(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 946 241**

(51) Int. CI.:

G06F 21/62 (2013.01)
H04L 9/32 (2006.01)
G06F 21/64 (2013.01)
H04L 9/40 (2012.01)
H04L 9/00 (2012.01)
G06F 21/60 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **05.12.2019 E 19383082 (5)**

(97) Fecha y número de publicación de la concesión europea: **26.04.2023 EP 3832510**

(54) Título: **Método, sistema y programas informáticos para la ordenación, replicación y registro transparente no repudiable de operaciones que implican datos personales**

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
14.07.2023

(73) Titular/es:

TELEFONICA DIGITAL ESPAÑA, S.L.U. (100.0%)
Gran Via 28
28013 Madrid, ES

(72) Inventor/es:

BREZO FERNÁNDEZ, FÉLIX y
RUBIO VIÑUELA, YAIZA

(74) Agente/Representante:

ARIZTI ACHA, Monica

DESCRIPCIÓN

Método, sistema y programas informáticos para la ordenación, replicación y registro transparente no repudiable de operaciones que implican datos personales

Campo técnico

La presente invención está dirigida, en general, a la tecnología de cadena de bloques. En particular, la invención se refiere a un método, sistema y programas informáticos para la ordenación, replicación y registro transparente no repudiable de operaciones que implican datos personales.

Antecedentes de la invención

La definición del protocolo que trajo consigo Bitcoin en 2008 [1] fue el comienzo de multitud de aplicaciones y casos de uso que se basan en la tecnología que lo hace posible: la cadena de bloques. Desde las primeras definiciones del protocolo Bitcoin, el concepto ha evolucionado mucho más allá de la creación del sistema de pago electrónico entre pares descrito en el documento original al proponer un libro mayor público distribuido que realiza un seguimiento de todas las operaciones en la red para que permanezcan tanto inmutables como verificables. Los bloques minados aproximadamente cada 10 minutos han dado lugar a proyectos como Lightning Network [2] que permiten la realización de pagos prácticamente instantáneos basándose en una red Bitcoin separada.

Pero los cambios han ido mucho más allá de este uso. En Namecoin [3], la cadena de bloques se usa para registrar de forma permanente la información de registro de dominio; en Ethereum [4], se amplía el paradigma introduciendo el concepto de contratos inteligentes, que se programan en un lenguaje de programación específico tal como Solidity con el que los usuarios pueden interactuar y operar de forma autónoma; mientras que en Monero [5] las operaciones registradas en la cadena de bloques pública no exponen ni el saldo transferido ni el origen o destinatario de una transacción como sucede en Bitcoin.

Cada uno con sus matices, el número de protocolos e implementaciones de tecnologías que heredan esta filosofía se cuenta por cientos [6]. Sin embargo, todas ellas comparten un concepto común: el uso de un registro contable distribuido, inmutable y persistente. Se dice que es inmutable porque las operaciones se agrupan en bloques que están criptográficamente encadenados al anterior, de modo que solo modificando un bit en un bloque anterior dará como resultado una cadena completamente diferente además de ser inválida. También se dice que es persistente porque en cada nodo participante se almacena una copia de las operaciones ancladas en la cadena, lo que hace potencialmente imposible censurar todo el registro sin afectar gravemente a otros servicios que operan en la red.

Las soluciones propuestas en Bitcoin en busca de la descentralización asumen que los participantes se encuentran en un entorno hostil. Por ello, la definición de protocolos de consenso que reduzcan la posibilidad de que se censuren sus operaciones ha centrado el debate a lo largo de estos años. Todavía, los usuarios pueden encontrar un servicio que se usará para almacenar datos vinculados a transacciones monetarias, por un lado, y estructuras de datos más complejas, por el otro, incluso cuando no confían en sus pares.

La replicación de información en cadenas de bloques convencionales democratiza el acceso a los sistemas de notarización al ofrecer garantías de la inmutabilidad y persistencia de los datos anclados. Su almacenamiento distribuido y la garantía de inmutabilidad facilitada desde el encadenamiento de bloques es un punto fuerte que permite que un tercero pueda usarlo para probar la existencia de una cierta información en un momento específico de tiempo cuando está anclada en una transacción que está registrada en la cadena de forma verificable.

Estas características significan que la tecnología pueda usarse en casos de uso que requieren esta garantía de inmutabilidad y persistencia, pero han de tener en cuenta que la información generada de esta manera no puede eliminarse o borrarse de ellos ya que toda ella, como en su conjunto, puede ser necesaria para el mantenimiento de una cadena de bloques coherente desde su inicio.

Además, las complejidades son significativas a la hora de garantizar el derecho al olvido de los datos anclados, teniendo en cuenta que la información se copia y mantiene por los diferentes nodos de la red. Por ejemplo, el Reglamento General de Protección de Datos [7] (en adelante, RGPD) establece las obligaciones inexcusables de los responsables del procesamiento de datos personales a la hora de guardar esta información. Precisamente, este respeto a la ley, que hace que no todas las soluciones basadas en tecnología sean implementables en el entorno empresarial ya que no pueden garantizar que la información anclada vinculada a un usuario identificable sea borrable cuando se necesita la información exacta para calcular correctamente el estado actual de la cadena de bloques.

De hecho, ya existe un método para gestionar la información personal que respeta la filosofía de privacidad desde el diseño, así como las normativas vigentes en materia de protección de datos [8]. Sin embargo, este sistema no se centra en el no repudio. Aquellos actores a cargo de administrar o simplemente escribir en la cadena de bloques

autorizada/permisiva aún podrían amarrar operaciones adicionales o datos personales que podrían desvelarse a los implicados a voluntad.

El escenario principal es aquel en el que una organización desea probar a una parte interesada que se ha producido un evento. Esta información podría codificarse con función de troceo y emitirse en cadenas de bloques transparentes tales como Bitcoin o Ethereum. Sin embargo, no puede emitirse desde direcciones que pertenezcan a un usuario ya que la mera existencia de esa dirección o contrato vinculable a él conlleva la operación con datos personales que podrían usarse para identificarlo basándose en hábitos, rutinas u otra información.

Para lograr este objetivo, la organización encargada de anclar los datos en la cadena de bloques podría emitir una transacción firmada digitalmente (probando su integridad y origen) que además incluiría la prueba criptográfica del evento a sellar. Como primer ejercicio de transparencia, este actor podría manejar los datos que conducen a esa prueba.

Sin embargo, este enfoque no es suficiente teniendo en cuenta que este mismo administrador también podría haber anclado individualmente otros vinculados a un usuario que no se revela al perfil involucrado. En este punto, el compromiso con las acciones conocidas podría confirmarse por el usuario, pero el organizador aún podría anclar datos adicionales que no se revelaron al usuario cuando se crearon, lo que conduce a posibles disputas.

Por lo tanto, es importante definir un sistema que garantice que las operaciones que implican datos personales que están amarradas en una cadena de bloques también estén encadenadas a la operación anterior de un usuario. Al hacerlo, el usuario cuyos datos se usan puede garantizar que, incluso cuando el que guarda la cadena de bloques es un tercer actor con posibilidades de enviar cualquier tipo de nuevas transacciones, tiene un compromiso verificable con la información proporcionada por el actor con permisos de escritura al sistema.

Al mismo tiempo, las propuestas de cadena de bloques actuales tienen un límite en términos de operaciones realizadas por segundo. Si las operaciones que implican datos personales necesitan escalar, se necesitan proponer soluciones de funciones de troceo más complejas que las proporcionadas en [8] para abordar estos problemas, una característica que también se detalla en las líneas que siguen.

