(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 811 979**

(51) Int. CI.:

H04L 29/08 (2006.01)
H04L 29/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **12.12.2013 PCT/US2013/074509**

(87) Fecha y número de publicación internacional: **19.06.2014 WO14093561**

(96) Fecha de presentación y número de la solicitud europea: **12.12.2013 E 13818561 (6)**

(97) Fecha y número de publicación de la concesión europea: **05.08.2020 EP 2932690**

(54) Título: **Descarga de copias para proveedores de descarga dispares**

(30) Prioridad:

12.12.2012 US 201213711637

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
15.03.2021

(73) Titular/es:

MICROSOFT TECHNOLOGY LICENSING, LLC (100.0%)
One Microsoft Way
Redmond, WA 98052 , US

(72) Inventor/es:

GREEN, DUSTIN L.

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Descarga de copias para proveedores de descarga dispares

Antecedentes

Un mecanismo para transferir datos es leer los datos de un archivo de una ubicación de origen a una memoria principal y escribir los datos desde la memoria principal a una ubicación de destino. Si bien en algunos entornos, esto puede funcionar aceptablemente para datos relativamente pequeños, conforme los datos aumentan, el tiempo que se tarda en leer los datos y trasferir los datos a otra ubicación aumenta. Adicionalmente, si a los datos se accede a través de una red, la red puede imponer retrasos adicionales en la trasmisión de los datos desde la ubicación de origen a la ubicación de destino. Es más, cuestiones de seguridad combinadas con la complejidad de disposiciones de almacenamiento pueden complicar la trasferencia de datos.

La materia de asunto reivindicada en esta memoria no se limita a realizaciones que resuelven desventajas o que funcionan únicamente en entornos tales como los descritos anteriormente. En cambio, estos antecedentes únicamente se proporcionan para ilustrar un área tecnológica ejemplar donde se pueden poner en práctica algunas realizaciones descritas en esta memoria. El documento US 2012/0079583 A1 describe aspectos relacionados con lecturas y escrituras de descarga. Un solicitante que busca trasferir datos envía una solicitud de una representación de los datos. En respuesta, el solicitante recibe uno o más tokens que representan unos datos. El solicitante puede entonces proporcionar uno o más de estos tokens a un componente con una solicitud para escribir datos representados por la uno o más tokens. La patente europea EP 2 262 164 A1 describe un método para asegurar la trasferencia de datos entre entidades, que comprende: establecer un primer canal seguro entre una primera entidad que tiene al menos una primera credencial y una segunda entidad que tiene al menos una segunda credencial, establecer un segundo canal seguro entre dicha primera entidad y una tercera entidad, siendo dicha tercera entidad de confianza para dicha primera entidad, a través de dicho segundo canal seguro entre dicha primera entidad y dicha tercera entidad.

Compendio

La invención definida en las reivindicaciones 1, 3 y 4 independientes está relacionada con tecnología de descarga. En aspectos, un proveedor de descarga de origen puede trasferir datos masivos a un proveedor de descarga de destino incluso si los proveedores de descarga son diferentes e independientes entre sí y no tienen conocimiento previo entre sí. En preparación para transferir datos masivos, la confianza se puede extender a los proveedores de descarga. Tras la autenticación, los proveedores de descarga pueden trasferir todos o una parte de los datos masivos a través de un canal seguro sin que los datos atraviesen el iniciador de la trasferencia.

Este Compendio se proporciona para identificar brevemente algunos aspectos de la materia de asunto que se describe adicionalmente en la Descripción Detallada más adelante. Este Compendio no pretende identificar rasgos clave o esenciales de la materia de asunto reivindicada, ni se pretende que sea usado para limitar el alcance de la materia de asunto reivindicada.

La frase “materia de asunto descrita en esta memoria” se refiere a materia de asunto descrita en la Descripción Detallada a menos que el contexto lo indique claramente de otro modo. El término “aspectos” se debe entender como “al menos un aspecto”. Identificar aspectos de la materia de asunto descrita en la Descripción Detallada no pretende identificar rasgos clave o esenciales de la materia de asunto reivindicada.

Los aspectos descritos anteriormente y otros aspectos de la materia de asunto descrita en esta memoria se ilustran a modo de ejemplo y no limitados en las figuras adjuntas en las que numerales de referencia semejantes indican elementos similares y en las que:

Breve descripción de los dibujos

La figura 1 es un diagrama de bloques que representa un entorno informático ejemplar de finalidad general en el que se pueden incorporar aspectos de la materia de asunto descrita en esta memoria;

las figuras 2-4 son diagramas de bloques que representan disposiciones ejemplares de componentes de sistemas en los que pueden funcionar aspectos de la materia de asunto descrita en esta memoria;

la figura 5 es un diagrama de bloques que ilustra proveedores de descarga dispares según aspectos de la materia de asunto descrita en esta memoria;

la figura 6 es un diagrama de flujo que representa generalmente acciones ejemplares que pueden ocurrir en un iniciador según aspectos de la materia de asunto descrita en esta memoria;

la figura 7 es un diagrama de flujo que representa generalmente acciones ejemplares que pueden ocurrir en un destino según aspectos de la materia de asunto descrita en esta memoria; y

la figura 8 es un diagrama de flujo que representa generalmente acciones ejemplares que pueden ocurrir en un origen

según aspectos de la materia de asunto descrita en esta memoria.

Descripción detallada

Definiciones

La frase “materia de asunto descrita en esta memoria” se refiere a materia de asunto descrita en la Descripción Detallada a menos que el contexto lo indique claramente de otro modo. El término “aspectos” se debe entender como “al menos un aspecto”. Identificar aspectos de la materia de asunto descrita en la Descripción Detallada no pretende identificar rasgos clave o esenciales de la materia de asunto reivindicada.

Como se emplea en esta memoria, el término “incluye” y sus variantes se tienen que entender como términos amplios que significan “incluye, aunque sin limitarse a esto”. El término “o” se tiene que entender como “y/o” a menos que el contexto lo indique claramente de otro modo. El término “basado en” se tiene que entender como “basado al menos en parte en”. El término “una realización” se tiene que entender como “al menos un realización”. El término “otra realización” se tiene que entender como “al menos otra realización”.

Como se emplea en esta memoria, términos tales como “un”, “una”, “el” y “la” son inclusivos de uno o más del elemento o acción indicados. Por ejemplo, en las reivindicaciones una referencia a un elemento generalmente significa que al menos un elemento de este tipo está presente y una referencia a una acción significa que se realiza al menos un caso de la acción.

A veces en esta memoria se pueden usar los términos “primero”, “segundo”, “tercero” etc. Sin contexto adicional, el uso de estos términos en las reivindicaciones no pretende implicar una ordenación sino que en cambio se usan para motivos de identificación. Por ejemplo, las frases “primera versión” y “segunda versión” no significan necesariamente que la primera versión es una primerísima versión o que fue creada antes de la segunda versión o incluso que la primera versión se solicita o se utiliza antes que la segunda versión. En cambio, estas frases se usan para identificar versiones diferentes.

Los encabezados son por conveniencia únicamente; información sobre un tema dado se puede encontrar fuera de la sección cuyo encabezado indica ese tema.

A continuación se pueden incluir otras definiciones, explícitas e implícitas.

Entorno de funcionamiento ejemplar

La figura 1 ilustra un ejemplo de un entorno adecuado de sistema informático 100 en el que se pueden implementar aspectos de la materia de asunto descrita en esta memoria. El entorno de sistema informático 100 es únicamente un ejemplo de un entorno informático adecuado y no pretende sugerir ninguna limitación en cuanto al alcance de uso o funcionalidad de aspectos de la materia de asunto descrita en esta memoria. El entorno informático 100 no debe ser interpretado como que tiene dependencia o requisito relacionado con uno cualquiera o una combinación de componentes ilustrados en el entorno de funcionamiento 100 ejemplar.

Aspectos de la materia de asunto descrita en esta memoria son operacionales con otros numerosos entornos o configuraciones de sistema informático de finalidad general o finalidad especial. Ejemplos de sistemas, entornos o configuraciones informáticos muy conocidos que pueden ser adecuados para usar con aspectos de la materia de asunto descrita en esta memoria comprenden ordenadores personales, ordenadores servidores — ya sea físicos o como máquinas virtuales —, dispositivos informáticos de mano o portátiles, sistemas multiprocesador, sistemas basados en microcontrolador, descodificadores, electrónica de consumo programable y no programable, PC de red, miniordenadores, ordenadores mainframe, ayudantes digitales personales (PDA), dispositivos de juegos, impresoras, aparatos que incluyen descodificadores, centro multimedia, u otros aparatos, dispositivos informáticos integrados o conectados a automóviles, otros dispositivos móviles, dispositivos de telefonía incluidos teléfonos móviles, teléfonos inalámbricos y teléfonos cableados, entornos informáticos distribuidos que incluyen cualquiera de los sistemas o dispositivos anteriores, y similares.

Aspectos de la materia de asunto descrita en esta memoria se pueden describir en el contexto general de instrucciones ejecutables por ordenador, tales como módulos de programa, que son ejecutados por un ordenador. Generalmente, los módulos de programa incluyen rutinas, programas, objetos, componentes, estructuras de datos, etc., que realizan tareas particulares o implementan tipos de datos abstractos particulares. Aspectos de la materia de asunto descrita en esta memoria también se pueden poner en práctica en entornos informáticos distribuidos en los que dispositivos de procesamiento a distancia realizan tareas que se vinculan a través de una red de comunicaciones. En un entorno informático distribuido, se pueden ubicar módulos de programa en medios de almacenamiento informático tanto locales como a distancia que incluyen dispositivos de almacenamiento en memoria.

Como alternativa o adicionalmente, la funcionalidad descrita en esta memoria puede ser realizada, al menos en parte, por uno o más componentes lógicos de hardware. Por ejemplo, y sin limitación, tipos ilustrativos de componentes lógicos de hardware que se pueden usar incluyen matrices de puertas programables en campo (FPGA), Circuitos Integrados Específicos de Programa (ASIC), Productos Estándar Específicos de Programa (ASSP), sistemas de

Sistema-en-un-chip (SOC), Dispositivos Lógicos Programables Complejos (CPLD), y similares.

Con referencia a la figura 1, un sistema ejemplar para implementar aspectos de la materia de asunto descrita en esta memoria incluye un dispositivo informático de finalidad general en forma de ordenador 110. Un ordenador puede incluir cualquier dispositivo electrónico que sea capaz de ejecutar una instrucción. Componentes del ordenador 110 pueden incluir una unidad de procesamiento 120, una memoria de sistema 130 y un bus de sistema 121 que acopla diversos componentes de sistema que incluye la memoria de sistema a la unidad de procesamiento 120. El bus de sistema 121 puede ser cualquiera de varios tipos de estructuras de bus que incluye un bus de memoria o controlador de memoria, un bus periférico, y un bus local usando cualquiera de una variedad de arquitecturas de bus. A modo de ejemplo, y no limitación, tales arquitecturas incluyen bus de Arquitectura Estándar de la Industria (ISA), bus de Arquitectura MicroCanal (MCA), bus ISA Mejorado (EISA), bus local de la Asociación de Estándares Electrónicos de Vídeo (VESA), bus de Interconexión de Componente Periférico (PCI) también conocido como bus Mezzanine, bus de Interconexión de Componente Periférico Extendido (PCI-X), Puerto Avanzado de Gráficos (AGP), y PCI express (PCIe).

La unidad de procesamiento 120 se puede conectar a un dispositivo de seguridad de hardware 122. El dispositivo de seguridad 122 puede almacenar y ser capaz de generar claves criptográficas que se pueden usar para asegurar diversos aspectos del ordenador 110. En una realización, el dispositivo de seguridad 122 puede comprender un chip de Módulo de Plataforma de Confianza (TPM), Dispositivo de Seguridad TPM, o algo semejante.

El ordenador 110 típicamente incluye una variedad de medios legibles por ordenador. Medios legibles por ordenador pueden ser cualesquiera medios disponibles a los que puede acceder el ordenador 110 e incluye medios tanto volátiles como no volátiles, y medios retirables y no retirables. A modo de ejemplo, y no limitación, medios legibles por ordenador pueden comprender medios de almacenamiento informático y medios de comunicación.

