(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 750 652**

(51) Int. CI.:

H04L 9/00 (2006.01)
H04L 9/32 (2006.01)
G06F 16/22 (2009.01)
G06F 16/23 (2009.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **30.06.2015 PCT/SE2015/050767**

(87) Fecha y número de publicación internacional: **05.01.2017 WO17003331**

(96) Fecha de presentación y número de la solicitud europea: **30.06.2015 E 15739696 (1)**

(97) Fecha y número de publicación de la concesión europea: **14.08.2019 EP 3318001**

(54) Título: **Métodos y dispositivos para manejar firmas de datos basados en árboles hash**

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
26.03.2020

(73) Titular/es:

TELEFONAKTIEBOLAGET LM ERICSSON (PUBL) (100.0%)
164 83 Stockholm, SE

(72) Inventor/es:

HAKALA, HARRI;
JAATINEN, MIKAEL;
LEHTINEN, HANNU y
MATTILA, LEENA MARJATTA

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Métodos y dispositivos para manejar firmas de datos basados en árboles hash

Campo técnico

La tecnología descrita en la presente memoria se relaciona de manera general con el campo de la protección de la integridad de los datos, y en concreto con los métodos y dispositivos para manejar las firmas de datos basadas en árboles hash.

Antecedentes

La protección tradicional de la integridad de los datos requiere que se instale una herramienta específica en un cliente o que se integre en una aplicación. Por ejemplo, una de dichas herramientas para la integridad de los datos comprende un software que monitoriza y alerta al usuario de eventos sospechosos tales como ciertos tipos de cambios de archivos, protegiendo de este modo el activo de datos. Una alternativa para la protección de la integridad de los datos sobre sistemas de comunicación no seguros es usar una infraestructura de Clave Pública (PKI) basada en tecnología de firma de datos que utiliza parejas de claves pública-privada, en donde la clave privada se mantiene en secreto. En PKI, se usa la firma digital para autenticar un mensaje y de este modo evitar que el mensaje sea alterado en el tránsito. La firma digital cifra el mensaje con la clave privada (firma) del remitente y si se puede descifrar la firma con la clave pública (verificación) asociada del remitente se establecerá la identidad del remitente y se verificará que el mensaje no ha sido alterado desde que se firmó.

La Infraestructura de Firma sin Clave (KSI) es un árbol hash más reciente basado en la tecnología de firma de datos que proporciona servicios de integridad de datos, marcación de la hora e identificación del firmante. La tecnología de firma basada en KSI proporciona una solución alternativa a PKI en la protección de la integridad de datos con prueba de integridad fiable sin asumir el secreto continuo de las claves. La tecnología KSI utiliza hash de datos y árboles de hash para generar testigos de firma para los datos a ser protegidos. KSI tiene algunas ventajas comparada con la tecnología de protección de la integridad de datos tradicional, cuando existe la necesidad de proteger la integridad de cantidades masivas de datos y/o cuando se requiere protección de la integridad de los datos durante un largo tiempo. Por ejemplo, para conjuntos de datos que son tan largos que las aplicaciones de procesamiento de datos tradicionales son inadecuadas, los así llamados Grandes Datos. La tecnología KSI se ajusta mejor que la tecnología PKI.

La tecnología KSI es una excelente opción para la protección de la integridad en, por ejemplo, un entorno de máquina a máquina (por ejemplo, una medición inteligente) y un entorno de Internet de las Cosas (IoT). Un dispositivo que desea usar un servicio KSI para enviar solicitudes de firmas de datos y solicitudes de verificación. El dispositivo combina los datos a ser protegidos y una firma KSI, y por lo tanto necesita almacenar las firmas KSI; por tanto, esto requiere que el dispositivo tenga una capacidad de almacenamiento suficientemente grande. El tamaño típico de una firma KSI es de más de 3 kilobytes, y el tamaño máximo de la firma KSI puede ser de hasta 5 kilobytes (kB). La capacidad de almacenamiento resulta un problema en concreto cuando el dispositivo genere muchos datos que necesiten estar protegidos en su integridad mediante el uso del servicio KSI. Por ejemplo, una media de 1000 firmas KSI generadas durante 24 horas requerirá más de 3 megabytes (MB) de almacenamiento de datos sólo para las firmas KSI.