La solicitud de patente US2019182047 desvela un método para compartir de forma segura información de validación de uno o más archivos de datos almacenados en diferentes servidores en la nube usando tecnología de libro mayor distribuido. El método incluye solicitar acceso a los archivos de datos y calcular una función de troceo de los mismos. Un árbol de Merkle estructurado se construye usando la función de troceo y funciones de troceo adicionales de otros archivos de datos a los que un usuario no ha concedido acceso, pero que ha usado para construir un árbol de Merkle correspondiente para el que el usuario ha comprometido un valor raíz a una cadena de bloques principal. Se comprueba si el valor raíz del árbol de Merkle es el mismo que el que ha comprometido el usuario, y si la función de troceo de los archivos de datos se almacena en un bloque de una cadena de bloques satelital vinculada a la cadena de bloques principal y operada por un subconjunto de nodos de la cadena de bloques principal que confían entre sí.

Descripción de la invención

Para ese fin, la presente invención propone, de acuerdo con un aspecto, un método de ordenación, replicación y registro transparente no repudiable de operaciones que implican datos personales. El método comprende realizar, por un usuario a través de un dispositivo informático, por ejemplo, un teléfono móvil, una operación que usa un nodo informático de controlador de datos, teniendo dicho nodo informático de controlador de datos autorización de escritura en una primera cadena de bloques, siendo la primera cadena de bloques una cadena de bloques permisiva; almacenar, por el nodo informático de controlador de datos, información personal del usuario en un sistema de almacenamiento separado de la primera cadena de bloques; y anclar, por el nodo informático de controlador de datos, una prueba criptográfica de dicha operación en dicha primera cadena de bloques usando un algoritmo de función de troceo que encadena la operación a una operación anterior realizada por dicho usuario, estando anclada la prueba criptográfica junto con una cadena criptográfica de una cierta longitud.

La primera cadena de bloques es permisiva pero también es pública en términos de legibilidad. Es decir, solo la organización puede escribir, pero cualquiera puede leer la información (que nunca es información personal). Esto implica de facto que la organización tiene un compromiso con los datos publicados aunque es la única que puede escribir. Además, pudiendo escribir cualquier cosa, la organización teóricamente podría añadir operaciones maliciosas sin que el usuario lo sepa. Este problema se soluciona encadenando las operaciones de usuario a la anterior. Por lo tanto, el usuario siempre podría rastrear sus operaciones.

En una realización, la cadena criptográfica es una cadena aleatoria. En particular, la cadena aleatoria tiene una longitud de 32 caracteres hexadecimales.

En una realización, la información personal del usuario se almacena en el sistema de almacenamiento separado

usando un mecanismo de índice de operaciones (o txid). La información del usuario se almacena particularmente con un primer campo de datos que detalla la información cuya prueba se ha anclado en la primera cadena de bloques y un segundo campo de datos que incluye la prueba criptográfica.

En una realización, el algoritmo de función de troceo comprende un árbol de Merkle. Los árboles de Merkle facilitas la escalada, lo que permite anclar más operaciones de datos personales en una única transacción de cadena de bloques.

Por lo tanto, con el método propuesto, el usuario podría: aplicar función de troceo a los datos de una operación bajo "datos"; ir a la cadena de bloques permisiva; encontrar el txid; extraer la función de troceo de la operación y compararla con la generada a partir de los datos conocidos; y comparar la función de troceo calculada localmente con la obtenida de la cadena de bloques permisiva. Si coinciden, se dice que los datos proporcionados tienen una marca de tiempo en el bloque en el que se encuentra el txid. Obsérvese que este flujo se refiere a las operaciones cuyas pruebas se almacenan en conceptos básicos de 1 a 1 con respecto a las transacciones (es decir, una operación tiene una función de troceo y cada función de troceo se almacena en una transacción diferente). Para mejorar la escalabilidad, usar árboles de Merkle puede conducir a un sistema de anclaje de prueba de 1 a n (es decir, generar un árbol de Merkle 100 operaciones y almacenar únicamente la función de troceo raíz en la transacción). Esto introduciría un poco de sobrecarga en el almacén de datos convencional, pero es un enfoque válido si el cuello de botella es el repositorio de la cadena de bloques.

En otra realización más, el método también ancla, mediante el nodo informático de controlador de datos, una prueba de dicho anclaje de la prueba criptográfica en una segunda cadena de bloques. Esta segunda cadena de bloques puede ser cualquier cosa, incluyendo cualquier tipo de sistema de plataforma de notarización de terceros. En una realización particular, la segunda cadena de bloques es una cadena de bloques de Bitcoin.

De acuerdo con otro aspecto, la presente invención también proporciona un sistema para ordenar, replicar y registrar de forma transparente y no repudiable operaciones que implican datos personales. El sistema propuesto comprende un dispositivo informático de un usuario, un nodo informático de controlador de datos; una primera cadena de bloques y un sistema de almacenamiento separado de dicha primera cadena de bloques. El nodo informático de controlador de datos tiene autorización de escritura en la primera cadena de bloques, siendo esta última una cadena de bloques permisiva.

En el sistema propuesto, el nodo informático de controlador de datos está configurado para recibir una operación del usuario y almacenar información personal del usuario en dicho sistema de almacenamiento y anclar una prueba criptográfica de dicha operación recibida en la primera cadena de bloques usando un algoritmo de función de troceo que encadena la operación a una operación anterior realizada por el usuario, estando anclada la prueba criptográfica junto con una cadena criptográfica de cierta longitud.

Por lo tanto, la presente invención proporciona una plataforma en la que cada actor con permisos de escritura escribirá pruebas criptográficas de eventos vinculados a una persona/usuario que están encadenados individualmente a eventos anteriores de esa misma persona/usuario, manteniendo cualquier información personal relevante fuera de la cadena en un almacenamiento convencional separado que mantendrá en cadena estas pruebas criptográficas. Al hacerlo, la empresa/organización podría entregar al usuario todos y cada uno de los eventos en los que está implicado e incluso borrar sus datos de los sistemas corporativos mientras el usuario aún puede probar a terceros la integridad de todas sus transacciones.

Otras realizaciones de la invención que se desvelan en el presente documento también incluyen programas de software para realizar las etapas y operaciones de la realización del método resumidas anteriormente y desveladas en detalle a continuación. Más particularmente, un producto de programa informático es una realización que tiene un medio legible por ordenador que incluye instrucciones de programa informático codificadas en el mismo que, cuando se ejecutan en al menos un procesador en un sistema informático, provocan que el procesador realice las operaciones indicadas en el presente documento como realizaciones de la invención.

La tecnología de cadena de bloques ofrece una serie de garantías respecto a la inmutabilidad de la información anclada en ella. Los derechos de las partes interesadas en cuanto a sus datos personales conducen a varias normas que deben ser observadas por quienes los gestionan. Aunque estas obligaciones limitan la viabilidad de algunos casos de uso, la tecnología de cadena de bloques en combinación con otras herramientas criptográficas descritas en este documento sigue facilitando el desarrollo de modelos de relación con el cliente más transparentes.

Por las razones expuestas en este documento, cualquier solución que implemente la tecnología debe tener en cuenta la normativa de protección de datos desde su fase inicial para cumplir apropiadamente con las obligaciones definidas en materia de respeto a los derechos de los ciudadanos. La presente invención aborda el problema separando el almacenamiento de las pruebas criptográficas de existencia de las operaciones que implican los propios datos personales. Por tanto, las principales mejoras de la invención son las siguientes:

- El mantenimiento de un registro ordenado, inmutable y consistente, no repudiable, de todas las operaciones y acciones que una organización realiza con sus usuarios, desde registros operativos comerciales hasta otras operaciones de gestión que implican datos personales, mediante la adquisición y exposición de compromisos públicamente verificables.
- El establecimiento de una arquitectura que permita a la parte interesada tener una copia verificable de todas las operaciones en las que hayan estado implicados sus propios datos personales, con garantía de inmutabilidad que seguiría siendo comprobable por terceros.
- El mantenimiento de la trazabilidad anterior respetando estrictamente el derecho a solicitar el borrado de los datos personales a la empresa que adquirió los compromisos, permitiendo al usuario probar su existencia con la única presentación del registro completo de las operaciones que los implican.