Medios de almacenamiento informático incluyen tanto medios volátiles como no volátiles, retirables y no retirables, implementados en cualquier método o tecnología para almacenamiento de información tal como instrucciones legibles por ordenador, estructuras de datos, módulos de programa, u otros datos. Medios de almacenamiento informático incluyen RAM, ROM, EEPROM, almacenamiento de estado sólido, memoria flash u otra tecnología de memoria, CD-ROM, discos digitales versátiles (DVD) u otro almacenamiento de disco óptico, cartuchos magnéticos, cinta magnética, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, o cualquier otro medio que se pueda usar para almacenar la información deseada y a la que pueda acceder el ordenador 110. Medios de almacenamiento informático no incluyen medios de comunicación.

Medios de comunicación típicamente incorporan instrucciones legibles por ordenador, estructuras de datos, módulos de programa u otros datos en una señal de datos modulada tal como una onda portadora u otro mecanismo de transporte e incluyen cualesquiera medios de entrega de información. El término “señal de datos modulada” significa una señal que tiene una o más de sus características establecidas o cambiadas de tal manera como para codificar información en la señal. A modo de ejemplo, y no limitación, medios de comunicación incluyen medios cableados tales como una red cableada o conexión cableada directa, y medios inalámbricos tales como medios acústicos, RF, infrarrojos y otros inalámbricos. Combinaciones de cualquiera de los anteriores también se deben incluir dentro del alcance de medios legibles por ordenador.

La memoria de sistema 130 incluye medios de almacenamiento informático en forma de memoria volátil y/o no volátil tal como memoria de solo lectura (ROM) 131 y memoria de acceso aleatorio (RAM) 132. Un sistema básico de entrada/salida 133 (BIOS), que contiene las rutinas básicas que ayudan a trasferir información entre elementos dentro del ordenador 110, tal como durante el arranque, se almacena típicamente en ROM 131. La RAM 132 típicamente contiene datos y/o módulos de programa que son inmediatamente accesibles y/o actualmente son operados por la unidad de procesamiento 120. A modo de ejemplo, y no limitación, la figura 1 ilustra el sistema operativo 134, programas de aplicación 135, otros módulos de programa 136, y datos de programa 137.

El ordenador 110 también puede incluir otros medios de almacenamiento informático retirables/no retirables, volátiles/no volátiles. A modo de ejemplo únicamente, la figura 1 ilustra una unidad de disco duro 141 que lee o escribe en medios magnéticos no retirables no volátiles, un disco duro magnético 151 que lee o escribe en un disco magnético retirable no volátil 152, y un disco duro óptico 155 que lee o escribe en un disco óptico retirable no volátil 156 tal como un CD ROM, DVD, u otros medios ópticos. Otros medios de almacenamiento informático retirables/no retirables, volátiles/no volátiles, que se pueden usar en el entorno de funcionamiento ejemplar incluyen cartuchos de cinta magnética, tarjetas de memoria flash y otros dispositivos de almacenamiento de estado sólido, discos versátiles digitales, otros discos ópticos, cinta de vídeo digital, RAM de estado sólido, ROM de estado sólido y similares. La unidad de disco duro 141 se puede conectar al bus de sistema 121 a través de la interfaz 140, y la unidad de disco magnético 151 y la unidad de disco óptico 155 se puede conectar al bus de sistema 121 mediante una interfaz para memoria retirable no volátil tal como la interfaz 150.

Las unidades y sus medios de almacenamiento informático asociados, tratados anteriormente e ilustrados en la figura 1, proporcionan almacenamiento de instrucciones legibles por ordenador, estructuras de datos, módulos de programa y otros datos para el ordenador 110. En la figura 1, por ejemplo, la unidad de disco duro 141 se ilustra como que almacena el sistema operativo 144, programas de aplicación 145, otros módulos de programa 146, y datos de programa 147. Obsérvese que estos componentes pueden ser iguales o diferentes al sistema operativo 134,

programas de aplicación 135, otros módulos de programa 136, y datos de programa 137. El sistema operativo 144, programas de aplicación 145, otros módulos de programa 146, y datos de programa 147 reciben diferentes números en esta memoria para ilustrar que, como mínimo, son copias diferentes.

Un usuario puede introducir órdenes e información en el ordenador 110 a través de dispositivos de entrada tales como un teclado 162 y un dispositivo señalizador 161, comúnmente se les hace referencia como ratón, ratón de bola o panel táctil. Otros dispositivos de entrada (no se muestra) pueden incluir un micrófono (p. ej., para introducir voz u otro audio), joystick, mando de juegos, antena parabólica, escáner, una pantalla sensible al tacto, una tableta de escritura, una cámara (p. ej., para introducir gestos u otra entrada visual), o algo semejante. Estos y otros dispositivos de entrada a menudo se conectan a la unidad de procesamiento 120 a través de una interfaz de aporte de usuario 160 que se acopla al bus de sistema, pero se puede conectar mediante otra interfaz y estructuras de bus, tales como un puerto paralelo, puerto de juegos o un bus serie universal (USB).

Mediante el uso de uno o más de los dispositivos de entrada identificados anteriormente se puede establecer una interfaz de Usuario Natural (NUI). Una NUI, puede depender de reconocimiento de voz, tacto y reconocimiento de puntero, reconocimiento de gestos tanto en pantalla como adyacente a la pantalla, gestos en el aire, seguimiento de cabeza y ojos, voz y habla, visión, tacto, gestos, inteligencia de máquina, y similares. Alguna tecnología ejemplar de NUI que se puede emplear para interactuar con un usuario incluye pantallas sensibles al tacto, reconocimiento de voz y habla, entendimiento de intención y meta, detección de gestos de movimiento usando cámaras de profundidad (tales como sistemas de cámaras estereoscópicas, sistemas de cámaras de infrarrojos, sistemas de cámaras RGB, y combinaciones de los mismos), detección de gestos de movimiento usando acelerómetros/giroscopios, reconocimiento facial, pantallas 3D, seguimiento de cabeza, ojos y mirada, realidad aumentada inmersiva y sistemas de realidad virtual, así como tecnologías para sentir la actividad cerebral usando electrodos sensitivos a campo eléctrico (métodos EEG y relacionados).

Un monitor 191 u otro tipo de dispositivo de exposición también se conecta al bus de sistema 121 por medio de una interfaz, tal como una interfaz de vídeo 190. Además del monitor, los ordenadores también pueden incluir otros dispositivos de salida periféricos tales como altavoces 197 e impresora 196, que se pueden conectar a través de una interfaz periférica de salida 195.

El ordenador 110 puede funcionar en un entorno de red usando conexiones lógicas a uno o más ordenadores remotos, tales como un ordenador remoto 180. El ordenador remoto 180 puede ser un ordenador personal, un servidor, un rúter, un PC en red, un dispositivo parejo u otro nodo de red común, y típicamente incluye muchos o todos los elementos descritos anteriormente respecto al ordenador 110, en la figura 1 aunque únicamente se ha ilustrado un dispositivo de almacenamiento de memoria 181. Las conexiones lógicas representadas en la figura 1 incluyen una red de área local (LAN) 171 y una red de área amplia (WAN) 173, pero también puede incluir redes telefónicas, redes de campo cercano y otras redes. Tales entornos de red son corriente común en oficinas, redes informáticas por toda la empresa, intranets, e internet.

Cuando se usa en un entorno de red LAN, el ordenador 110 se conecta a la LAN 171 a través de una interfaz de red o adaptador 170. Cuando se usa en un entorno de red WAN, el ordenador 110 puede incluir un módem 172 u otros medios para establecer comunicaciones por la WAN 173, tal como internet. El módem 172, que puede ser interno o externo, se puede conectar al bus de sistema 121 por medio de la interfaz de aporte de usuario 160 u otro mecanismo apropiado. En un entorno en red, módulos de programa representados respecto al ordenador 110, o partes del mismo, pueden ser almacenados en el dispositivo de almacenamiento de memoria remoto. A modo de ejemplo, y no limitación, la figura 1 ilustra programas de aplicación remotos 185 como residentes en el dispositivo de memoria 181. Se apreciará que las conexiones de red mostradas son ejemplares y se pueden usar otros medios para establecer un enlace de comunicaciones entre los ordenadores.

Lecturas y escrituras de descarga

Como se ha mencionado previamente, algunas operaciones tradicionales de trasferencia de datos pueden no ser eficientes o incluso trabajar en entornos de almacenamiento actuales.

Las figuras 2-5 son diagramas de bloques que representan disposiciones ejemplares de componentes de sistemas en los que pueden funcionar aspectos de la materia de asunto descrita en esta memoria. Los componentes ilustrados en las figuras 2-5 son ejemplares y no se entienden como inclusivos de componentes que pueden ser necesarios o estar incluidos. En otras realizaciones, los componentes y/o funciones descritos conjuntamente con las figuras 2-5 pueden ser incluidos en otros componentes (mostrados o no mostrados) o colocados en subcomponentes sin salir del alcance de aspectos de la materia de asunto descrita en esta memoria. En algunas realizaciones, los componentes y/o las funciones descritos conjuntamente con las figuras 2-5 pueden ser distribuidos en múltiples dispositivos.

Cambiando a la figura 2, el sistema 205 puede incluir un iniciador 210, componentes de acceso a datos 215, proveedor(es) de token 225, un almacén 220, y otros componentes (no se muestran). El sistema 205 puede ser implementado por medio de uno o más dispositivos informáticos. Tales dispositivos pueden incluir, por ejemplo, ordenador personales, ordenadores servidores, dispositivos de ordenador de mano o portátil, sistemas multiprocesador, sistemas basados en microcontrolador, descodificadores, electrónica programable de consumo, PC

de red, miniordenadores, ordenadores mainframe, teléfonos móviles, ayudantes digitales personales (PDA), dispositivos de juego, impresoras, aparatos que incluyen descodificador, centro multimedia, u otros aparatos, dispositivos informáticos integrados en o conectados a automóvil, otros dispositivos móviles, entornos informáticos distribuidos que incluyen cualquiera de los sistemas o dispositivos anteriores, y similares.

Donde el sistema 205 comprende un único dispositivo, un dispositivo ejemplar que se puede configurar para actuar como sistema 205 comprende el ordenador 110 de la figura 1. Donde el sistema 205 comprende múltiples dispositivos, uno o más de los múltiples dispositivos puede comprender el ordenador 110 de la figura 1 donde los múltiples dispositivos se pueden configurar de manera similar o de manera diferente.

Los componentes de acceso a datos 215 se pueden usar para trasmitir datos hacia y desde el almacén 220. Los componentes de acceso a datos 215 puede incluir, por ejemplo, uno o más de: gestores de E/S, filtros, impulsores, componentes de servidor de archivo, componentes en una red de área de almacenamiento (SAN) u otro dispositivo de almacenamiento, y otros componentes (no se muestra). Como se emplea en esta memoria, un SAN se puede implementar, por ejemplo, como dispositivo que expone objetivos de almacenamiento lógico, como red de comunicación que incluye tales dispositivos, o algo semejante.

En una realización, un componente de acceso a datos puede comprender cualquier componente al que se da la oportunidad de examinar E/S entre el iniciador 210 y el almacén 220 y que sea capaz de cambiar, completar o fallar la E/S o realizar otras acciones basadas en el mismo o ninguna. Por ejemplo, donde el sistema 205 reside en un único dispositivo, los componentes de acceso a datos 215 pueden incluir cualquier objeto en una pila E/S entre el iniciador 210 y el almacén 220. Donde el sistema 205 es implementado por múltiples dispositivos, los componentes de acceso a datos 215 pueden incluir componentes en un dispositivo que hospeda el iniciador 210, componentes en un dispositivo que proporciona acceso al almacén 220, y/o componentes en otros dispositivos y similares. En otra realización, los componentes de acceso a datos 215 pueden incluir cualesquiera componentes (p. ej., tal como un servicio, base de datos, o algo semejante) usados por un componente a través de los que pasa la E/S incluso si los datos no fluyen a través de los componentes usados.

Como se emplea en esta memoria, el término componente se tiene que entender como que incluye hardware tal como todo o una parte de un dispositivo, una colección de uno o más módulos de software o partes del mismo, alguna combinación de uno o más módulos de software o partes del mismo y uno o más dispositivos o partes del mismo, y similares. Un componente puede incluir o ser representado por código.

Como se emplea en esta memoria, el término código informático se tiene que entender como que incluye instrucciones que indican acciones que debe realizar un ordenador. Estas instrucciones se pueden incluir en cualesquiera medios legibles por ordenador, volátiles o no volátiles.