Sin embargo, dichos dispositivos normalmente tienen limitada la cantidad de su capacidad de almacenamiento de datos, y simplemente extender su capacidad de almacenamiento no es siempre factible, por ejemplo, por razones de coste o simplemente debido a la ausencia de espacio. Otro problema si se aumenta la capacidad de almacenamiento de datos en el dispositivo, es la recuperación de las firmas del mismo, que sería gravosa de ser necesaria por ejemplo para investigaciones forenses digitales, ya que se tendría que recuperar una gran cantidad de firmas desde una gran cantidad de dispositivos.

Además, el ancho de banda de un enlace de comunicación ha de ser capaz de transportar las cantidades de datos necesarias mediante la transferencia de la firma sin poner en peligro la transferencia de otro tráfico hacia y desde el dispositivo. Los dispositivos inteligentes, por ejemplo, pueden tener acceso a una red a través de una interfaz de radio, por ejemplo el Sistema Global para Comunicaciones Móviles (GSM) o el Acceso Múltiple por División de Código de Banda Ancha (WCMDA). Dichas interfaces de radio del dispositivo inteligente podrían no haber sido dimensionadas para transmitir paquetes del tamaño requerido para la transferencia de la firma, mientras también se transmite los datos.

El documento “Infraestructura de Firmas sin Claves: Cómo Construir Árboles Hash Distribuidos Globales” A. Buldas et al. ASOCIACIÓN INTERNACIONAL PARA LA INVESTIGACIÓN CRIPTOLÓGICA, se refiere a una manera de implementar y operar una Infraestructura de Firmas sin Clave (KSI) con un retardo razonable y estable.

El documento US 2009/164517 A1 describe un método para gestionar las firmas digitales de una manera eficiente cuando se aplica en el contexto de investigaciones forenses informáticas, usando firmas para identificar objetivos que tienen un contenido similar.

El documento US 2012/161154 A1 se refiere a un método de cómo acceder a una ubicación lógica que es un archivo almacenado en un sistema de almacenamiento de contenido direccionable.

El número de dispositivos conectados está creciendo de manera exponencial y con dicho aumento de la cantidad de datos generados por ordenador, existe la necesidad de soluciones escalables que puedan proporcionar una prueba de que la operación está libre de manipulación y corrupción.

Compendio

Un objetivo de la presente descripción es solucionar o al menos aliviar al menos uno de los problemas anteriormente mencionados.

El objetivo es según un aspecto logrado mediante un método de manejo de una firma de datos basada en árbol hash. El método se realiza en un primer dispositivo y comprende: recibir, desde un segundo dispositivo, una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación sobre el tipo de almacenamiento de la firma de datos generada; generar, en respuesta a la solicitud de generación de la firma de datos, la firma de datos usando un método de firma de datos basado en árbol hash; y proporcionar, al segundo dispositivo, una referencia a la firma de datos generada, en donde la firma de datos generada se puede obtener por medio de la referencia.

El método permite al proveedor del servicio KSI ofrecer servicios mejorados a sus usuarios. Por ejemplo, proporcionando la referencia a una firma en lugar de la firma, lo que requiere que se usen más recursos de comunicación, se ofrece una comunicación de radio más eficiente al usuario del servicio KSI. Además, almacenando la firma en nombre del dispositivo, se alivia al dispositivo del requisito de una gran capacidad de almacenamiento, lo que es una solución rentable para el dueño del dispositivo. El método proporciona de este modo mejoras a la tecnología KSI existente para alcanzar mejor los requisitos de por ejemplo los dispositivos inteligentes.

También, el riesgo de que la firma se pierda si el dispositivo falla o se compromete, es eliminada almacenando las firmas en la red KSI en lugar de en el dispositivo.

La capacidad de una red KSI, de la cual puede ser parte el primer dispositivo, se puede usar en la implementación de un almacenamiento de firmas “central” que haga la utilización de la infraestructura KSI más eficiente.

El objetivo es según un aspecto alcanzado por un programa informático para un primer dispositivo manejar las firmas de datos basadas en árbol hash. El programa informático comprende código de programa informático, que al ser ejecutado en al menos un procesador en el primer dispositivo provoca que el primer dispositivo realice el método como anteriormente.

El objetivo es según un aspecto alcanzado por un producto de programa informático que comprende un programa informático como anteriormente y unos medios legibles por ordenador en los que se almacena el programa informático.