Breve descripción de los dibujos

Las anteriores y otras ventajas y características se entenderán más completamente a partir de la siguiente descripción detallada de las realizaciones, con referencia a las figuras adjuntas, que deben considerarse de una manera ilustrativa y no limitante, en las que:

La Figura 1 representa gráficamente el curso del método propuesto para ordenar, replicar y registrar operaciones transparentes no repudiables que implican datos personales.

La Figura 2 representa gráficamente las interacciones entre el usuario y los diferentes tipos de nodos y las conexiones entre nodos.

Descripción detallada de realizaciones preferidas

Los libros de contabilidad proporcionados por las diferentes tecnologías de cadenas de bloques ofrecen la posibilidad de mantener una red de nodos conectados que almacenan un historial de registros inmutables, persistentes y potencialmente incensurables. Sin embargo, la inclusión de estos sistemas en el entorno empresarial conlleva obligaciones legislativas que necesitan ser atendidas, especialmente cuando estas interacciones implican el tratamiento de información cubierta por las leyes de protección de datos vigentes. Por lo tanto, la presente invención se beneficia de las ventajas de tener un registro distribuido en términos de inmutabilidad y trazabilidad mientras que mantiene un estricto cumplimiento de estas normas de protección de datos. Para lograr esto, véase la Figura 1, la información personal junto con un identificador que representa la operación anterior realizada por el usuario se ancla usando un resumen criptográfico con dos objetivos: en primer lugar, no anclar la información personal en una cadena de bloques inmutable y, en segundo lugar, proporcionar una prueba para el usuario de que la información no ha sido manipulada y un tercero ha podido llevar a cabo operaciones sin su consentimiento. Además, los datos en sí se almacenan en una base de datos convencional en la que es posible borrar la información cuando se solicite.

La presente invención propone un nuevo esquema de arquitectura que, como se observará en las siguientes secciones, incluye diferentes elementos que harán posible, por un lado, garantizar la existencia de cierta información usando una cadena de bloques y, por otro, borrar los datos personales anclados en un repositorio convencional.

A continuación, se detallan los diferentes procesos que hacen posible la creación de un nodo informático. En primer lugar, se describe parte de la terminología técnica necesaria para comprender el alcance de la invención. En segundo lugar, también se detallan los diferentes elementos que componen el sistema. En tercer lugar, se describe la operación del sellado de eventos. En cuarto lugar, se describe el proceso de verificación de la información anclada. Finalmente, se describe cómo manejar una solicitud para borrar información de la red a nivel técnico.

Terminología técnica:

Los siguientes conceptos se refieren a la terminología técnica que se usará a lo largo de la descripción de la presente invención.

- Dirección. Las direcciones están asociadas a un par de claves criptográficas cuya parte pública se crea siguiendo procedimientos conocidos para generar un identificador único. En tecnologías tales como Bitcoin o Litecoin, los diferentes actores pueden ser destinatarios de diferentes transacciones y operar con las unidades monetarias recibidas usando el componente privado de este par de claves para firmar y llevar a cabo operaciones de salida. Aunque es criptográficamente robusto, el principal impacto de este modelo está vinculado a la importancia de la custodia de las claves privadas ya que, en caso de pérdida, destrucción o robo de las mismas el legítimo propietario perderá el control total de la dirección. Esta realidad representa una importante barrera que algunos servicios intentan sortear ofreciendo sistemas de custodia de claves.

- Billeteras. Las billeteras se entienden como piezas de software (o hardware) que implementan un sistema de gestión de diferentes claves privadas de una o varias criptomonedas. Las billeteras pueden almacenar las claves

en diferentes soportes y con diferentes medidas de seguridad, tanto si pueden estar conectadas a la red ("billeteras calientes") como si no ("billeteras frías").

- Transacción. Unidad transaccional en la que un actor prueba criptográficamente su voluntad de registrar una operación. Estas operaciones pueden incluir la transferencia de unidades monetarias entre dos cuentas u operaciones más complejas que implican cambios de estado dependiendo la cadena o que almacenan una pieza de dato.

- Bloque. Un bloque es una unidad contenedora compuesta por un conjunto de nuevas transacciones a registrar en la cadena. La característica más relevante de esta estructura es que contiene una referencia criptográfica del bloque anterior, lo que permite encadenarlos sucesivamente al bloque génesis original. La más mínima modificación en cualquiera de ellos dará como resultado una cadena completamente diferente.

- Mecanismo de consenso. En las cadenas de bloques públicas, se usa la minería de bloques para minimizar el riesgo de que una transacción no se incluya en un bloque por voluntad del minero. Los diferentes algoritmos existentes implementan diferentes procedimientos tales como prueba de trabajo para acordar qué nodo será el responsable de añadir el siguiente bloque. Estos procesos están definidos de forma que la adición de un nuevo bloque no puede ser conocida a priori y se distribuye de forma homogénea entre los participantes. En el caso de la prueba de trabajo, los nodos participantes llevan a cabo operaciones de generación de función de troceo criptográfica sobre las transacciones pendientes a añadir a la cadena en busca de una función de troceo con condiciones específicas. La naturaleza misma de una operación de función de troceo significa que quien tenga la capacidad informática para llevar a cabo estas operaciones tendrá más posibilidades de encontrar una función de troceo objetivo válida, pero incluso si tiene una mayoría, no puede estar seguro de quién añadirá el siguiente bloque, sino más bien las posibilidades de hacerlo. De esta forma, si la capacidad informática de la red está convenientemente distribuida, la censura o rechazo de una transacción no puede prolongarse indefinidamente en el tiempo. Algunos de los mecanismos de consenso más conocidos que se usan son la "Prueba de trabajo" o la "Prueba de participación".

- Cadena de bloques. Corresponde a un conjunto de bloques encadenados entre sí a partir del bloque de génesis original que da origen a la cadena.

- Cadena de bloques pública. Se trata de una cadena de bloques cuya información es accesible para cualquier persona y en la que cualquiera puede contribuir a su mantenimiento. Ejemplos de cadenas de bloques públicas son Bitcoin, Litecoin o Ethereum.

- Cadenas de bloques permisivas. Se trata de una cadena de bloques (primera cadena de bloques de las reivindicaciones) en la que las operaciones tales como el acceso, la adición de bloques, la creación de transacciones u otras operaciones están autorizadas y controladas por un actor dado. Este tipo de tecnología puede usarse como repositorios compartidos entre diferentes organizaciones o departamentos. Las tecnologías de soporte normalmente se pueden configurar para funcionar como cadenas de bloques públicas. Hyperledger Fabric [10] o Multichain [11] son implementaciones de este tipo de filosofías.

- Datos anclados. El concepto se refiere a los datos insertados en una transacción que se registra en una cadena de bloques. Para evitar el abuso del sistema, diferentes cadenas de bloques establecen diferentes tipos de restricciones, tales como limitar el espacio usable para este propósito. En primer lugar, la longitud de los datos anclados es arbitraria y depende de las restricciones que establezca el protocolo en cada caso. Por ejemplo, en el caso de Bitcoin, el número máximo de bytes que se pueden anclar es de 80 bytes por transacción [12] aunque en otras cadenas de bloques se puede configurar a tamaño a espacios mucho mayores [11]. Por otro lado, las comisiones necesarias para que se añada una transacción en un bloque y se registre en todos los participantes de la cadena sirven como incentivo para que los nodos de la red guarden la cadena (y los datos por extensión) logrando el efecto de hacer que sea costoso para un atacante potencial inundar la red.