En una realización, el almacén 220 es cualquier medio de almacenamiento capaz de almacenar datos. El almacén 220 puede incluir memoria volátil (p. ej., un caché) y memoria no volátil (p. ej., un almacenamiento persistente). El término datos se tiene que entender ampliamente como que incluye cualquier cosa que pueda ser representada por uno o más elementos de almacenamiento informático. Lógicamente, los datos se pueden representar como series de unos y ceros en memoria volátil o no volátil. En ordenadores que tienen un medio de almacenamiento no binario, se pueden representar datos según las capacidades del medio de almacenamiento. Los datos se pueden organizar en diferentes tipos de estructuras de datos que incluyen tipos de datos simples tales como números, letras, y similares, tipos de datos jerárquicos, enlazados u otros relacionados, estructuras de datos que incluyen otras múltiples estructuras de datos o tipos de datos simples, y similares. Algunos ejemplos de datos incluyen información, código de programa, estado de programa, datos de programa, órdenes, otros datos, o algo semejante.

El almacén 220 puede comprender almacenamiento en disco duro, estado sólido, u otro almacenamiento no volátil, memoria volátil tal como RAM, otro almacenamiento, alguna combinación de los anteriores, y similares y se puede distribuir en múltiples dispositivos (p. ej., múltiples SAN, múltiples servidores de archivos, una combinación de dispositivos heterogéneos, y similares). Los dispositivos usados para implementar el almacén 220 se pueden ubicar físicamente juntos (p. ej., en un único dispositivo, en un centro de datos, o algo semejante) o distribuidos geográficamente. El almacén 220 se puede disponer en una disposición de almacenamiento escalonado o una disposición de almacenamiento no escalonado. El almacén 220 puede ser externo, interno, o incluir componentes que son tanto internos como externos a uno o más dispositivos que implementan el sistema 205. El almacén 220 puede ser formateado (p. ej., con un sistema de archivos) o no formateado (p. ej., raw).

En otra realización, el almacén 220 puede ser implementado como contenedor de almacenamiento en lugar de almacenamiento físico directo. Un contenedor de almacenamiento puede incluir, por ejemplo, un archivo, volumen, disco, disco virtual, unidad lógica, disco lógico, clone escribible, instantánea de volumen, instantánea de disco lógico, disco físico, almacenamiento de estado sólido (SSD), duro disco, flujo de datos, flujo de datos alternos, flujo de metadatos, o algo semejante. Por ejemplo, el almacén 220 puede ser implementado por un servidor que tiene múltiples dispositivos de almacenamiento físico. En este ejemplo, el servidor puede presentar una interfaz que permite a un componente de acceso a datos acceder a datos de un almacén que se implementa usando uno o más de los dispositivos de almacenamiento físico o partes del mismo del servidor.

El nivel de abstracción puede ser repetido a cualquier profundidad arbitraria. Por ejemplo, el servidor que proporciona un contenedor de almacenamiento a los componentes de acceso a datos 215 también puede depender de un contenedor de almacenamiento para acceder a datos.

En otra realización, el almacén 220 puede incluir un componente que proporciona una vista a datos que puede persistir en almacenamiento no volátil o no persistir en almacenamiento no volátil.

Uno o más de los componentes de acceso a datos 215 pueden residir en un aparato que hospeda el iniciador 210 mientras uno o más de otros de los componentes de acceso a datos 215 pueden residir en un aparato que hospeda o proporciona acceso al almacén 220. Por ejemplo, si el iniciador 210 es una aplicación que se ejecuta en un ordenador personal, uno o más de los componentes de acceso a datos 215 puede residir en un sistema operativo hospedado en el ordenador personal. Un ejemplo de esto se ilustra en la figura 3.

Como otro ejemplo, si el almacén 220 es implementado por una red de área de almacenamiento (SAN), uno o más de los componentes de acceso a datos 215 pueden implementar un sistema operativo de almacenamiento que gestiona y/o proporciona acceso al almacén 220. Cuando el iniciador 210 y el almacén 220 se hospedan en un único aparato, todos o muchos de los componentes de acceso a datos 215 también pueden residir en el aparato.

Una lectura de descarga permite a un iniciador obtener un token que representa datos de un almacén. Usando este token, el iniciador u otro iniciador puede solicitar una escritura de descarga. Una escritura de descarga permite a un iniciador provocar que un proveedor de descarga escriba algunos o todos los datos representados por el token.

En una realización, un token es un número impredecible que se obtiene por medio de una lectura de descarga exitosa. Un token representa datos que son inmutables siempre que el token sea válido. A los datos que un token representa a veces se le hace referencia como datos masivos.

Un proveedor de descarga es una entidad (posiblemente que incluye múltiples componentes dispersos en múltiples dispositivos) que proporciona acceso indirecto a datos asociados con un token. Lógicamente, un proveedor de descarga es capaz de realizar lectura de descarga y/o escritura de descarga. Físicamente, un proveedor de descarga puede ser implementado por uno o más de los componentes de acceso a datos 215 y un proveedor de token.

Al servir una lectura de descarga o una escritura de descarga, un proveedor de descarga puede realizar lógicamente operaciones sobre los datos del almacén y/o en tokens asociados con un proveedor de token. Por ejemplo, para una lectura de descarga, un proveedor de descarga puede copiar lógicamente datos desde un contenedor de almacenamiento lógico respaldado por datos de un almacén a un token (que también pueden ser respaldados por datos del almacén), mientras que para una escritura de descarga, el proveedor de descarga puede copiar lógicamente datos desde un token a un contenedor de almacenamiento lógico respaldado por datos del almacén.

Un proveedor de descarga puede trasferir datos desde un almacén de origen, escribir datos a un almacén de destino, y mantener datos a proporcionar al recibir un token asociado con los datos. En algunas implementaciones, un proveedor de descarga puede indicar que una orden de escritura de descarga se completa después de que los datos se han escrito lógicamente al almacén de destino. Adicionalmente, un proveedor de descarga puede indicar que una orden de escritura de descarga se completa pero se aplaza escribir físicamente datos asociados con la escritura de descarga hasta que sea conveniente.

En algunas implementaciones, un proveedor de descarga puede compartir datos entre un primer contenedor de almacenamiento lógico y un segundo contenedor de almacenamiento lógico, y puede compartir datos entre un token y un contenedor de almacenamiento. El proveedor de descarga puede dejar de compartir datos como parte de realizar una escritura en ubicaciones de almacenamiento físico del almacén que de otro modo provocaría que más de un contenedor de almacenamiento fuera modificado, o de otro modo provocaría que los datos representados por un token cambien.

En algunas implementaciones, un proveedor de descarga puede realizar una copia lógica desde un contenedor de almacenamiento a un token o una copia lógica desde un token a un contenedor de almacenamiento al empezar a compartir datos entre un token y un contenedor de almacenamiento. Por ejemplo, el proveedor de descarga puede realizar una lectura de descarga al copiar lógicamente los datos desde el contenedor de almacenamiento de origen al token al empezar a compartir datos entre el contenedor de almacenamiento de origen y el token. En otro ejemplo, el proveedor de descarga puede realizar una escritura de descarga al copiar lógicamente los datos desde el token al contenedor de almacenamiento de destino al empezar a compartir datos entre el token y el contenedor de almacenamiento de destino.

En algunas implementaciones, un proveedor de descarga puede invalidar un token para, por ejemplo, evitar compartir datos y/o evitar copiar físicamente datos. Por ejemplo, el proveedor de descarga puede realizar una escritura de descarga al copiar lógicamente datos desde el token al contenedor de almacenamiento de destino al actualizar las estructuras de datos del contenedor de almacenamiento de destino para referirse a la ubicaciones de almacenamiento físico del almacén referenciado por el token, y conjuntamente con el mismo, invalidar lógicamente al menos una parte del token. Obsérvese que esto todavía puede resultar en que los contenedores de almacenamiento de origen y de destino compartan datos.

En algunas implementaciones, un proveedor de descarga puede empezar a compartir ubicaciones de almacenamiento de datos entre todos tokens y contenedores de almacenamiento que ya comparten los datos, y adicionalmente, otro contenedor de almacenamiento o token. Por ejemplo, para servir una lectura de descarga, un proveedor de descarga puede empezar a compartir entre un contenedor de almacenamiento de origen y un token. Entonces, para servir una escritura de descarga usando el token, el proveedor de descarga puede empezar a compartir entre el contenedor de almacenamiento de origen, el token, y el contenedor de almacenamiento de destino. Si el token es invalidado más tarde, se detiene la compartición del token, pero la compartición entre contenedores de almacenamiento de origen y de destino puede continuar (p. ej., hasta que se recibe una escritura que se dirige a esos datos).

Como se emplea en esta memoria, en una implementación, un proveedor de token es parte de un proveedor de descarga. En esta implementación, donde un proveedor de token se describe como que realizar acciones, se tiene que entender que el proveedor de descarga que incluye el proveedor de token está realizando esas acciones. En otra implementación, un proveedor de token puede estar separado del proveedor de descarga.

Para iniciar una lectura de descarga de datos del almacén 220, el iniciador 210 puede enviar una solicitud para obtener un token que representa los datos usando una orden predefinida (p. ej., por medio de una API). En respuesta, uno o más de los componentes de acceso a datos 215 pueden responder al iniciador 210 al proporcionar uno o más tokens que representan los datos o un subconjunto del mismo. Un token puede ser representado por una secuencia de bytes que se usan para representar datos inmutables. El tamaño de los datos inmutables puede ser mayor, menor o el mismo tamaño que el token.

Con un token, el iniciador 210 puede solicitar que todos o parte de los datos representados por el token sean escritos lógicamente. A veces en esta memoria esta operación se llama escritura de descarga. El iniciador 210 puede hacer esto al enviar el token junto con un o más compensaciones y longitudes a los componentes de acceso a datos 215.

Los componentes de acceso a datos 215 pueden ser implementados como pila de almacenamiento donde cada capa de la pila puede realizar una función diferente. Por ejemplo, los componentes de acceso a datos pueden partir datos, dividir peticiones de lectura o escritura de descarga, datos de caché, datos de verificación, datos de instantánea, y similares.

Una o más capas de la pila se pueden asociar con un proveedor de token. Un proveedor de token puede incluir uno o más componentes que pueden generar u obtener tokens que representan partes de los datos del almacén 220 y proporcionan estos tokens a un iniciador.

Para una parte de una escritura de descarga, para un token implicado, una compensación relativa a token se puede indicar también como compensación relativa a destino. Ya sea una o ambas compensaciones pueden ser implícitas o explícitas. Una compensación relativa a token pude representar varios bytes (u otras unidades) desde el comienzo de datos representados por el token, por ejemplo. Una compensación relativa a destino puede representar el número de bytes (u otras unidades) desde el comienzo de datos en el destino. Una longitud puede indicar varios bytes (u otro unidades) que empiezan en la compensación.

Si un componente de acceso a datos 215 falla una lectura o escritura de descarga, se puede devolver un código de error que permite a otro componente de acceso a datos o el iniciador intentar otro mecanismo para leer o escribir los datos.

La figura 3 es un diagrama de bloques que representa generalmente una disposición ejemplar de componentes de sistemas en la que un proveedor de token es hospedado por el dispositivo que hospeda el almacén. Como se ilustra, el sistema 305 incluye el iniciador 210 y el almacén 220 de la figura 2. Los componentes de acceso a datos 215 de la figura 3 se dividen entre los componentes de acceso a datos 310 que residen en el dispositivo 330 que hospeda el iniciador 210 y los componentes de acceso a datos 315 que residen en el dispositivo 335 que hospeda el almacén 220. En otra realización, donde el almacén 220 es externo al dispositivo 335, puede haber componentes de acceso a datos adicionales que proporcionan acceso al almacén 220.

El dispositivo 335 se puede considerar un ejemplo de un proveedor de descarga ya que este dispositivo incluye componentes para realizar lecturas y escrituras de descarga y gestionar tokens.

El proveedor de token 320 puede generar, validar e invalidar tokens. Por ejemplo, cuando el iniciador 210 pregunta a un token por datos en el almacén 220, el proveedor de token 320 puede generar un token que representa los datos. Este token puede ser enviado entonces de regreso al iniciador 210 por medio de los componentes de acceso a datos 310 y 315.