El objetivo es según un aspecto alcanzado por un primer dispositivo manejar una firma de datos basada en árbol hash. El primer dispositivo se configura para: recibir, desde un segundo dispositivo, una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de la firma de datos generada; generar, en respuesta a la solicitud de generación de firma de datos, la firma de datos que usa un método de firma de datos basado en árbol hash; y proporcionar, al segundo dispositivo, una referencia a la firma de datos generada, en donde la firma de datos generada se puede obtener por medio de la referencia.

El objetivo es según un aspecto alcanzado por un método de manejo una forma de datos basada en árbol hash. El método se realiza en un segundo dispositivo y comprende; enviar, a un primer dispositivo, una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de una firma de datos generada; y recibir, desde el primer dispositivo, una referencia a una firma de datos generada en respuesta a la solicitud, en donde la firma de datos generada se puede obtener por medio de la referencia.

El método permite el uso eficiente de los escasos recursos de la interfaz de radio incluso aunque el dispositivo use tecnología KSI, que convencionalmente involucraría la transmisión de firmas KSI bastante grandes. En concreto, ya que se envía la referencia a una firma en lugar de la firma (como en la técnica anterior), se pueden requerir y usar menos recursos de la interfaz de radio. Además, ya que la firma no necesita ser almacenada en el segundo dispositivo (por ejemplo, un dispositivo inteligente) es menos vulnerable al acceso ilegal. Los dispositivos inteligentes normalmente residen en áreas no seguras mientras que el almacenamiento de firmas de la red KSI se ubica normalmente dentro de una zona segura.

El objetivo es según un aspecto alcanzado por un programa informático para un segundo dispositivo manejar las firmas de datos basadas en árbol hash. El programa informático comprende código de programa informático, que, al ser ejecutado en al menos un procesador en el segundo dispositivo provoca que el segundo dispositivo realice el método como anteriormente.

El objetivo es según un aspecto alcanzado por un producto de programa informático que comprende un programa informático como anteriormente y unos medios legibles por ordenador en los que se almacena el programa informático.

El objetivo es según un aspecto alcanzado por un segundo dispositivo para manejar una firma de datos basada en árbol hash. El segundo dispositivo se configura para: enviar, a un primer dispositivo, una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de la firma de datos generada; y recibir, desde el primer dispositivo, una referencia a una firma de datos generada en respuesta a la solicitud, en donde la firma de datos generada se puede obtener por medio de la referencia.

Características y ventajas adicionales de las realizaciones de las presentes enseñanzas resultarán evidentes tras la lectura de la siguiente descripción y de los dibujos adjuntos.

Breve descripción de los dibujos

La Figura 1 ilustra un sistema de infraestructura de firma sin clave.

La Figura 2 ilustra una cadena hash en un proceso de verificación.

La Figura 3 ilustra el flujo de señalización entre componentes según las realizaciones de las presentes enseñanzas.

La Figura 4 ilustra un entorno en el que se pueden implementar las realizaciones según las presentes enseñanzas.

La Figura 5 ilustra un diagrama de flujo sobre los pasos de una realización de un método en un primer dispositivo de acuerdo con las presentes enseñanzas.

La Figura 6 ilustra de manera esquemática un primer dispositivo y los medios para implementar las realizaciones del método de acuerdo con las presentes enseñanzas.

La Figura 7 ilustra un primer dispositivo que comprende los módulos de función/módulos de software para implementar las realizaciones de las presentes enseñanzas.

La Figura 8 ilustra un diagrama de flujo sobre los pasos de una realización de un método en un segundo dispositivo de acuerdo con las presentes enseñanzas.

La Figura 9 ilustra de manera esquemática un segundo dispositivo y los medios para implementar las realizaciones del método de acuerdo con las presentes enseñanzas.

La Figura 10 ilustra un segundo dispositivo que comprende los módulos de función/módulos de software para implementar las realizaciones de las presentes enseñanzas.

Descripción detallada

En la siguiente descripción, por propósitos de explicación y no de limitación, se exponen los detalles específicos tales como las arquitecturas concretas, las interfaces, las técnicas, etc. para proporcionar una comprensión total. En otros ejemplos, la descripción detallada de los dispositivos, circuitos, y métodos bien conocidos se omiten para no oscurecer la descripción con detalles innecesarios. Los mismos números de referencia se refieren a los mismos o similares elementos a lo largo de la descripción.