- Función de troceo. Una función de troceo o función de resumen es una función que puede usarse para mapear datos de cualquier longitud en un tamaño delimitado. Una buena función de troceo debe ser determinista (es decir, que dados los mismos datos de entrada siempre genera la misma salida), uniforme (es decir, que las posibles entradas se mapeen de la manera más homogénea posible en todo el intervalo de posibles salidas) y no reversible (es decir, que a pesar de ser computable de forma sencilla, el proceso inverso de obtención de la entrada que genera una cierta salida debe ser irreversible en un tiempo real). En este documento se usará SHA-3_512 [13] como función de troceo.

- Sal. Cadena de bits aleatoria usada como una de las entradas en una función de troceo. Su misión es dificultar la realización de ataques de fuerza bruta asegurando que los datos de cada registro de función de troceo sean diferentes aunque la parte útil sea idéntica. Se usan comúnmente para almacenar contraseñas en bases de datos para evitar que dos usuarios que usan la misma contraseña tengan el mismo valor de contraseña con función de

troceo.

- Árbol de Merkle [14]. Se trata de una estructura criptográfica patentada en 1979 por Ralph C. Merkle en la que cada nodo que no es hoja se etiqueta con el resumen criptográfico resultante de la concatenación de los nodos inferiores o nodos hijos. De esta forma, conociendo el nodo raíz, podría comprobarse la posición concreta ocupada por la función de troceo de un objeto en el árbol y la trayectoria seguida hasta el nodo raíz si este objeto se utilizó o no para generar la función de troceo de la raíz. Este método se puede usar para anclar una única función de troceo, manteniendo fuera de la cadena las pruebas necesarias para corroborar la existencia del objeto y reduciendo el volumen de datos que tiene que aparecer en la propia cadena.

- Sistema de almacenamiento convencional. Corresponde a un sistema de almacenamiento convencional en el que la información no está encadenada y que permite realizar las funciones básicas de un sistema de almacenamiento persistente convencional (CRUD, "Crear", "Leer", "Actualizar" y "Borrar").

Arquitectura de sistema:

Puede haber diferentes tipos de funciones dentro de la arquitectura:

- Nodos completos. Un nodo completo está compuesto por tres elementos: el que interactúa con la cadena de bloques, el sistema de almacenamiento de información CRUD convencional y la interfaz responsable de recibir órdenes de anclaje así como de interactuar de forma transparente con la cadena de bloques y el sistema de almacenamiento convencional.

- Nodos maestros o nodos de administración. Están compuestos por los mismos elementos que un nodo completo pero tienen la responsabilidad adicional de gestionar las funciones de envío y recepción de transacciones entre otras tareas de administración. Además, son nodos con capacidad de minar, es decir, añadir nuevos bloques a la cadena. Este nodo puede autorizar la creación de nuevos nodos con permisos para añadir bloques a la cadena para ofrecer una mejor tolerancia a fallos. Obsérvese que estos nodos potenciales con permisos para añadir bloques no tendrán incentivos a nivel de protocolo por la capacidad de cálculo y almacenamiento dadas, por lo que deberán ser respaldados por cualquier organización con un interés legítimo en el sistema.

- Nodos de lectura. Estos nodos tendrán un cliente que se conecta a la cadena de bloques con permisos de conexión, pero no tendrán permisos de escritura. Se pueden usar para proporcionar una API para consultar información almacenada (por ejemplo, exponer una interfaz tal como Multichain Explorer [15]) o para permitir que los usuarios avanzados realicen verificaciones manuales directamente contra la cadena.

La cadena de bloques:

Como se indicó anteriormente, las cadenas de bloques pueden ser públicas o permisivas. En la presente invención, se usa un modelo permisivo en el que la lectura de la información de la cadena es pública pero en el que existen restricciones respecto a quién puede enviar y recibir operaciones o minar nuevos bloques.

Para el establecimiento del proceso de adición de nuevos bloques, se pueden elegir diferentes soluciones tales como PoW o PoS entre otras, ajustando el nivel de diversidad minada (entendido como "diversidad minada" el número de bloques que un nodo tendrá que esperar hasta que uno nuevo pueda minarse nuevamente por sí mismo) de acuerdo con el tamaño estimado de la red. En los nodos de la red propuesta, se deja a discreción de los administradores de la red el mecanismo de consenso a usar.

En este sentido, las cadenas de bloques basadas en Multichain facilitan estas tareas de administración configurando aspectos tales como los siguientes: nodos que se pueden conectar, nodos que pueden enviar o recibir transacciones, nodos que pueden minar nuevos bloques, criterios mínimos aceptables de diversidad, los detalles del sistema de consenso, el espacio de tiempo entre bloques esperado, el tamaño máximo de los bloques, el tamaño máximo permitido en campos de almacenamiento especiales tales como OP_RETURN.

Por estas razones, en una realización, la presente invención usa un nodo Multichain. Este nodo interno puede gestionarse por medio de una interfaz JSON RPC, aprovechando que está basada e inspirada en la interfaz de gestión del mismo tipo disponible para Bitcoin.

El sistema de almacenamiento convencional:

Como se indicó anteriormente, la información personal no se almacenará en la cadena de bloques. En su lugar, se utilizará un sistema de almacenamiento desplegado en paralelo (es decir, separado) de la cadena de bloques. Para permitir una mayor flexibilidad en cuanto a la naturaleza de los datos anclados, se prefiere una base de datos NoSQL tal como MongoDB.

Particularmente, las diferentes operaciones ancladas en el sistema de almacenamiento tendrán una estructura que permitirá el acceso por índice de transacción (txid) y contendrán un campo de datos adicional con el detalle de la información cuya prueba se ha anclado en la cadena de bloques y otro (prueba) con la propia prueba criptográfica.

Interfaz de control del nodo:

Para controlar las operaciones de anclaje y almacenamiento de información, se proporciona una interfaz JSON RPC para su administración y control. Las llamadas API incluidas se definen de la siguiente manera:

- info. Proporciona detalles sobre la versión del nodo ejecutado.
- delete_data_from_tx. Se encarga de eliminar del repositorio convencional los datos asociados a la transacción en cuestión. El resultado corresponderá al número de elementos borrados, 1 si existe.
- get_data_from_tx. Se encarga de recuperar los datos asociados a la transacción en cuestión. El resultado corresponderá al objeto JSON asociado a esa información.
- get_proof_from_tx. Se encarga de recuperar la función de troceo anclada en la transacción en cuestión. El resultado corresponderá a una función de troceo de SHA 3.
- moor_data. Se encarga de anclar los datos en la cadena de bloques. Si el objeto proporcionado es un diccionario, sus claves se ordenarán alfabéticamente. El resultado corresponderá a la transacción en la que se ancla la función de troceo SHA 3 de la información proporcionada y se ancla la función de troceo:

```
{
 "txid":
 "proof": "00112233..ff"
}
```

- lista de moor. Se encarga de anclar una serie de objetos en la cadena de bloques. Requiere que el objeto provisto sea una lista de objetos JSON, cada uno de los cuales será serializado y anclado en un árbol de Merkle. El resultado corresponderá a la transacción en la que se ancle la raíz del árbol de Merkle y la lista serializada de pruebas generadas para cada objeto de la lista en el mismo orden en que se entregaron.

- use_blockchain. Se encarga de solicitar una tarea de consulta directamente contra la cadena de bloques subyacente. Se puede usar para lanzar comandos específicos, tales como asignar nuevos permisos o consultar información de transacción.

Cuando los efectos de una acción deben compartirse con otros nodos dentro de la red, el registro debe compartirse con las partes implicadas para que puedan interpretar apropiadamente su significado. Algunas soluciones tales como Multichain 2.0 [19] implementan la comunicación de registros privados entre dos partes por medio de transacciones en cadena. Sin embargo, este enfoque tiene dos efectos:

- El almacenamiento del propio registro, aunque esté encriptado, en una sola operación de la cadena de bloques lo reproduce en todos los nodos participantes obligándolos a almacenar información encriptada que nunca usarán.