Conjuntamente con generar un token, el proveedor de token 320 puede crear una entrada en el almacén de tokens 325. Esta entrada puede asociar el token con datos que indican dónde se pueden encontrar en el almacén 220 los datos representados por el token. La entrada también puede incluir otros datos usados para gestionar el token tal como cuándo a invalidar el token, un tiempo de vida para el token, otros datos, y similares.

Cuando el iniciador 210 o cualquier otra entidad proporcionan el token al proveedor de token 320, el proveedor de token 320 puede realizar una búsqueda en el almacén de tokens 325 para determinar si el token existe. Si el token

existe y es válido, el proveedor de token 320 puede proporcionar información de ubicación a los componentes de acceso a datos 315 de modo que estos componentes pueden leer o escribir lógicamente o realizar lógicamente otras operaciones con los datos como se ha solicitado.

En otra disposición ejemplar similar a la figura 3, el proveedor de token 320 y el almacén de tokens 325 puede ser incluido en el dispositivo 330, y los componentes de acceso a datos 310 conectados al proveedor de token 320. Por ejemplo, un sistema operativo (SO) del dispositivo 330 puede incluir el proveedor de token 320 y el almacén de tokens 325. En este ejemplo, el iniciador 210 puede asumir la existencia de un proveedor de token y almacén de tokens para todas copias realizadas por el iniciador 210. Con esta suposición, el iniciador 210 se puede implementar para omitir código que se retrasa a lectura y escritura normales.

En el ejemplo anterior, el SO puede implementar lectura de descarga al leer los datos solicitados de los componentes de acceso a datos 315 y almacenar los datos en almacenamiento (volátil o no volátil) del dispositivo 330, crear un nuevo valor token, y asociar el valor token recientemente creado con los datos leídos. El SO puede implementar escritura de descarga al copiar (p. ej., escribir) los datos asociados con el token al destino especificado por el iniciador 210. En este ejemplo, el iniciador 210 puede tener que reintentar una copia en la etapa de lectura de descarga en algunos escenarios, pero este reintento puede ser menos pesado para el iniciador que retrasarse a la lectura y escritura normales.

La figura 4 es un diagrama de bloques que representa generalmente otro entorno ejemplar en el que se pueden implementar aspectos de la materia de asunto descrita en esta memoria. Como se ilustra, el entorno incluye un iniciador de origen 405, un iniciador de destino 406, un contenedor de almacenamiento de origen 410, un contenedor de almacenamiento de destino 411, un almacén físico de origen 415, un almacén físico de destino 416, un proveedor de descarga 420, y puede incluir otros componentes (no se muestran).

El iniciador de origen 405 y el iniciador de destino pueden ser implementados de manera similar al iniciador 210 de la figura 2. El iniciador de origen 405 y el iniciador de destino 406 pueden ser dos entidades separadas o una única entidad.

Si el contenedor de almacenamiento de origen 410 y el contenedor de almacenamiento de destino 411 son implementados por un único sistema, el proveedor de descarga 420 puede ser implementado como uno o más componentes del sistema que implementan los contenedores de almacenamiento. Si el contenedor de almacenamiento de origen 410 y el contenedor de almacenamiento de destino 411 son implementados por diferentes sistemas, el proveedor de descarga 420 puede ser implementado como uno o más componentes que se distribuyen en los sistemas que implementan los contenedores de almacenamiento.

Es más, puede haber más de dos casos de contenedores de almacenamiento y almacenes físicos. Por ejemplo, para un token dado obtenido de un origen, puede haber más de un destino especificado. Por ejemplo, se pueden emitir múltiples escrituras de descarga que se refieren a un único token, y cada escritura de descarga puede tener como objetivo exitosamente cualquier destino conocido por el proveedor de descarga 420.

El almacén físico de origen 415 y el almacén físico de destino 416 pueden ser el mismo almacén o almacenes diferentes. Estos almacenes físicos almacenan datos físicos que devuelven los contenedores de almacenamiento de origen y de destino, y también pueden devolver los datos representados por los tokens.

Aunque únicamente se ilustra como que tiene un contenedor de almacenamiento entre el iniciador y los almacenes físicos, como se ha mencionado previamente, en otras realizaciones puede haber múltiples capas de contenedores de almacenamiento entre el iniciador y los almacenes físicos.

El iniciador de origen 405 puede obtener un token al emitir una lectura de descarga. En respuesta, el proveedor de descarga 420 puede generar un token y proporcionarlo al iniciador de origen 405.

Si el iniciador de origen 405 y el iniciador de destino 406 son entidades separadas, el iniciador de origen 405 puede proporcionar el token al iniciador de destino 406. El iniciador de destino 406 puede entonces usar el token para emitir una escritura de descarga al contenedor de almacenamiento de destino 411.

Al recibir la solicitud de escritura de descarga, el proveedor de descarga 420 puede validar el token y escribir lógicamente datos al contenedor de almacenamiento de destino 411 como es indicado por la solicitud de escritura de descarga.

Proveedores de descarga dispares

La figura 5 es un diagrama de bloques que ilustra proveedores de descarga dispares según aspectos de la materia de asunto descrita en esta memoria. En algunos casos, puede ser deseable obtener un token usando una lectura de descarga dirigida a un proveedor de descarga y luego usar ese token para realizar una escritura de descarga en un proveedor de descarga diferente. Antes de la solicitud de escritura de descarga, los dos proveedores de descarga pueden tener o no cualquier conocimiento de uno acerca del otro.

Según aspectos de la materia de asunto descrita en esta memoria, el proveedor de descarga de origen 520 (más adelante en esta memoria a veces se le hace referencia como “el origen 520”) y el proveedor de descarga de destino 521 (más adelante en esta memoria a veces se le hace referencia como “el destino 521”) puede ser cualquiera de dos proveedores de descarga con un protocolo compatible de proveedor de descarga y un camino de comunicaciones entre ellos (p. ej. ruta de red válida por medio de internet). No se requiere relación de confianza preexistente entre el origen 520 y el destino 521. El iniciador de la lectura de descarga (p. ej., el iniciador de lectura de descarga 505) y el iniciador de la escritura de descarga (p. ej., el iniciador de escritura de descarga 506) pueden ser la misma entidad o no.

El iniciador de lectura de descarga 505 (más adelante en esta memoria a veces se le hace referencia como “el iniciador”) confía en el proveedor de descarga de origen 520 por motivos de ser el origen de los datos solicitados. El iniciador de escritura de descarga 506 confía en el destino por motivos de escribir datos masivos representados por el token a un contenedor de almacenamiento de destino solicitado. El iniciador de lectura de descarga 505 confía en el iniciador de escritura de descarga 506 por motivos de que tiene acceso a los datos masivos y no divulgar los datos masivos a partes inapropiadas. El iniciador de escritura de descarga 506 confía en el iniciador de lectura de descarga 505 por motivos de leer datos.

Estas relaciones de confianza se pueden usar/extender al origen 520 y el destino 521. Los iniciadores pueden demostrar al origen 520 que el destino 521 es un destino correcto para una trasferencia particular (entre potencialmente más de un destino correcto). En una implementación, relaciones de confianza se pueden extender al enviar datos de confianza. Datos de confianza pueden incluir un secreto o quizás datos no secretos. En otras implementaciones, relaciones de confianza se pueden extender por medio de otros medios criptográficos tales como criptografía de clave pública/privada u otros medios criptográficos.

La comunicación entre el origen 520 y el destino 521 puede ser encriptada usando una clave proporcionada por los iniciadores. Esta clave puede ser única para una trasferencia particular. El origen 520 valida que la comunicación desde el destino 521 es desde un proveedor de descarga de destino válido (p. ej., de confianza por los iniciadores). El destino 521 valida que la comunicación es realmente desde el origen 520.

La dirección de red del origen 520 puede ser trasladada al destino 521 en una variedad de maneras que incluyen, por ejemplo, ser incrustado en el token, como datos auxiliares trasladado o conjuntamente con el token, o algo semejante. La dirección de red puede ser cualquier tipo de dirección relevante para cualquier tecnología de comunicaciones digitales, tales como una dirección de internet IPV4 y puerto, dirección IPV6, dirección Ethernet, o algo semejante. La dirección de red puede requerir traslación o conversión en el primer uso. Por ejemplo, la dirección puede ser un identificador que se asigna a una dirección IP actual por medio de una tabla hash distribuida. Dependiendo del tipo de dirección y la alcanzabilidad de la dirección, la dirección puede ser relevante y utilizable en alcance geográfico de cualquier tamaño (desde pequeño a todo el mundo). En una implementación, la dirección de red del destino 521 puede ser trasmitida al origen 520 que puede iniciar el establecimiento del canal de comunicación entre el origen 520 y el destino 521.

La palabra “trasferencia” como se usa más adelante se usa para referirse a una secuencia global de lectura de descarga + escritura de descarga en donde algunos datos masivos se trasfieren desde un origen a un destino sin que los datos masivos tengan que transitar el iniciador(se) de la lectura de descarga y de escritura de descarga.

En una implementación, el token no se usa como el único secreto compartido. Un descarga de copia puede tener destinos potencialmente múltiples que conocen el valor de token. En esta implementación, solo conociendo el valor de token puede no permitir a un presunto proveedor de descarga de origen demostrar al proveedor de descarga de destino que el presunto proveedor de descarga de origen es realmente el proveedor de descarga de origen, en lugar de un proveedor de descarga de destino no cooperante diferente.

A continuación hay algunas etapas ejemplares para trasferir datos desde el origen 520 al destino 521:

1. En un ejemplo, el iniciador de lectura de descarga 505 envía una clave específica de trasferencia (“Ksecret”) al origen 520 junto con la solicitud de lectura de descarga. En otro ejemplo, el origen 520 genera y proporciona una clave separada del valor de token en la respuesta de lectura de descarga.

Esto establece un secreto compartido (“Ksecret”) entre el origen 520 y el iniciador de lectura de descarga 505, con el secreto compartido tan seguro como habrían sido los datos masivos (p. ej., trasladados por medio del mismo canal de comunicación que habrían sido los datos masivos, si se usara lectura/escritura normal en lugar de descarga). En algunas implementaciones, el valor de Ksecret puede ser el mismo para más de una trasferencia, potencialmente por el uso de más de un token.

2. El iniciador de lectura de descarga 505 puede reenviar el token al iniciador de escritura de descarga 506 (o pueden ser el mismo iniciador). Si el iniciador de lectura de descarga 505 y el iniciador de escritura de descarga 506 son iniciadores separados, el iniciador de escritura de descarga 506 puede ser tratado como parte de la pila de almacenamiento de destino. Con esto en mente, a continuación, el iniciador de lectura de descarga y el iniciador de escritura de descarga se describen como ambos iniciados por el iniciador de lectura de descarga 505 (más adelante

en esta memoria a veces se le hace referencia simplemente como “el iniciador”).

En una implementación, las comunicaciones de ida y vuelta se pueden minimizar para eliminar cualesquiera comunicaciones que no sean necesarias. Una escritura de descarga es una escritura de descarga independientemente de la forma. Por ejemplo, reenviar un token a una máquina diferente que a su vez emite una escritura de descarga es realmente justo una manera diferente para que un iniciador de lectura de descarga inicie una escritura de descarga.

3. Cuando se realiza una escritura de descarga, el iniciador 505 envía el token (como es habitual) y, adicionalmente también puede enviar <salt, Ksecret(salt)>. Obsérvese que Ksecret(salt) es un valor numérico (la salt) encriptado con la clave secreta (Ksecret) que es conocida únicamente por el origen 520 y el iniciador. El destino 521 no conoce el propio Ksecret, pero el destino 521 conoce que Ksecret es conocido únicamente por el iniciador y el origen 520. El destino 521 confía implícitamente en el iniciador para comportarse correctamente con respecto a esta trasferencia, puesto que el iniciador justo también podría escribir cualquier cosa que el iniciador quiera escribir.

El destino 521 envía Ksecret(salt) al origen 520 (pero no envía el valor salt al origen), y el origen 520 puede entonces desencriptar Ksecret(salt) para obtener el valor salt, y responder (p. ej., por medio de datos encriptados mediante la clave compartido entre el origen y destino) al destino 521 con el valor salt. El destino 521 conoce entonces que el origen 520 conoce Ksecret, y por lo tanto que el origen es realmente el origen (el iniciador ya es implícitamente de confianza con respecto a la escritura solicitada, por adelantado - pero ahora el iniciador ha convencido al destino 521 que confíe en el origen 520 con respecto a la escritura solicitada), sin que el destino 521 conozca Ksecret.