En aras de la integridad y para proporcionar un entendimiento completo de las presentes enseñanzas, se proporciona de manera inicial la descripción sobre algunos conceptos básicos de las infraestructuras de Firma Sin Clave (KSI).

La Figura 1 ilustra los componentes de una arquitectura KSI y es un entorno también en el que se pueden implementar aspectos de las presentes enseñanzas. Una red KSI comprende cuatro capas principales.

Una primera capa es una capa 100 de aplicación de usuario KSI para aplicaciones adaptadas a KSI. La aplicación 100 adaptada a KSI (de aquí en adelante también denotada aplicación KSI o cliente KSI) es la fuente de datos a ser firmada mediante el uso de la red 1 KSI. La aplicación 100 KSI integra la Interfaz de Programación de Aplicaciones (API) del Paquete de Desarrollo de Software de Infraestructura de Firma Sin Clave (KSI SDK) 101 para comunicarse con un servicio de firma y el servicio de extensión/verificación de una capa 200 de una puerta de enlace KSI. En KSI, a los datos a ser firmados se les hace hash mediante la aplicación 100 KSI y el valor se envía a la red 1 KSI para conseguir una firma para éstos En el contexto de la solicitud de firma, se observa que los datos en sí no se envían desde la aplicación 100 KSI, sólo el valor hash de éstos.

La aplicación 100 KSI puede usar cualquier tipo de cripto-librería para generar hash de datos o puede utilizar el apoyo de una función hash para del SDK KSI. Tal como se mencionó en la sección de antecedentes, para ser capaz de proporcionar la verificación de la integridad de unos datos, la aplicación KSI debe, según la técnica anterior, almacenar el testigo de firma y enlazarlo con el elemento de datos respectivo de alguna manera, por ejemplo con

metadatos, o almacenando el testigo de firma con el elemento de datos. Las presentes enseñanzas eliminan este requisito almacenando en su lugar la firma en la red 1 KSI y proporcionando la aplicación 100 KSI con una referencia a la firma.

Una segunda capa es la capa 200 de puerta de enlace. La puerta de enlace (GW) KSI proporciona una interfaz entre la aplicación 100 KSI y la red 1 KSI y maneja la firma de datos y las solicitudes de verificación/extensión desde la aplicación 100 KSI. La GW KSI implementa el primer nivel de un árbol hash de agregación global que agrega los hash de datos enviados desde la aplicación 100 KSI. Desde el GW KSI la solicitud de firma procede a una capa 300 de la red de agregación usando por ejemplo un protocolo basado en el Protocolo de Datagramas de Usuario (UDP). Las solicitudes de firma se envían a todos los agregadores 301, 302, 303 padre en un grupo que sirve a la GW KSI concreta.

Una tercera capa es la capa 300 de red de agregación. En KSI, se usa un componente de árbol hash denotado agregador y cada agregador actúa de manera asíncrona. En la figura 1, uno de dichos agregadores está indicado con el número de referencia 310. El agregador 310 toma valores hash como entrada, genera un valor hash raíz y envía el valor hash raíz a uno o más agregadores padre.

Un hash raíz del primer nivel de agregación global, también denotado agregación de nivel puerta de enlace, se envía a la red 300 de agregación. La red 300 de agregación comprende servidores de agregación distribuidos llamados agregadores, que comprenden una red de agregación distribuida global. Por razones de resistencia los agregadores se pueden desplegar como conjuntos de servidores. Cada miembro del conjunto recibe la misma solicitud de entrada del agregador hijo y ellos la ejecutan de manera independiente de una manera no sincronizada. La capa 300 de red de agregación es un segundo nivel de agregación.

Una cuarta capa es la capa de grupo de núcleo. El grupo 400 de núcleo comprende servidores geográficamente distribuidos ubicados en diferentes centros de datos e implementa un árbol hash de agregación de nivel superior, un árbol hash de calendario, una base de datos 401 de calendario y funciones 402 de publicación. Los valores hash de raíz superiores que son generados por la red 300 de agregación se almacenan en la base de datos 401 de calendario por ejemplo una por segundo. El árbol 403 hash de calendario es un tipo especial de árbol hash construido usando valores hash de raíz superiores del árbol hash de agregación superior (sólo uno ilustrado en la figura 1) como nodos hoja, una hoja por cada segundo. La función 402 de publicación es un conjunto de valores que comprenden código de publicación y tiempos de verificación de publicación publicados de manera periódica en medios impresos y electrónicos (por ejemplo, Internet) y almacenados en un archivo de publicación. El código de publicación es una cadena de caracteres usada para verificar una firma.