- Si el cifrado de la información anclada se llevara a cabo con una única clave pública, la solicitud de borrado de los datos del sistema por una única parte interesada obligaría a olvidar la clave privada asociada al nodo y anonimizar todos los registros. Sin embargo, esto podría abordarse si el usuario usara claves independientes para encriptar la información de cada usuario.

Debido a esto, esto propone la transmisión de información fuera de la cadena usando sistemas de almacenamiento convencionales de los que es posible retirar la información.

C) El sellado de eventos de usuario:

Como tal, las características permiten el anclaje de información en la cadena de bloques. Sin embargo, para que un usuario tenga garantías en cuanto al sellado y encadenamiento de eventos, es necesario establecer un modelo de datos que garantice la trazabilidad de las acciones que se le atribuyen. El detalle de las decisiones presentadas en este punto es ajeno al proceso de anclar información en una red, pero implica algunas de las decisiones de diseño del sistema detalladas en esta sección.

Para ilustrar la operación del sistema de almacenamiento y los procedimientos de sellado y verificación de operaciones, se usará un registro de modelo de la siguiente manera:

```
{
 "nombre": "John",
```

```
 "apellido": "Doe",
 "operación": {
    "tipo": "pago",
    "valor": 10
}
}
```

Encadenamiento de pruebas de operaciones de usuario:

En las cadenas de bloques convencionales, el seguimiento de las operaciones de un usuario es sencillo, ya que el usuario mantiene su propia clave privada. Usando la dirección asociada, un usuario puede descubrir fácilmente los elementos que afectan su propia dirección explorando la cadena de bloques. Sin embargo, en la cadena de bloques usada por la presente invención, el usuario no tiene sus propias direcciones para evitar el riesgo de identificación indirecta del usuario basándose en patrones asociados con sus hábitos de uso, por ejemplo. Por lo tanto, las transacciones solo se emiten desde una cuenta o cuentas controladas por la organización.

Sin embargo, si las operaciones se guardaron como se definió anteriormente, un tercer actor malicioso podría corromper el espíritu del sistema presentando una secuencia incompleta de operaciones. Por ejemplo, un usuario puede ser informado de que las operaciones t1, t2 y t4 están asociadas a él. A continuación, podría verificar que, de hecho, se han almacenado tal como se presentan en la cadena, pero está expuesto al mantenedor del archivo que tiene otras operaciones (por ejemplo, t3) cuyo contenido no se ha informado. Este escenario abre la puerta a una recuperación selectiva de las pruebas dependiendo de la conveniencia del operador de la cadena, algo que no se desea.

Para proteger al usuario de la presentación no autorizada de operaciones de las que no tiene conocimiento, se introduce el concepto de "últimas acciones encadenadas". Cada operación de un usuario (con la excepción de la primera que contendrá una cadena vacía porque no se puede vincular con ninguna otra) tendrá una propiedad _last_chained_action que almacenará el txid de la operación anterior del usuario. De esta forma, el encadenamiento de cada operación con la anterior garantizará al usuario que ningún tercero ha podido introducir operaciones sin su conocimiento ya que podrá comprobar si cada nueva operación llevada a cabo sobre su persona está encadenada con aquellas de las que ya es consciente.

Como resultado de este encadenamiento, se añadirá al modelo de datos de ejemplo una nueva propiedad denominada _última acción encadenada (obsérvese el carácter _ al comienzo de la propiedad, que indica su carácter especial que contiene esta referencia.

```
{
 "nombre": "John",
 "apellido": "Doe",
 "operación": {
    "tipo": "pago",
    "concepto": "Comprar libro electrónico",
    "valor": 10,0
},
 "_last_chained_action": "112233..ff"
}
```

Para proporcionar más flexibilidad mediante la vinculación de operaciones que implican a varios usuarios, se aceptan dos formatos diferentes para la _última acción encadenada. Por un lado, la cadena de texto que representa directamente el identificador de la transacción anterior. Por otro lado, un diccionario en el que las claves hacen referencia a los identificadores de cada actor implicado mientras que los valores muestran sus transacciones correspondientes.

De esta forma, un ejemplo de implementación de una operación de transferencia de 10 unidades entre el usuario con identificador usuario1 y el usuario con identificador usuario2 podría implementarse de la siguiente manera: "operación":

```
{
 "operación": {
  "tipo": "transferencia",
  "valor": 10.0,
  "de": "usuario1",
  "a": "usuario2"
 },
 _última acción encadenada" {
  "de": "112233..ff",
```

```
    "a": "778899..00"
   }
}
```

Anclaje de datos personales pseudoanonimizados:

Como se mencionó anteriormente, la información relacionada con cada operación no estará directamente anclada en la cadena de bloques. En su lugar, se aplicará un algoritmo de función de troceo a los datos. Dependiendo del volumen de operaciones, existen dos posibilidades: anclaje directo del resumen criptográfico de una operación o anclaje del resumen criptográfico del árbol de Merkle en el que se almacena la información.

Función de troceo directa de los objetos que contienen datos personales:

Aunque las funciones de troceo son unidireccionales, dependiendo de los datos incluidos, existe el riesgo de que el procesador de datos o cualquier persona con acceso a la función de troceo pueda encontrar el registro original que dio lugar a la función de troceo (ataques de preimagen). Por ejemplo, en el modelo propuesto como ejemplo, se asume que el número de nombres y apellidos existentes es grande pero lo suficientemente limitado como para que un actor no autorizado pueda forzarlo por fuerza bruta generando combinaciones de nombres, apellidos y operaciones hasta que se encuentren los datos que generan la función de troceo.

Para solucionar este problema, es necesario añadir una cadena criptográfica (o sal criptográfica) al registro. Con esta sal, la presente invención consigue que si dos personas físicas hubieran generado un registro idéntico porque tienen los mismos datos no generarán el mismo resumen criptográfico. Para satisfacer esta condición, es necesario que la cadena añadida sea un elemento aleatorio conocido que se define, de acuerdo con una realización particular, como una cadena aleatoria de 32 caracteres hexadecimales. Partiendo de estas premisas, se añadirá un nuevo atributo _sal a la información almacenada en la base de datos privada:

```
{
 {"nombre": "John",
 "apellido": "Doe",
 "operación": {
  "tipo": "pago",
  "valor": 10
},
 "_última_acción_encadenada": "112233..ff",
 "_sal": "7a15f5ec6bc44e7ba6c69f792eb31785"
}
```

Dado un registro de estas características, el almacenamiento de este en una colección de "operaciones" del almacenamiento convencional se realizará como el valor de una propiedad denominada "datos", que irá acompañada de una propiedad "txid" que hará referencia a la transacción en la que se encuentra la función de troceo de estos datos y una tercera propiedad denominada "prueba" con la propia prueba criptográfica.

De esta forma, cuando el nodo reciba una solicitud de información sobre la información anclada en un txid especificado, será posible acceder a los datos que le dan origen usando esta referencia. Dado el ejemplo anterior, el objeto final almacenado en el repositorio será de la siguiente manera:

```
{
 {"txid": "112233..ff",
 "datos": {
    "nombre": "John",
    "apellido": "Doe",
    "operación": {
       "tipo": "pago",
       "valor": 10
    },
    "_última acción encadenada": "112233..ff",
    "_sal": "7a15f5ec6bc44e7ba6c69f792eb31785"
 },
  "proof": "ccbbaa...00"
}
```

Obsérvese que, en el caso de objetos complejos tales como diccionarios o listas, estos deben serializarse convenientemente antes de se les aplique función de troceo, prestando especial atención al orden de los pares clave-valor.