En otra implementación, el origen 520 puede enviar Kchannel(marca de tiempo) o Kchannel (<salt proporcionado por el destino 521>) para demostrar que el origen 520 pudo desencriptar Ksecret (<Kchannel, “válido”>), demostrando al destino 521 que el origen 520 conoce Ksecret y por lo tanto conoce que es el origen 520 real.

4. El canal 530 usado para comunicación entre el origen 520 y el destino 521 es un canal nuevo. Desde el punto de vista de seguridad, en una implementación, el canal 530 es al menos tan seguro como los canales entre el iniciador y el origen 520, y entre el iniciador y el destino 521. La parte “al menos tan seguro como” puede cambiar conforme evolucionan la capacidad de computación y las capacidades de desencriptación. En una implementación, el canal puede ser encriptado con una clave algo larga adecuado para datos sumamente sensible. Para tecnología actual una clave AES de 256 bit puede ser suficiente. La longitud y el tipo de esta clave pueden cambiar con la tecnología. Cada uno del origen 520, el destino 516 y el iniciador pueden rehusar participar en una trasferencia si la longitud de clave no es suficientemente larga (p. ej., como especifican sus respectivas configuraciones).

La clave usada para encriptar el canal entre origen y destino puede ser generada por el origen 520, el iniciador o el destino 521. A la clave usada para asegurar el canal a veces se le hace referencia en esta memoria como “Kchannel” para distinguirla de “Ksecret”. En una implementación, Kchannel se genera en el origen 520 y el origen 520 envía Kchannel al iniciador en la respuesta de lectura de descarga.

En otra implementación, Kchannel se genera en el iniciador. En esta implementación, no es necesario enviar la clave desde el origen 520 al iniciador. Adicionalmente, el iniciador puede generar una clave por origen, pareja de destino (y por trasferencia).

5. El iniciador envía <Kchannel, Ksecret(<Kchannel, “válido”>)> al destino 521.

6. El destino 521 extrae la dirección de red del origen 520 del token, o de información trasladada junto con el token (p. ej., desde el origen 520 al iniciador al destino 521).

7. El destino 521 envía Ksecret(<Kchannel, “válido”>) al origen 520.

8. El origen 520 desencripta usando Ksecret para obtener <Kchannel, “válido”>. La parte “válido” que coincide con el valor esperado en el origen 520 demuestra al origen 520 que el destino 521 que ha enviado el mensaje es realmente un destino válido.

9. El origen 520 puede enviar Kchannel(marca de tiempo) o Kchannel (<salt proporcionado por el destino 521>) para demostrar que el origen 520 pudo desencriptar Ksecret (<Kchannel, “válido”>), demostrando al destino 521 que el origen 520 conoce Ksecret y por lo tanto conoce que es el origen real.

10. Desde este punto en adelante, el origen 520 y el destino 521 pueden comunicarse usando mensajes encriptados que son encriptados con Kchannel. Cada mensaje puede incluir un valor de comprobación de validez y número de secuencia que únicamente coincidirá con el valor esperado si el mensaje no ha sido alterado. Ninguna otra entidad puede conocer Kchannel distinto a aquellas entidades que tendrían acceso a los datos masivos aunque se haya usado lectura/escritura normal en lugar de operaciones de descarga.

Para seguridad añadida, cualquier capa de la pila de almacenamiento de destino que necesita procesar una escritura de descarga al enviar dos o más escrituras de descarga a dos o más destinos puede evitar el uso del mismo clave Kchannel con dos destinos diferentes, especialmente cuando hay cualquier posibilidad de que los dos o más destinos pudieran no estar dentro precisamente de las mismas fronteras de confianza. Esto se puede hacer, por ejemplo, para

impedir que un destino se entrometa en el canal entre el origen 520 y un destino diferente para determinar que el destino diferente ha recibido algunos datos del mismo token. En cambio, este tipo de capa puede evitar esto al limitar su manejo de la escritura de descarga a un único destino por clave Kchannel.

Obsérvese que relaciones de confianza en la pila de destinos son de manera que todas las capas “más altas” de la pila de destinos pueden tener la clave Kchannel usada en una capa inferior. Dos bifurcaciones separadas de una capa inferior también pueden tener la misma clave entre sí o pueden tener claves diferentes para seguridad añadida.

Donde ocurre truncamiento durante una operación de descarga, el truncamiento puede ser propagado justo hacia atrás al iniciador de la lectura de descarga, que entonces puede generar una nueva clave Kchannel y emitir una nueva escritura de descarga usando el mismo token.

Si el procesamiento parcial es demasiado oneroso, el iniciador puede proporcionar más de una clave Kchannel en la escritura de descarga inicial (junto con Ksecret(Kchannel, “válido”) para cada uno), y una capa de pila puede evitar usar la misma clave Kchannel en dos bifurcaciones para capas inferiores.

Si el iniciador soporta más de un token por trasferencia, una clave Kchannel separada por destino puede ser suficiente (en lugar de un Kchannel separado por token).

En una implementación, el canal entre origen 520 y destino 521 puede ser usado para mover datos masivos para más de un token, siempre que todos los tokens sean para la misma trasferencia y el mismo origen.

11. El destino 521 envía el token(s), encriptados usando Kchannel, al origen 520, junto con una indicación de qué partes de los datos masivos del token(s) están siendo solicitados. El destino 521 puede abstenerse de decir al origen 520 dónde se escribirán los datos. Enviar el token al origen encriptado de esta manera no divulga Kchannel a ninguna entidad que aún no la conoce, y no divulga el token al receptor del mensaje a menos que el receptor del mensaje tenga Kchannel.

12. El origen 520 envía los datos solicitados masivos al destino, encriptados usando Kchannel.

13. El destino 521 escribe al menos algunos de los datos masivos a las compensaciones del destino 521 solicitado en la orden de escritura de descarga.

14. El destino 521 completa la orden de escritura de descarga, que informa al iniciador de que la trasferencia está hecha.

Los mensajes implicados entre iniciador, origen 520 y destino 521 pueden ser combinados de manera que únicamente se necesita una solicitud de lectura de descarga/respuesta desde el iniciador al origen 520, únicamente se necesita una escritura de descarga entre iniciador y destino 521, y únicamente se necesita un intercambio desde destino 521 al origen 520 y hacia atrás. Esto se puede hacer, por ejemplo, para reducir la latencia total (p. ej., de ida y vuelta desde el iniciador al origen 520, de ida y vuelta desde el iniciador al destino 516, y de ida y vuelta del destino 521 al origen 520).

Datos trasmitidos a través del canal encriptado pueden ser comprimidos.

Múltiples trasferencias pueden ocurrir concurrentemente, especialmente cuando se tiene que copiar una gran cantidad de datos masivos. Se pueden hacer trasferencias simultáneas de tal manera como para mantener el enlace de red desde el origen 520 al destino 521 ocupado trasfiriendo datos masivos.

Las figuras 6-8 son diagramas de flujo que representan generalmente acciones ejemplares que pueden ocurrir según aspectos de la materia de asunto descrita en esta memoria. Por simplicidad de explicación, la metodología descrita conjuntamente con las figuras 6-8 se representa y describe como serie de actos. Se tiene que entender y apreciar que aspectos de la materia de asunto descrita en esta memoria están limitados por los actos ilustrados y/o por el orden de los actos. En una realización, los actos ocurren en un orden como se describe más adelante. En otras realizaciones, sin embargo, los actos puede ocurrir en paralelo, en otro orden, y/o con otros actos no presentados y descritos en esta memoria. Es más, no todos los actos ilustrados pueden ser requeridos para implementar la metodología según aspectos de la materia de asunto descrita en esta memoria. Adicionalmente, los expertos en la técnica entenderán y apreciarán que la metodología como alternativa se podría representar como serie de estados interrelacionados por medio de un diagrama de estado o como acontecimientos.

La figura 6 es un diagrama de flujo que generalmente representa acciones ejemplares que pueden ocurrir en un iniciador según aspectos de la materia de asunto descrita en esta memoria. En 605, comienzan las acciones.

En 610, se envía una solicitud para un token de lectura de descarga. Por ejemplo, haciendo referencia a la figura 5, el iniciador 505 envía una solicitud de lectura de descarga a la pila de almacenamiento de origen 515. La pila de almacenamiento de origen 515 entrega la solicitud de lectura de descarga al origen 520.

En 615, en respuesta a la solicitud, se recibe un token del proveedor de descarga de origen. Por ejemplo, haciendo

referencia a la figura 5, el origen 520 envía un token al iniciador 505. El token puede ser trasladado en el mismo mensaje con o por separado de otros datos tales como un secreto para extender la confianza y un canal secreto como se ha mencionado previamente.

En 620, se genera/obtiene un secreto. Si se genera en el iniciador, el secreto puede ser generado antes de las acciones indicadas por 610 y ser enviado en la solicitud por el token. En una alternativa, el secreto también puede ser generado después de enviarse la solicitud para el token y puede ser enviado después de enviarse la solicitud.

Como otra alternativa, en lugar de generar el secreto por el iniciador, el secreto puede ser obtenido del proveedor de descarga de origen. En esta alternativa, el secreto puede ser obtenido de la respuesta de lectura de descarga que traslada el token o en un mensaje separado.

Las acciones de 625 y 630 pueden ser combinadas o separadas. En una implementación, el secreto se puede proporcionar junto con el token en una única orden de escritura de descarga. En otra implementación, el secreto se puede proporcionar en un mensaje separado de la orden de escritura de descarga en la que se envía el token.

En 625, se proporciona el secreto al proveedor de descarga de destino. Por ejemplo, haciendo referencia a la figura 4, el iniciador de lectura de descarga 505 proporciona el secreto al iniciador de escritura de descarga 506 que proporciona el secreto al proveedor de descarga de destino 521 por medio de la pila de almacenamiento de destino 516. El secreto permite la extensión de confianza como se ha indicado previamente. En particular, el secreto permite al proveedor de descarga de origen confiar en el proveedor de descarga de destino ya que el proveedor de descarga de origen confía en el iniciador y para que el proveedor de descarga de destino confíe en el proveedor de descarga de origen ya que el proveedor de descarga de destino confía en el iniciador.

En una implementación, el secreto se puede proporcionar junto con el token en una única orden de escritura de descarga. En otra implementación, el secreto se puede proporcionar en un mensaje separado de la orden de escritura de descarga en la que se envía el token.

En 630, se proporciona el token al proveedor de descarga de destino en una orden de escritura de descarga. Por ejemplo, haciendo referencia a la figura 4, el iniciador de lectura de descarga 505 puede proporcionar el token al iniciador de escritura de descarga 506 (si son entidades separadas) que proporciona el token al proveedor de descarga de destino 521 por medio de la pila de almacenamiento de destino 516. La orden de escritura de descarga indica una instrucción para copiar al menos una parte de los datos inmutables desde el proveedor de descarga de origen al proveedor de descarga de destino.

El proveedor de descarga de origen y el proveedor de descarga de destino pueden ser el mismo proveedor de descarga o pueden ser independientes entre sí. Independientes significa que son proveedores de descarga diferentes y no son el mismo proveedor de descarga. En una implementación, proveedores de descarga independientes pueden ser respaldados por uno o más de los mismos dispositivos de almacenamiento y/o pueden compartir uno o más componentes. En otra implementación, proveedores de descarga independientes no son respaldados por ningún almacenamiento común y no comparten ningunos componentes.

Un ejemplo donde el origen y los proveedores de descarga de destino son el mismo proveedor de descarga (y por tanto no independientes) se ilustra en la figura 4 con el proveedor de descarga 420.

En el bloque 635, se pueden realizar otras acciones, si las hay. Otras acciones pueden incluir, por ejemplo, proporcionar una clave Kchannel con la que asegurar un canal de comunicaciones entre el proveedor de descarga de origen y el proveedor de descarga de destino. Esta clave Kchannel puede ser generada por el iniciador, el origen o el destino. Puede haber una clave Kchannel adicional para cada proveedor de descarga de destino adicional a la que se pueden copiar datos representados por el token.

Otras acciones también pueden incluir, por ejemplo, proporcionar una dirección de red del proveedor de descarga de origen al proveedor de descarga de destino al incrustar la dirección de red en el token.