La Figura 2 ilustra una cadena hash en un proceso de verificación, y proporciona también una breve descripción de los principios de estructuración de las cadenas hash.

Un dispositivo KSI envía un hash de un activo de datos, por ejemplo un documento, al servicio KSI y recibe un testigo de firma de datos, que es la prueba de que los datos existen de una cierta forma en un momento dado y que la solicitud fue recibida a través de un punto de acceso específico. Todas las solicitudes recibidas se agregan juntas a un gran árbol hash. El testigo de firma contiene los datos para reconstruir una ruta a través del árbol hash comenzando desde un valor hash firmado (una hoja, por ejemplo X_3 de la figura 2) hasta el valor superior (X_{18} de la figura 2). Por ejemplo, dejar X_3 indica el hash de datos original e y un nuevo valor hash de los mismos datos de los cuales se ha de verificar la integridad. Entonces los nodos X_4, X_{12} y X_{58} son necesarios con la información de orden de concatenación para generar y_4, tal como se ilustra mediante la cadena hash en el lado derecho de la figura 2. Esto es, y primero se concatena con X_4 y se calcula un valor hash $y_2 = h(y|X_4|)$, que se usa como entrada al siguiente paso hash con X_{12}, obteniendo y_4 y así sucesivamente. Si $y_4 = X_{18}$, entonces y debe ser igual que X_3 y por tanto X_3 debe ser una parte del árbol hash original que prueba que los datos sobre los que se generó el hash X_3 no se han cambiado. Por tanto, si $y_4 = X_{18}$, entonces es seguro suponer que y_4 estaba en el árbol hash original (lado izquierdo de la figura 2).

Con referencia de nuevo a la figura 1, la estructura de red de agregación se ilustra en alto nivel. El árbol hash de agregación de nivel más bajo de la red 300 de agregación recibe hash raíz de un número de puertas de enlace, los establece como hojas de su árbol hash y los agrega al nuevo valor hash raíz en línea con los principios descritos anteriormente con referencia a la figura 2. El nuevo hash raíz se envía después al siguiente árbol hash de agregación de nivel superior (agregador padre) generado el siguiente hash raíz (3er nivel de agregación) respectivamente. El hash raíz superior generado por este nivel de agregación se envía después al grupo 400 de núcleo donde se realiza la agregación final de nivel superior.

Por razones de redundancia un hash raíz de un agregador hijo se envía a diversos agregadores padre en un grupo de agregadores. La primera respuesta de un agregador de nivel superior se acepta y se descartan las respuestas posteriores mediante un agregador hijo.

Tal como se comentó brevemente, las presentes enseñanzas sugieren en lugar de devolver la firma KSI real al dispositivo que posee los datos y usar una red KSI para protegerlos, devolver sólo una referencia a la firma mientras que la firma en sí se almacena en la red KSI.

Como opción adicional, y de acuerdo con las presentes enseñanzas, concretamente para dispositivos que no tienen suficiente capacidad de almacenamiento para almacenar sus datos, los datos en sí se pueden almacenar también junto con la firma en la red KSI, por ejemplo en una base de datos de firmas. Los datos en su nombre se pueden almacenar en la red KSI también en nombre de los dispositivos que tienen la capacidad de almacenamiento, por ejemplo por razones de seguridad.

Cuando la integridad de los datos necesita ser verificada, el dispositivo habilitado a KSI y/o su extremo de recepción de datos que se adapta también para usar KSI, puede usar la referencia a la firma para obtener la firma del almacenamiento de red KSI.

La figura 3 ilustra diversos aspectos de las presentes enseñanzas. Se ilustra una red 10 KSI que comprende un primer dispositivo 13, por ejemplo una puerta de enlace, recibir y manejar las solicitudes de firma KSI y las solicitudes de verificación. El primer dispositivo 13, en adelante ejemplificado por una puerta de enlace 14 KSI, se dispone para la comunicación con los clientes 12 KSI.