Uso de árboles de Merkle para la aplicación de función de troceo de varias operaciones:

Ofrece la posibilidad de usar árboles de Merkle de forma nativa. Para ello, y asumiendo una lista de operaciones formada de acuerdo con las pautas definidas anteriormente, se generará el árbol de Merkle del conjunto de operaciones, anclando el nodo raíz en la primera cadena de bloques y devolviendo el conjunto de pruebas criptográficas junto con el identificador de la transacción en la que se ancló el valor del nodo raíz.

Por tanto, dada una lista de objetos, el almacenamiento de cada uno de ellos se hará en una colección de "operaciones" del repositorio convencional que tiene como valor la propiedad datos, junto con una propiedad txid que hará referencia a la transacción en la que se encontró la función de troceo de los datos y una propiedad de prueba que contiene los detalles de la prueba correspondiente.

La principal consecuencia de esta práctica es la reducción del número de transacciones llevadas a cabo la cadena de bloques. Por otro lado, cabe señalar que el proceso de verificación requiere más información complementaria que necesita almacenarse y un proceso computacional más costoso ya que implica el cálculo de un mayor número de funciones de troceo.

Operaciones de anclaje en redes públicas:

Registrar operaciones en su propia cadena de bloques es una primera etapa en la dirección correcta hacia una relación más transparente con otras partes interesadas. Sin embargo, no resuelve en absoluto el problema, teniendo en cuenta que el procesador podría conservar varias copias de cadenas de bloques alternativas en las que dicha operación nunca se produjo o en las que se retardó su incorporación. Cuanto mayor es el compromiso público de la organización con el estado de una cadena, menor es el riesgo de corrupción. Por lo general, esto se puede hacer aumentando el número de participantes en la red con acceso al registro de operaciones para que sea más fácil para un usuario verificar el estado de la red con otros nodos. Dado que la cadena no contiene información personal, este compromiso podría hacerse público permitiendo el acceso de terceros a la cadena.

Para aumentar la transparencia del sistema, el nodo informático de controlador de datos también puede optar por anclar una prueba de la inclusión de un bloque particular en una cadena de bloques alternativa. Por ejemplo, diariamente, cada n bloques o al alcanzar una altura de bloque divisible por 1000, la función de troceo de ese bloque estaría anclada en una transacción de Bitcoin o Ethereum.

Aunque la publicación de esta operación en otra cadena de bloques alternativa conlleva un pequeño gasto, la ejecución de esta operación permitirá a un observador externo consolidar el compromiso del responsable del tratamiento y su responsable con la cadena vigente. La confirmación vendrá entonces dada por el hecho de que la parte interesada tendrá la consolidación pública de un bloque posterior a aquel en el que se incluyó su propia transacción, siendo imposible generar la publicación sin la participación de la parte interesada para confirmar.

El proceso de verificación:

Los usuarios tendrán la capacidad de almacenar un duplicado de su información personal en sus propios dispositivos. El controlador de datos será el encargado de custodiar estas operaciones con el fin de atenderlas cuando el objeto de los datos lo solicite, por ejemplo, conectándose desde diferentes dispositivos una vez debidamente identificado.

1. Se comprueba que la cadena de operaciones a verificar sea consistente con el resto de operaciones de las que se tiene constancia, confirmando si es posible llevar a cabo el seguimiento a la primera operación de la que se tiene constancia.

2. La parte interesada extrae el txid asociado a la operación a verificar.

3. La parte interesada solicita de un nodo de la red el detalle asociado a la transacción txid de la cadena de bloques.

4. La parte interesada procede a extraer la información anclada en el campo OP_RETURN. Este contenido corresponde a una función de troceo SHA-3.

5. La parte interesada procederá a calcular la prueba correspondiente. Puede haber dos circunstancias:

   - ○ Calcular la función de troceo SHA-3 de la operación a verificar almacenada localmente.

   - ○ Verificar la integridad de la prueba de árbol de Merkle.

6. Se compara la correspondencia de ambos resúmenes: el anclado en la cadena y el generado por la parte

interesada.

7. Si la información se consolida en una cadena de bloques externa verificable, se comprueba si el bloque en el que se contabiliza la transacción txid ya ha sido confirmado en una cadena de bloques pública independiente. Esta última etapa será necesaria siempre que no se pueda proporcionar un acceso completo a la cadena. Si, por ejemplo, la transacción se registra en el bloque 104, sería necesario esperar hasta que se consolide externamente un bloque posterior (por ejemplo, 110). Mientras tanto, la transacción debe marcarse como no confirmada. En cualquier caso, la ventana de explotación de una situación de este tipo se reduce en el tiempo (como máximo, hasta que se confirme el bloque correspondiente en una cadena de bloques pública) y, en todo caso, la manipulación podría ser advertida posteriormente por la parte interesada.

Borrado de datos personales:

Si un usuario solicita el borrado de datos personales, el controlador debe actuar de la siguiente manera:

1. Generará una transacción convencional en la que se registra la solicitud de borrado de la información de un determinado usuario.

2. Una vez anclada la prueba de la transacción, compartirá la solicitud con el resto de nodos participantes

3. Archivar la operación en un repositorio independiente de las otras operaciones para los propósitos del registro y archivo de la solicitud bajo las excepciones expuestas en el artículo 17.3 párrafo e) que se refiere al almacenamiento para la formulación, ejercicio o defensa de reclamaciones. Esta operación deberá borrarse en los periodos establecidos por el responsable del servicio. Bajo ninguna circunstancia se usará esta información para ningún otro fin.

4. Borrará toda la información relacionada con el usuario presente en los repositorios privados de cada nodo.

Con el borrado de la información adicional necesaria, los resúmenes criptográficos anclados en la cadena de bloques pasarán de ser seudonimizados a anonimizados ya que los controladores ya no tendrán la información adicional necesaria para la identificación de las personas físicas detrás de cada registro.

El número de operaciones que pueden procesarse por las diferentes tecnologías de cadena de bloques actualmente es limitado. Tecnologías como Multichain [11] o Hyperledger Fabric [10] alcanzan niveles de 1000 y 3500 operaciones por segundo, lo que podría ser un factor limitante a la hora de anclar resúmenes criptográficos individuales para cada operación.

Sin embargo, la prueba de existencia también puede representar evidencias de un número indefinido de operaciones usando estructuras criptográficas tales como los árboles de Merkle [14] en las que una única función de troceo puede representar varias operaciones. Específicamente, los árboles de Merkle son una estructura criptográfica patentada en 1979 por Ralph C. Merkle en la que cada nodo que no es hoja se etiqueta con el resumen criptográfico resultante de la concatenación de los nodos inferiores o nodos hijos.

De esta forma, conociendo el nodo raíz o nodo raíz, podría comprobarse la posición concreta ocupada por la función de troceo de un objeto en el árbol y la trayectoria seguida hasta el nodo raíz si este objeto se utilizó o no para generar la función de troceo de la raíz. Este método puede usarse para anclar una única función de troceo, manteniendo fuera de la cadena las pruebas necesarias para corroborar la existencia del objeto y reduciendo el volumen de datos que aparecen en la cadena, por un lado, y reduciendo el número de operaciones ancladas, por el otro.

Los expertos en la materia reconocerán que las presentes enseñanzas son susceptibles de una diversidad de modificaciones y/o mejoras. Por ejemplo, aunque la implementación de diversos componentes descritos en el presente documento puede realizarse en un dispositivo de hardware, pueden implementarse también como una solución únicamente de software- por ejemplo, una instalación en un servidor existente. Además, la presente invención puede implementarse como un firmware, combinación de firmware/software, combinación de firmware/hardware, o una combinación de hardware/firmware/software.

El alcance de la presente invención se define en el siguiente conjunto de las reivindicaciones.