Otras acciones también pueden incluir, por ejemplo, proporcionar una dirección de red del proveedor de descarga de origen al proveedor de descarga de destino al incrustar en el token una clave de búsqueda utilizable para encontrar la dirección de red.

Otras acciones también pueden incluir, por ejemplo, cualesquiera otras acciones indicadas en esta memoria como concerniente al iniciador.

La figura 7 es un diagrama de flujo que generalmente representa acciones ejemplares que pueden ocurrir en un destino según aspectos de la materia de asunto descrita en esta memoria.

En 705, comienzan las acciones.

En 710, se recibe un token desde el iniciador. Por ejemplo, haciendo referencia a la figura 5, el destino 521 recibe un token del iniciador de lectura de descarga 505. El token es generado por un proveedor de descarga de origen que

asegura inmutabilidad de los datos siempre que el token sea válido.

En 715, se recibe un secreto desde el iniciador. Por ejemplo, haciendo referencia a la figura 5, el destino 521 recibe un secreto del iniciador de lectura de descarga 505. El secreto permite la extensión de confianza como se ha mencionado anteriormente.

El token y el secreto pueden ser recibidos en un único mensaje o en múltiples mensajes desde el iniciador.

En 720, se reciben otros datos en el destino. Estos otros datos pueden incluir, por ejemplo, una clave Kchannel con la que asegurar un canal de comunicaciones entre el proveedor de descarga de destino y el proveedor de descarga de origen, una dirección de red, u otros datos que han sido mencionados en esta memoria.

En 725, ocurre autenticación. Por ejemplo, haciendo referencia a la figura 5, usando el secreto y/u otros datos el destino 521 puede autenticar el origen 520 y viceversa.

En 730, se establece un canal seguro con el origen. Por ejemplo, haciendo referencia a la figura 5, el destino 521 establece un canal seguro de comunicaciones con el origen 520 usando una clave Kchannel.

En 735, se proporciona el token al origen. Por ejemplo, haciendo referencia a la figura 5, el destino 521 puede proporcionar el origen 520 con el token conjuntamente con una instrucción para copiar datos masivos representados por el token al destino 521.

En 740, se reciben los datos masivos. Por ejemplo, haciendo referencia a la figura 5, el destino 521 recibe los datos masivos por medio del canal seguro 530 desde el origen 520.

En 745, se pueden realizar otras acciones, si las hay. Otras acciones pueden incluir, por ejemplo, cualesquiera otras acciones descritas en esta memoria como concernientes al destino.

la figura 8 es un diagrama de flujo que representa generalmente acciones ejemplares que pueden ocurrir en un origen según aspectos de la materia de asunto descrita en esta memoria. En 805, comienzan las acciones.

En 810, se puede recibir una solicitud de lectura de descarga desde un iniciador. Por ejemplo, haciendo referencia a la figura 5, el origen 520 puede recibir una solicitud de lectura de descarga desde el iniciador 505. En una implementación, la solicitud de lectura de descarga puede incluir o ser trasladada conjuntamente con un secreto utilizable para extender confianza como se ha indicado previamente.

En otra implementación, en 815, el origen genera el secreto. Por ejemplo, haciendo referencia a la figura 5, el origen 520 puede generar el secreto para enviar de regreso al iniciador 505.

En 820, el token se envía potencialmente con un secreto dependiendo de la implementación. Por ejemplo, haciendo referencia a la figura 5, el origen 520 envía el token con el secreto al iniciador 505.

En 825, se recibe el secreto de un destino. Por ejemplo, haciendo referencia a la figura 5, el origen 520 recibe el secreto del destino 521. Recibir el secreto puede incluir recibir algo derivado del secreto que proporciona evidencia de que el remitente conoce el secreto.

En 830, se establece un canal seguro con el destino. Por ejemplo, haciendo referencia a la figura 5, el origen 520 establece un conexión segura con el destino 521.

En 835, se recibe un token desde el destino. Por ejemplo, haciendo referencia a la figura 5, el origen 520 recibe un token del destino 521. El token puede ser trasladado con una instrucción para enviar todos o una parte de datos masivos asociados con el token.

En bloque 840, se proporcionan los datos masivos. Por ejemplo, haciendo referencia a la figura 5, el origen 520 proporciona al menos algunos datos masivos representados por el token al destino 521.

En bloque 845, se pueden realizar otras acciones, si las hay. Otras acciones pueden incluir, por ejemplo, recibir una dirección de red del destino e iniciar, por parte del origen, el establecimiento de un canal de comunicación con el destino. Otras acciones también pueden incluir, por ejemplo, cualesquiera otras acciones descritas en esta memoria como concernientes al origen.

Como se puede ver de la descripción detallada anterior, se han descrito aspectos relacionados con tecnología de descarga. Si bien en esta memoria aspectos de la materia de asunto descrita son susceptibles a diversas modificaciones y construcciones alternativas, ciertas realizaciones ilustradas de la misma se muestran en los dibujos y se han descrito anteriormente en detalle. Se debe entender, sin embargo, que no hay intención de limitar aspectos de la materia de asunto reivindicada a la forma específica descrita, sino por el contrario, la intención es cubrir todas modificaciones que se encuentren dentro del alcance de la invención como se reivindica.

REIVINDICACIONES

1. Un método implementado por un ordenador, el método comprende en un iniciador:

enviar (610) una solicitud para un token que representa datos inmutables siempre que el token sea válido a un proveedor de descarga de origen;

en respuesta a la solicitud, recibir (615) el token y un secreto del proveedor de descarga de origen;

proporcionar (625) el secreto a un proveedor de descarga de destino, en donde

el secreto permite al proveedor de descarga de origen confiar en el proveedor de descarga de destino como el proveedor de descarga de origen confía en el iniciador, y

el secreto permite al proveedor de descarga de destino confiar en el proveedor de descarga de origen como el proveedor de descarga de destino confía en el iniciador;

proporcionar (630) el token al proveedor de descarga de destino con una instrucción para copiar al menos una parte de los datos inmutables desde el proveedor de descarga de origen al proveedor de descarga de destino, siendo el proveedor de descarga de origen y el proveedor de descarga de destino diferentes proveedores de descarga que son independientes entre sí;

proporcionar una dirección de red del proveedor de descarga de origen al proveedor de descarga de destino por uno de incrustar la dirección de red en el token o incrustar en el token una clave de búsqueda utilizable para encontrar la dirección de red; y

proporcionar una clave Kchannel para establecer un canal seguro de comunicaciones entre el proveedor de descarga de origen y el proveedor de descarga de destino usando la clave Kchannel.

2. El método de la reivindicación 1, que comprende además generar una clave Kchannel adicional para cada proveedor de descarga de destino adicional al que se van a copiar datos representados por el token.

3. Un sistema para un entorno informático, el sistema comprende:

un proveedor de descarga de destino que comprende uno o más ordenadores y uno o más dispositivos de almacenamiento, el proveedor de descarga de destino configurado para ejecutar acciones indicadas por código informático, las acciones comprenden:

recibir (710), desde un iniciador, un token que representa datos inmutables siempre que el token sea válido, el token es de un proveedor de descarga de origen que asegura inmutabilidad de los datos siempre que el token sea válido, siendo el proveedor de descarga de destino y el proveedor de descarga de origen proveedores de descarga diferentes que son independientes entre sí;

recibir (715), desde el iniciador, ya sea una dirección de red del proveedor de descarga de origen incrustado en el token o una clave de búsqueda incrustada en el token utilizable para encontrar la dirección de red;

recibir (715), desde el iniciador, un secreto, en donde

el secreto permite al proveedor de descarga de origen confiar en el proveedor de descarga de destino como el proveedor de descarga de origen confía en el iniciador, y

el secreto permite al proveedor de descarga de destino confiar en el proveedor de descarga de origen como el proveedor de descarga de destino confía en el iniciador;

autenticar (725) el proveedor de descarga de origen usando el secreto; y

establecer (730) un canal seguro de comunicaciones con el proveedor de descarga de origen usando una clave Kchannel,

4. Un medio de almacenamiento informático que tiene instrucciones ejecutables por ordenador, que cuando se ejecutan realizan acciones en un proveedor de descarga de origen, que comprende:

recibir (810), desde un iniciador, una solicitud para un token que representa datos inmutables siempre que el token sea válido;

generar (815) un secreto;

en respuesta a la solicitud, enviar (820) el token y el secreto al iniciador a proporcionar a un proveedor de descarga de destino con una instrucción para copiar al menos una parte de los datos inmutables desde el proveedor de descarga de origen al proveedor de descarga de destino, siendo el proveedor de descarga de origen y el proveedor de descarga

de destino proveedores de descarga diferentes que son independientes entre sí, el proveedor de descarga de origen usa el secreto para autentificar el proveedor de descarga de destino, en donde

el secreto permite al proveedor de descarga de origen confiar en el proveedor de descarga de destino como el proveedor de descarga de origen confía en el iniciador, y

el secreto permite al proveedor de descarga de destino confiar en el proveedor de descarga de origen como el proveedor de descarga de destino confía en el iniciador; y

recibir una dirección de red del proveedor de descarga de destino e iniciar el establecimiento de un canal de comunicación con el proveedor de descarga de destino usando una clave Kchannel.

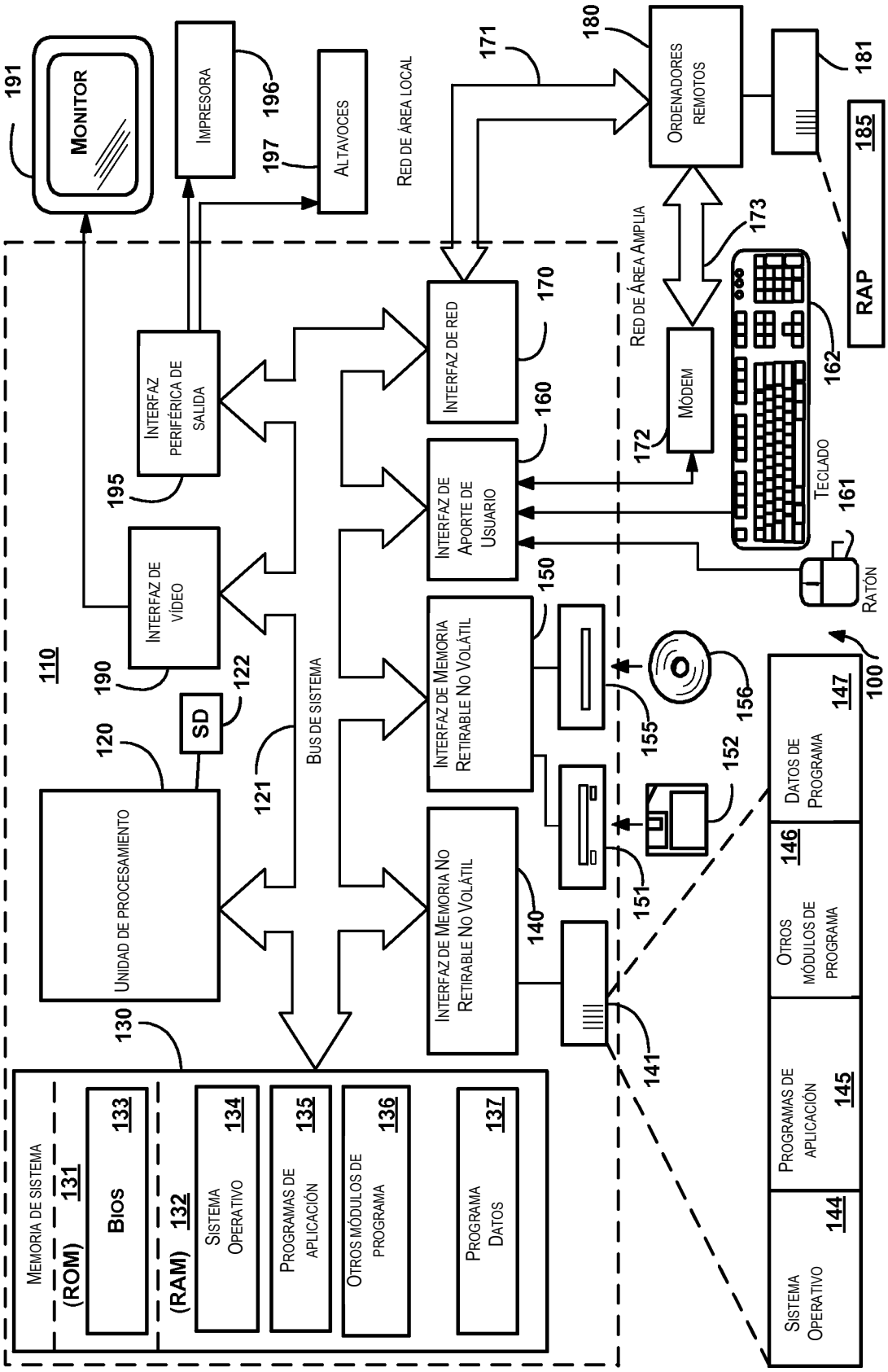
110
MONITOR
191
IMPRESORA
196
197
ALTAVOCES
RED DE ÁREA LOCAL
171
180
ORDENADORES REMOTOS
181
185
RAP
RED DE ÁREA AMPLIA
173
MÓDEM
172
162
TECLADO
161
RATÓN
INTERFAZ PERIFÉRICA DE SALIDA
195
INTERFAZ DE RED
170
160
INTERFAZ DE APORTE DE USUARIO
150
INTERFAZ DE VÍDEO
190
122
SD
120
UNIDAD DE PROCESAMIENTO
121
BUS DE SISTEMA
INTERFAZ DE MEMORIA RETIRABLE NO VOLÁTIL
155
156
152
INTERFAZ DE MEMORIA NO RETIRABLE NO VOLÁTIL
151
140
141
100
DATOS DE PROGRAMA
147
OTROS MÓDULOS DE PROGRAMA
146
PROGRAMAS DE APLICACIÓN
145
SISTEMA OPERATIVO
144
130
MEMORIA DE SISTEMA
(ROM) 131
BIOS 133
(RAM) 132
SISTEMA OPERATIVO 134
PROGRAMAS DE APLICACIÓN 135
OTROS MÓDULOS DE PROGRAMA 136
PROGRAMA DATOS 137

