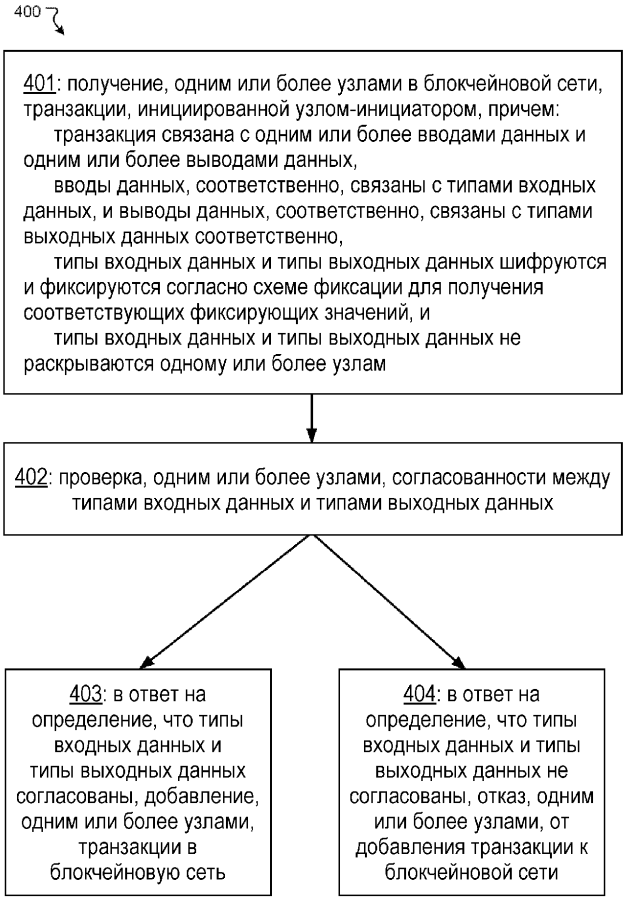
ФИГ.2

3


300
301: определение одного или более вводов данных и одного или более выводов данных для транзакции, причем вводы данных связаны с типами входных данных соответственно, и выводы данных связаны с типами выходных данных соответственно
302: шифрование типов входных данных и типов выходных данных
303: фиксация каждого из зашифрованных типов входных данных и зашифрованных типов выходных данных с помощью схемы фиксации для получения соответствующих фиксирующих значений
304: получение, по меньшей мере, параметра R на основании, по меньшей мере, фиксирующих значений
305: ппредоставление транзакции одному или более узлам в блокчейновой сети с раскрытием параметра R и без раскрытия типов входных данных и типов выходных данных для узлов для проверки согласованности между типами входных данных и типами выходных данных

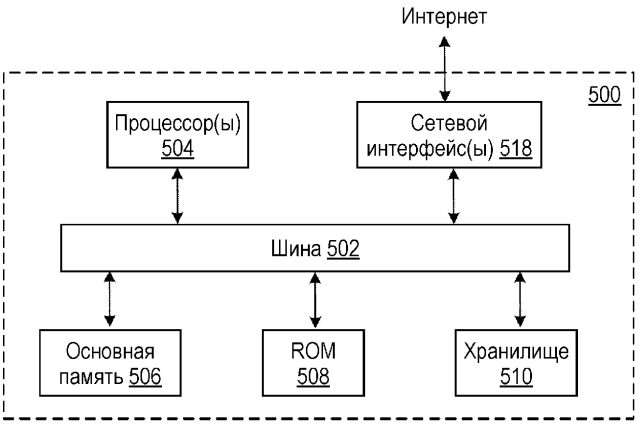
ФИГ.3

4


400
401: получение, одним или более узлами в блокчейновой сети, транзакции, инициированной узлом-инициатором, причем:
транзакция связана с одним или более вводами данных и одним или более выводами данных,
вводы данных, соответственно, связаны с типами входных данных, и выводы данных, соответственно, связаны с типами выходных данных соответственно,
типы входных данных и типы выходных данных шифруются и фиксируются согласно схеме фиксации для получения соответствующих фиксирующих значений, и
типы входных данных и типы выходных данных не раскрываются одному или более узлам
402: проверка, одним или более узлами, согласованности между типами входных данных и типами выходных данных
403: в ответ на определение, что типы входных данных и типы выходных данных согласованы, добавление, одним или более узлами, транзакции в блокчейновую сеть
404: в ответ на определение, что типы входных данных и типы выходных данных не согласованы, отказ, одним или более узлами, от добавления транзакции к блокчейновой сети

ФИГ.4

5


Интернет
500
Процессор(ы) 504
Сетевой интерфейс(ы) 518
Шина 502
Основная память 506
ROM 508
Хранилище 510

ФИГ.5