Referencias

[1] Nakamoto, Satoshi (2008). «Bitcoin: A Peer-to-Peer Electronic Cash System». Disponible en línea en: https://bitcoin.org/bitcoin.pdf [26/03/2019].

[2] Poon, Joseph y Dryja, Thaddeus (2016). «The Bitcoin Lightning Network: Scalable OffChain Instant Payments».

Disponible en línea en: https://lightning.network/lightning-network-paper.pdf [26/03/2019].

[3] Namecoin Developers (2018). «Namecoin FAQ». Disponible en línea en at: https://namecoin.org/ [26/03/2019].

[4] Wood, Gavin Dr. (2015). «Ethereum: A Secure Decentralised Generalised Transaction Ledger |EIP-150 Revision». Disponible en línea en: http://gavwood.com/Paper.pdf [26/03/2019].

[5] Alonso, Kurt M. y Koe (2018). «Zero To Monero: First Edition | A Tehcnical Guide to a Private Digital Currency; for Beginners, Amateurs, and Experts». Disponible en línea en: https://getmonero.org/library/Zero-to-Monero-1-0-0.pdf [26/03/2019].

[6] Coinmarketcap (2018). «Top 100 Cryptocurrencies by Market Capitalizations Available online at: https://coin-marketcap.com [26/03/2019].

[7] Parlamento Europeo y Consejo de la Unión Europea (2016). «Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo, de 27 de abril de 2016 relativo a la protección de las personas físicas en lo que respecta al tratamiento de dates personales y a la libre circulación de estos datos y por el que se deroga la Directiva 95/46/CEs Diario Oficial de la Unión Europea. Disponible en línea en: https://eur-lex.europa.eu/legal-content/ES/TXT/HTML/?uri=CELEX:32016R0679&from=EN [26/03/2019].

[8] KHI, Ira A. y Gorelik, Aleksandr (2019). «Method and system for managing personal information within independent computer systems and digital networkss European Patent Office EP3440823A1. Disponible en línea en: https://patents.google.com/patent/EP3440823A1 [26/03/2019].

[9] Hopwoord, Daira; Bowe, Sean; Hornby, Taylor and Wilcox, Nathan (2018). «Zcash Protocol Specification: Version 2018.0-beta-30». Disponible en línea en: https://github.com/zcash/zips/blob/master/protocol/protocol.pdf [26/03/2019].

[10] The Linux Foundation(2017). «Hyperledger Fabrics Available online at: https://github.com/dash-pay/dash/wiki/Whitepaper [26/03/2019].

[11] Greenspan, Dr. Gideon (2015). «MultiChain Private Blockchain — White Papers disponible en línea en: https://www.multichain.com/download/MultiChain-White-Paper.pdf [26/03/2019].

[12] Bitcoin Developers (2015). «OP_RETURN». Disponible en línea en: https://en.bitcoin.it/wiki/OP_RETURN [26/03/2019].

[13] Dang, Quynh H. (2015). «Secure Hash Standard (SHS)». Federal Information Processing Standards Publication, FIPS PUB 180-4. Disponible en línea en: https://www.nist.gov/publications/secure-hash-standard?pub_id=919060 [26/03/2019].

[14] Merkle, Ralph C. (1979). «Method of providing digital signatures». Patente de Estados Unidos US4309569A, presentada: 5 de septiembre de 1979. Disponible en línea en: https://patents.google.com/patent/US4309569A/en?oq=US4309569A [26/03/2019].

[15] Multichain Developers (2015). «Multichain Explorer: Web-based explorer for MultiChain blockchains». Disponible en línea en: https://github.com/MultiChain/multichain-explorer [26/03/2019].

REIVINDICACIONES

1. Un método para ordenar, replicar y registrar de manera transparente y no repudiable operaciones que implican datos personales, comprendiendo el método:

realizar, por un dispositivo informático, una operación que usa un nodo informático de controlador de datos, teniendo dicho nodo informático de controlador de datos autorización de escritura en una primera cadena de bloques, siendo la primera cadena de bloques una cadena de bloques permisiva; y
- almacenar, por el nodo informático de controlador de datos, información personal del usuario en un sistema de almacenamiento separado de la primera cadena de bloques;

caracterizado porque el método comprende, además:

anclar, por el nodo informático de controlador de datos, una prueba criptográfica de dicha operación en dicha primera cadena de bloques usando un algoritmo de función de troceo que encadena la operación a una operación anterior realizada por dicho usuario, estando anclada la prueba criptográfica junto con una cadena criptográfica, siendo la última una cadena aleatoria de una cierta longitud.

2. El método de la reivindicación 1, en donde la cadena aleatoria tiene una longitud de 32 caracteres hexadecimales.

3. El método de la reivindicación 1, en donde la información personal del usuario se almacena en el sistema de almacenamiento separado usando un mecanismo de índice de operaciones.

4. El método de la reivindicación 3, en donde la información personal del usuario se almacena con un primer campo de datos que detalla la información cuya prueba se ha anclado en la primera cadena de bloques y un segundo campo de datos que incluye la prueba criptográfica.

5. El método de la reivindicación 1, en donde el algoritmo de función de troceo comprende un árbol de Merkle.

6. El método de la reivindicación 1, que comprende además el anclaje, por el nodo informático de controlador de datos, de una prueba de dicho anclaje de la prueba criptográfica en una segunda cadena de bloques.

7. Un sistema para ordenar, replicar y registrar de manera transparente y no repudiable operaciones que implican datos personales, que comprende:

- un dispositivo informático de un usuario;
- un nodo informático de controlador de datos;
- una primera cadena de bloques; y
- un sistema de almacenamiento separado de dicha primera cadena de bloques;

en donde el nodo informático de controlador de datos tiene autorización de escritura en la primera cadena de bloques, siendo esta última una cadena de bloques permisiva; y en donde el nodo informático de controlador de datos está configurado para recibir una operación del usuario y almacenar información personal del usuario en dicho sistema de almacenamiento,
caracterizado porque el nodo informático de controlador de datos está configurado además para:

- anclar una prueba criptográfica de dicha operación recibida en la primera cadena de bloques usando un algoritmo de función de troceo que encadena la operación a una operación anterior realizada por el usuario, anclándose la prueba criptográfica junto con una cadena criptográfica, siendo la última una cadena aleatoria de una cierta longitud.

8. El sistema de la reivindicación 7, en donde la cadena aleatoria tiene una longitud de 32 caracteres hexadecimales.

9. El sistema de la reivindicación 7, en donde el algoritmo de función de troceo comprende un árbol de Merkle.

10. El sistema de la reivindicación 7, que comprende además una segunda cadena de bloques, estando configurado además el nodo informático de controlador de datos para anclar una prueba de dicho anclaje de la prueba criptográfica en la segunda cadena de bloques.

11. Un medio legible por ordenador no transitorio que comprende instrucciones de código que cuando se ejecutan por al menos un procesador de un sistema informático implementan el método de la reivindicación 1.