FIG. 1

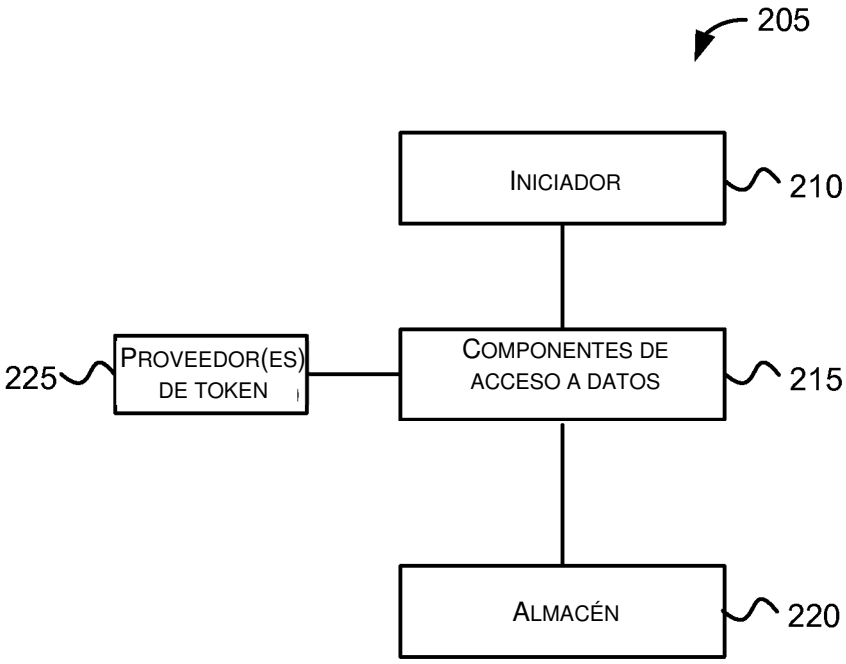
205
INICIADOR
210
COMPONENTES DE ACCESO A DATOS
215
PROVEEDOR(ES) DE TOKEN
225
ALMACÉN
220

FIG. 2

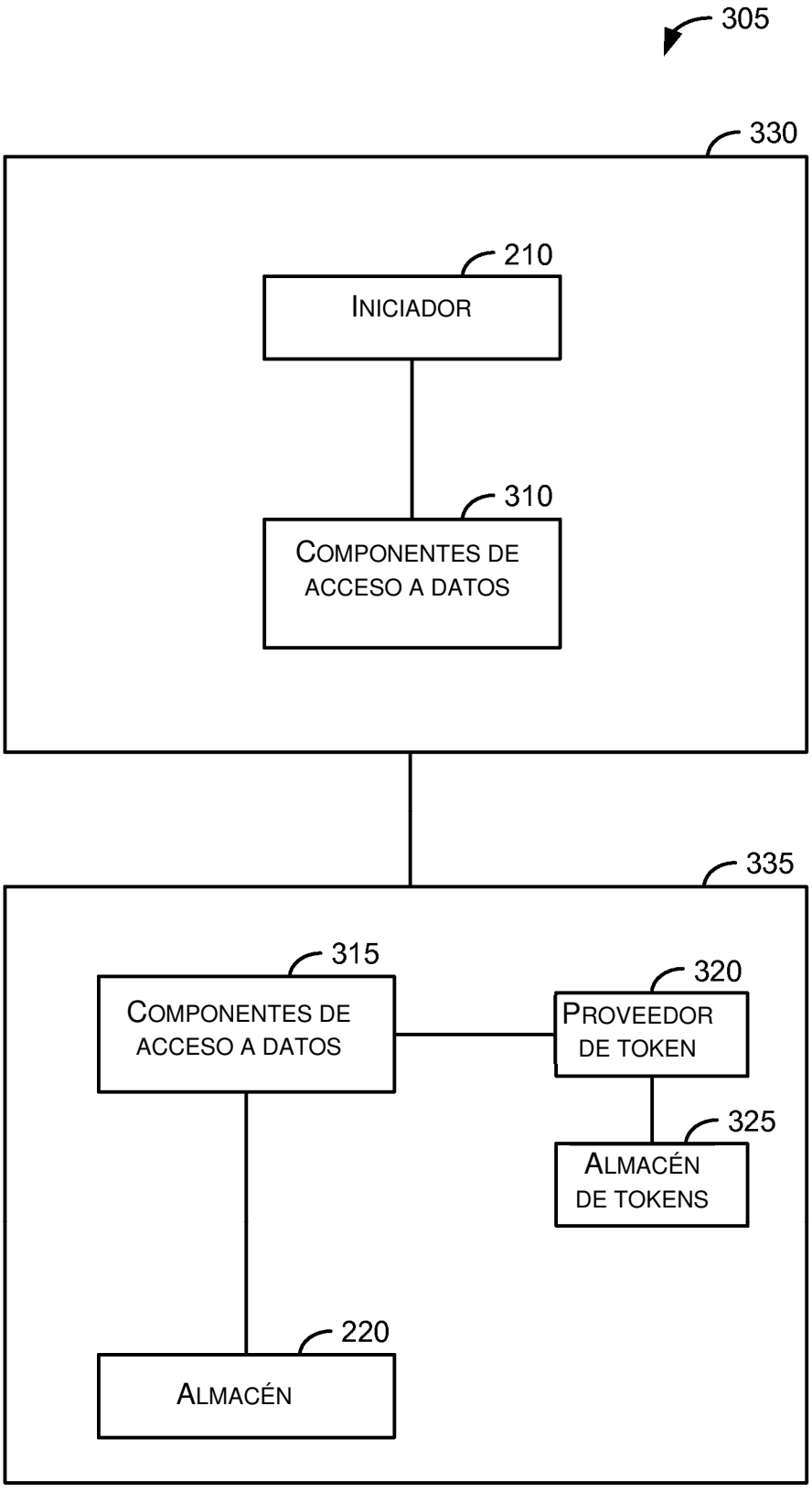
305
330
210
INICIADOR
310
COMPONENTES DE ACCESO A DATOS
335
315
COMPONENTES DE ACCESO A DATOS
320
PROVEEDOR DE TOKEN
325
ALMACÉN DE TOKENS
220
ALMACÉN

FIG. 3

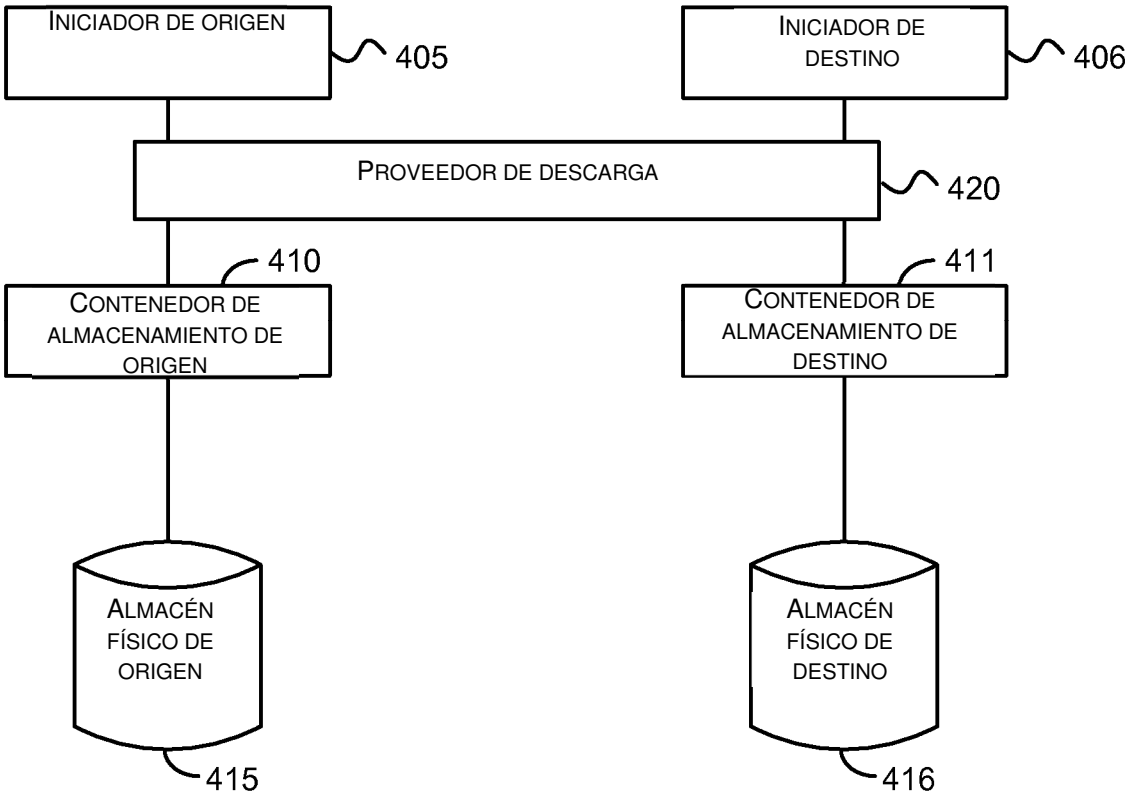
INICIADOR DE ORIGEN
405
INICIADOR DE DESTINO
406
PROVEEDOR DE DESCARGA
420
410
411
CONTENEDOR DE ALMACENAMIENTO DE ORIGEN
CONTENEDOR DE ALMACENAMIENTO DE DESTINO
ALMACÉN FÍSICO DE ORIGEN
ALMACÉN FÍSICO DE DESTINO
415
416