La red 10 KSI puede comprender uno o más almacenamientos 14 de datos. Dichos almacenamientos 14 de datos se pueden usar para permitir al operador 10 de la red KSI ofrecer servicios tales como almacenar el activo de datos en nombre de los clientes KSI, tal como se mencionó.

La red 10 KSI puede comprender también un número de servidores, máquinas virtuales y otros dispositivos de procesamiento, puertas de enlace, etc. para proporcionar una tecnología de firma de datos basada en hash, por ejemplo como la descrita por ejemplo con referencia a las figuras 1 y 2. La red 10 KSI puede por tanto comprender un grupo de núcleo, una red de agregación, y una red de verificadores (ejemplificada mediante una base de datos de calendario en la figura 3). Los detalles de la red 10 KSI, tales como el número de capas de los servidores de agregación, no son importantes para las presentes enseñanzas, y pueden ser según una red KSI convencional tal como la descrita con referencia a las figuras 1 y 2. La red 10 KSI puede comprender aún componentes convencionales adicionales, no ilustrados, tales como por ejemplo funciones de publicación.

Un segundo dispositivo 12 se ilustra también en la figura 3, que normalmente no es parte de la red 10 KSI. El segundo dispositivo 12 es un cliente KSI que busca los servicios proporcionados por la red 10 KSI. El cliente 12 KSI, también denotado cliente KSI adaptado o aplicación KSI, integra un Paquete de Desarrollo de Software (SDK) KSI 11 (Interfaz de Programación de Aplicaciones, API). El SDK KSI 11 proporciona una interfaz que es necesaria hacia un servicio KSI proporcionado por la red 10 KSI. Se envía una solicitud de firma de datos desde el cliente 12 KSI a través del SDK KSI 11, que proporciona también lógica de validación de datos y la interfaz de verificación necesaria hacia la red 10 KSI. El SDK KSI 11 es responsable de combinar los datos y la firma KSI, y de almacenar la firma KSI, o según las realizaciones de las presentes enseñanzas, de almacenar una referencia KSI a una firma.

El cliente 12 KSI puede por ejemplo ser cualquier tipo de dispositivo de Internet de las Cosas (IoT), esto es, cualquier tipo de dispositivo con conectividad a Internet, normalmente conectividad inalámbrica. El cliente 12 KSI puede comprender por ejemplo un así denominado dispositivo inteligente, que puede hacer referencia a cualquier dispositivo o aplicación conectada a Internet y/o a una red móvil. El cliente 12 KSI puede comprender un dispositivo de una red inteligente, es decir, una red eléctrica que utiliza tecnología de comunicaciones para recopilar (y actuar sobre) la información. Otros ejemplos de clientes 12 KSI comprenden implantes de monitorización del corazón, dispositivos sensores de redes de detección etc.

Se observa que el cliente 12 KSI no necesita tener conectividad a Internet, necesita ser capaz de comunicarse con la red 10 KSI, por ejemplo, con la puerta de enlace 13 KSI de la misma. El cliente 12 KSI puede comunicarse por ejemplo con la puerta de enlace 13 KSI sobre una red de comunicaciones móviles usando, por ejemplo, una interfaz de radio tal como GSM, WCDMA, Evolución a Largo Plazo (LTE), 3G, 4G, 5G o tecnologías de acceso tales como las del IEEE 802.11 (Red de Área Local Inalámbrica, WLAN) o las familias 802.16 (WiMAX) o aún otros sistemas de comunicación inalámbricos. El cliente 12 KSI puede comunicarse con la puerta de enlace 13 KSI de manera indirecta, a través de uno o más dispositivos intermedios (tal como se ejemplifica con referencia a la figura 4). Por ejemplo, si el cliente 12 KSI es un dispositivo sensor de una red de detección, puede comunicarse con una puerta de enlace de la red de detección, por ejemplo, sobre WLAN. La puerta de enlace de la red de detección se comunica entonces (de manera directa o indirecta a través de otros dispositivos) con la red 10 KSI, y puede proporcionar un primer nivel de agregación.

A continuación, se describen diversos aspectos de las presentes enseñanzas, con referencia aún a la figura 3.

Un cliente 12 KSI, por ejemplo, un dispositivo sensor, tiene algún activo de datos a proteger y usa la red 10 KSI para lograrlo. El activo de datos puede comprender cualquier pieza de información, por ejemplo, valores de medición, documentos u otros datos.