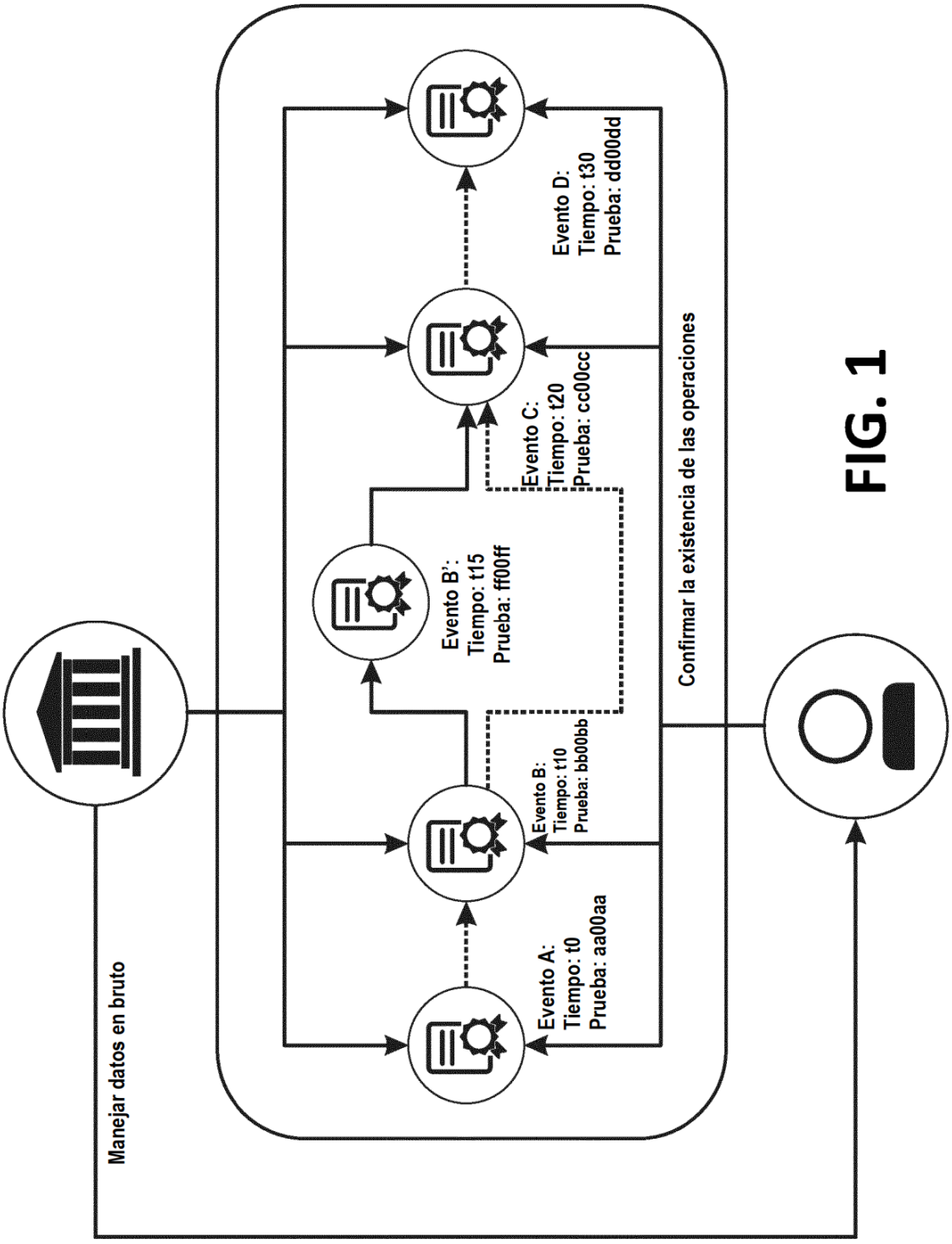
Manejar datos en bruto
Evento A:
Tiempo: t0
Prueba: aa00aa
Evento B:
Tiempo: t10
Prueba: bb00bb
Evento B':
Tiempo: t15
Prueba: ff00ff
Evento C:
Tiempo: t20
Prueba: cc00cc
Evento D:
Tiempo: t30
Prueba: dd00dd
Confirmar la existencia de las operaciones

FIG. 1

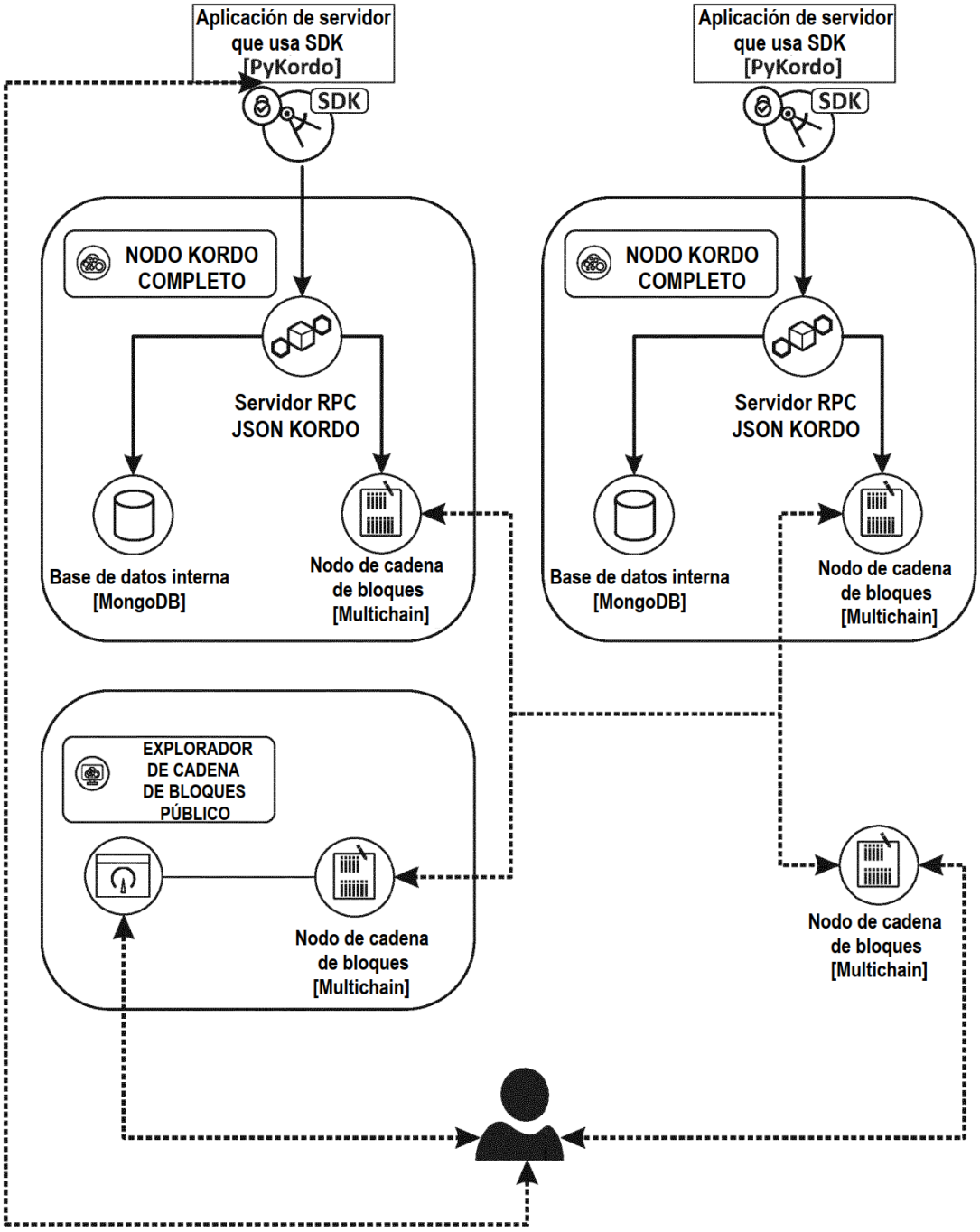
Aplicación de servidor que usa SDK [PyKordo]
SDK
NODO KORDO COMPLETO
Servidor RPC JSON KORDO
Base de datos interna [MongoDB]
Nodo de cadena de bloques [Multichain]
Aplicación de servidor que usa SDK [PyKordo]
SDK
NODO KORDO COMPLETO
Servidor RPC JSON KORDO
Base de datos interna [MongoDB]
Nodo de cadena de bloques [Multichain]
EXPLORADOR DE CADENA DE BLOQUES PÚBLICO
Nodo de cadena de bloques [Multichain]
Nodo de cadena de bloques [Multichain]

FIG. 2