FIG. 4

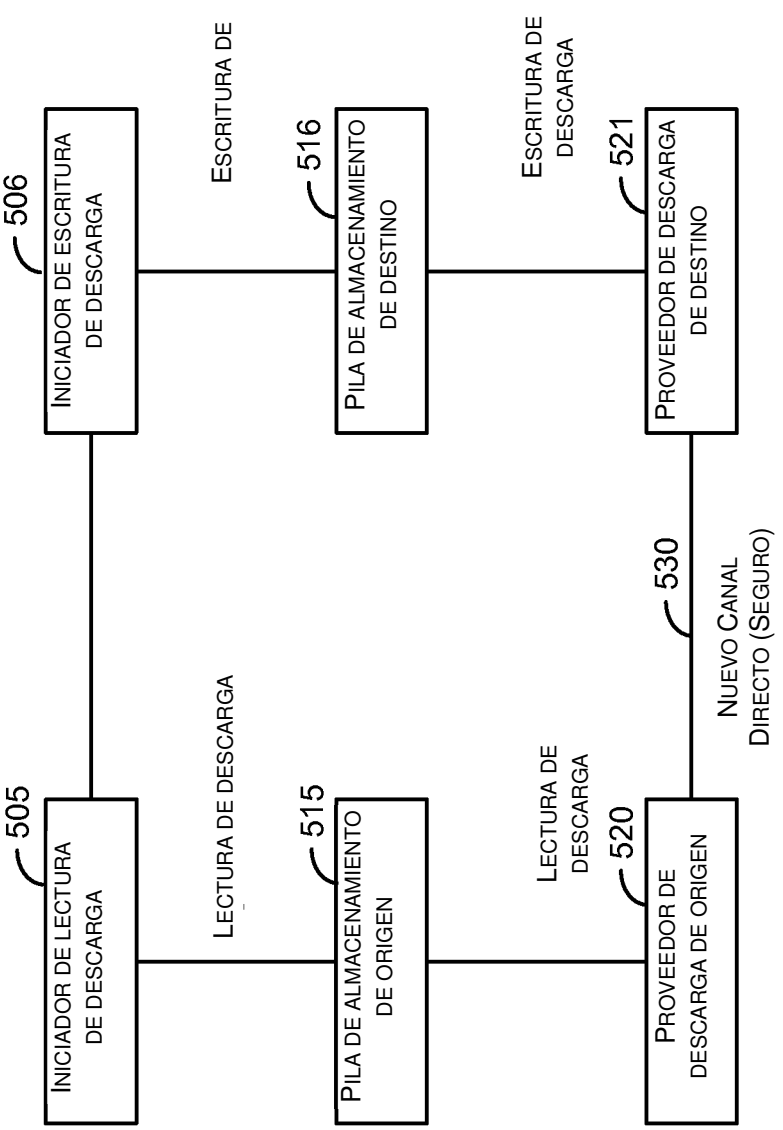
506
INICIADOR DE ESCRITURA DE DESCARGA
ESCRITURA DE
516
PILA DE ALMACENAMIENTO DE DESTINO
ESCRITURA DE DESCARGA
521
PROVEEDOR DE DESCARGA DE DESTINO
530
NUEVO CANAL DIRECTO (SEGURO)
505
INICIADOR DE LECTURA DE DESCARGA
LECTURA DE DESCARGA
515
PILA DE ALMACENAMIENTO DE ORIGEN
LECTURA DE DESCARGA
520
PROVEEDOR DE DESCARGA DE ORIGEN

FIG. 5

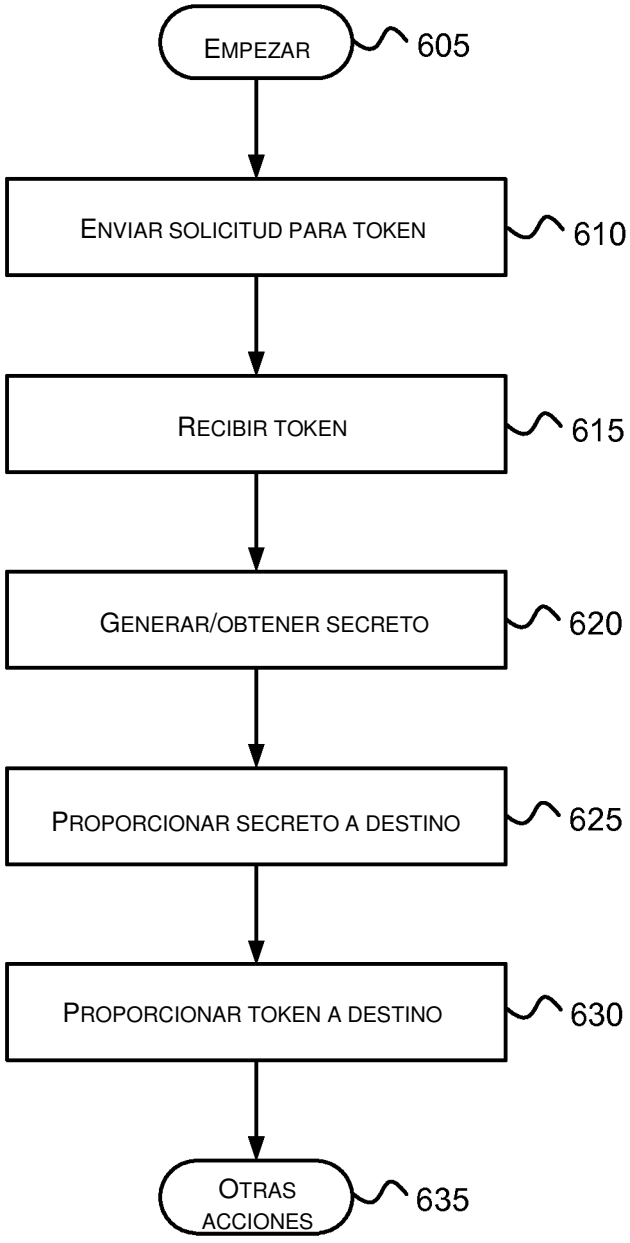
EMPEZAR
605
ENVIAR SOLICITUD PARA TOKEN
610
RECIBIR TOKEN
615
GENERAR/OBTENER SECRETO
620
PROPORCIONAR SECRETO A DESTINO
625
PROPORCIONAR TOKEN A DESTINO
630
OTRAS ACCIONES
635

FIG. 6

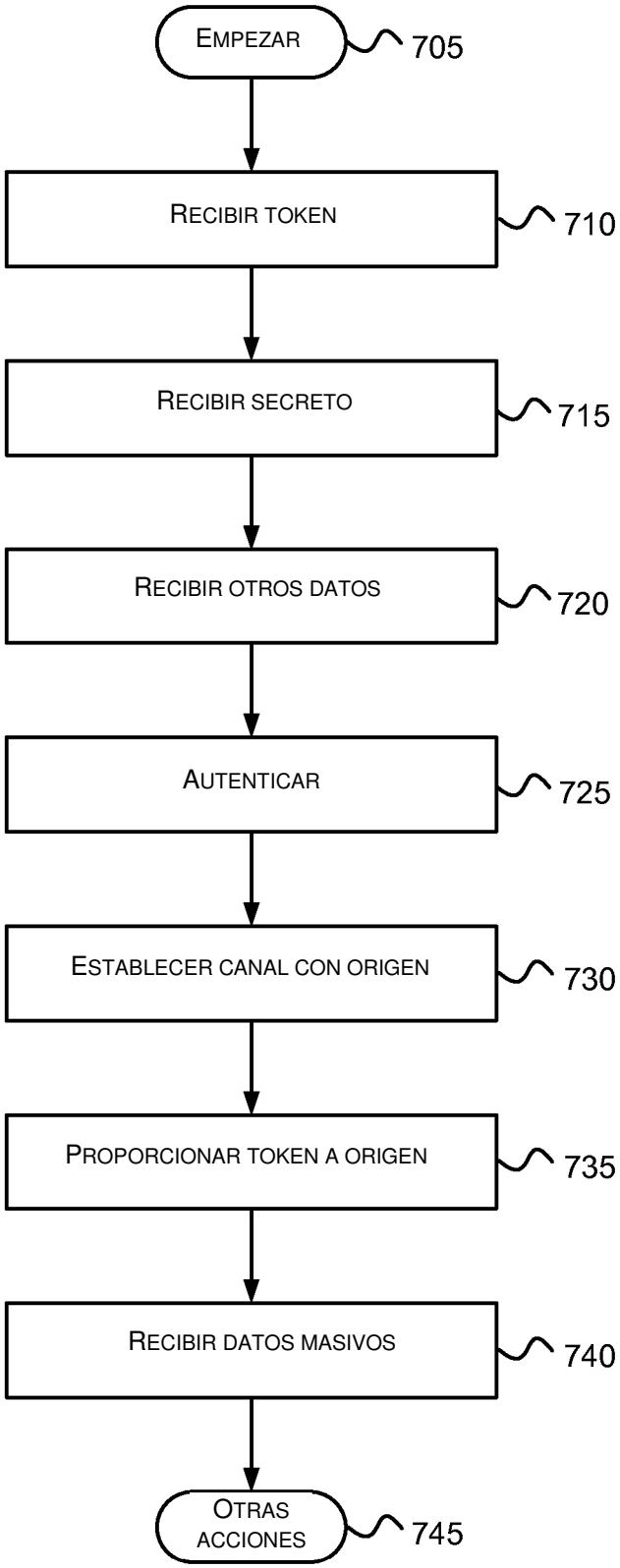
EMPEZAR
705
RECIBIR TOKEN
710
RECIBIR SECRETO
715
RECIBIR OTROS DATOS
720
AUTENTICAR
725
ESTABLECER CANAL CON ORIGEN
730
PROPORCIONAR TOKEN A ORIGEN
735
RECIBIR DATOS MASIVOS
740
OTRAS ACCIONES
745

FIG. 7

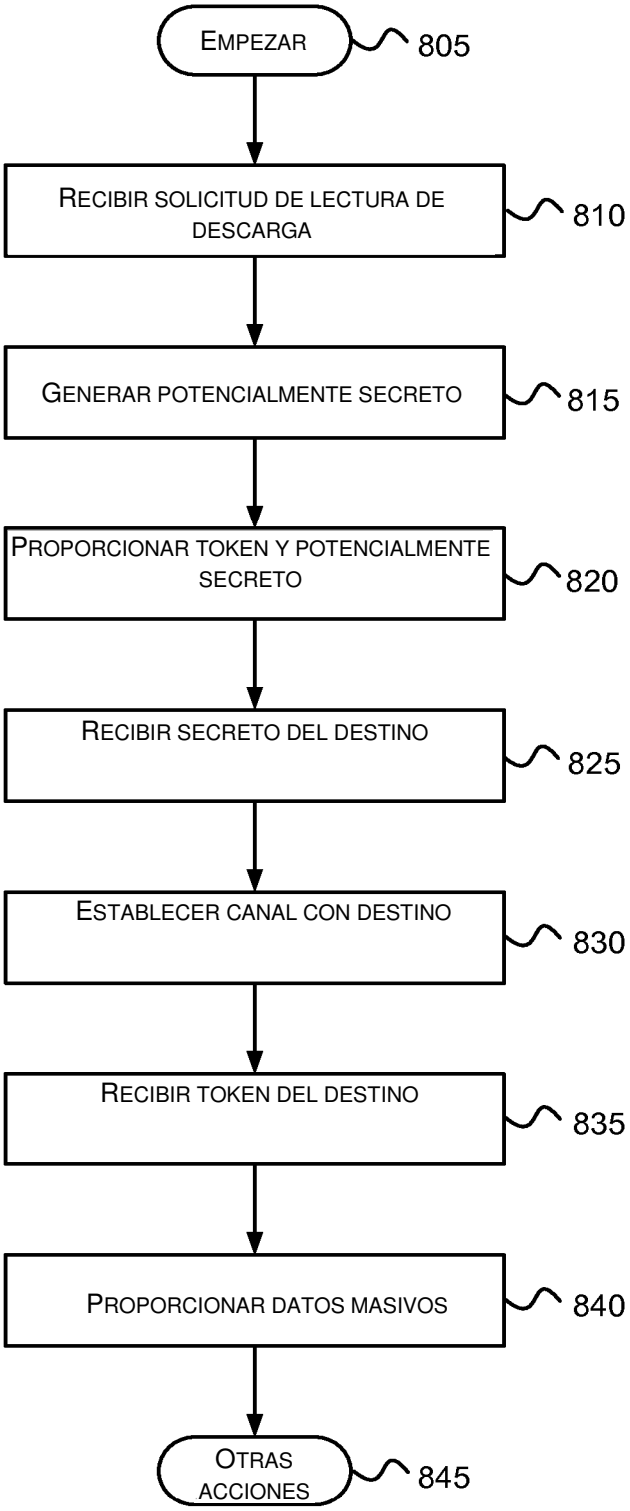
EMPEZAR
805
RECIBIR SOLICITUD DE LECTURA DE DESCARGA
810
GENERAR POTENCIALMENTE SECRETO
815
PROPORCIONAR TOKEN Y POTENCIALMENTE SECRETO
820
RECIBIR SECRETO DEL DESTINO
825
ESTABLECER CANAL CON DESTINO
830
RECIBIR TOKEN DEL DESTINO
835
PROPORCIONAR DATOS MASIVOS
840
OTRAS ACCIONES
845

FIG. 8