En la flecha indicada por A1, el SDK KSI 11 del cliente 12 KSI envía de este modo una solicitud de firma para solicitar una generación de firma. La firma que se ha de generar puede ser usada por el cliente 12 KSI para proteger el activo de datos. La solicitud de firma se envía a la puerta de enlace 13 KSI.

La solicitud de firma comprende, según las presentes enseñanzas, también una indicación de que el dispositivo 12 KSI quiere un almacenamiento de la firma basado en la red. Esto es, una indicación de que la red 10 KSI ha de almacenar la firma que genera para el dispositivo 12 KSI. Como ejemplo, se puede definir un indicador $I_{NW\_Almacenamiento}$, que indica si el dispositivo 12 KSI desea o no tener el almacenamiento de las firmas basado en la red. Si, por ejemplo, se establece el indicador $I_{NW\_Almacenamiento}$, entones el dispositivo 12 KSI quiere que la firma sea almacenada por la red 10 KSI, y si no se establece, entonces el dispositivo 12 KSI en sí almacena la firma que genera la red 10 KSI. La solicitud de firma puede comprender por tanto el indicador $I_{NW\_Almacenamiento}$ establecido de manera conveniente. El indicador $I_{NW\_Almacenamiento}$ define “almacenamiento de las firmas basado en la red solicitado”.

En las flechas indicadas por A2, la red 10 KSI procesa la solicitud de firma hacia arriba y hacia debajo de una manera convencional (esto es según la tecnología KSI existente), por ejemplo como se ha descrito con referencia a las figuras 1 y 2.

En la flecha indicada por A3, ya que se establece el nuevo indicador $I_{NW\_Almacenamiento}$, la puerta de enlace 13 KSI devuelve al dispositivo 12 KSI (en concreto al SDK KSI 11 del mismo) sólo una referencia a la firma de datos KSI generada, en lugar de devolver la firma de datos KSI completa. La referencia puede comprender un valor hash de la firma, una combinación de un valor hash y una marca de tiempo que indica el tiempo de agregación o sólo un valor hash.

Para proteger la integridad se puede calcular la referencia que el dispositivo KSI 12, o más bien el SDK KSI 11 del mismo, recibe y almacena, y almacenar un hash de la referencia en la puerta de enlace 13 KSI junto con la firma de datos KSI correspondiente.

En A4, el dispositivo 12 KSI (SDK KSI 11 del mismo) combina los datos originales con la referencia recibida y almacena la información.

En la flecha A5, se almacena la firma en la red 10 KSI, por ejemplo en la base de datos 14. La puerta de enlace 13 KSI se puede preconfigurar con un nuevo parámetro “Dirección de almacenamiento de red para firmas KSI”. Esta información se puede usar para determinar la ubicación del almacenamiento de la firma. La puerta de enlace 13 KSI puede, usando el parámetro sugerido “Dirección de almacenamiento de red para la firma KSI”, encontrar una firma correspondiente a una referencia.

En las flechas indicadas por A6a, A6b, se realiza la verificación de la firma. En la verificación de la firma, el dispositivo 12 KSI envía (flecha A6a) el hash de los datos originales y la referencia almacenada a la puerta de enlace 13. La puerta de enlace 13 KSI usa (flecha A6b) la referencia para correlacionarla con la firma completa, que se almacena en el almacenamiento 14 de datos, y desencadena una solicitud de verificación. En algunas realizaciones, la solicitud de verificación puede comprender una indicación de que la firma se almacena de manera central para indicar a la puerta de enlace 13 KSI que debería recuperar la firma. En otras realizaciones, el SDK KSI se puede adaptar de manera tal que la puerta de enlace 13 KSI entienda como manejar una firma KSI faltante en una solicitud de verificación, por ejemplo, para comprobar si es el caso de un almacenamiento basado en la red de la firma.

La puerta de enlace 13 KSI puede verificar también la integridad de la referencia tomando un hash de la referencia recibida y comparándolo con el hash almacenado de la referencia, o si el hash sólo se usa como una referencia, entonces la GW KSI comprueba que el valor hash recibido desde el dispositivo 12 KSI coincide con el almacenado dentro de la firma.

En la flecha A7, el resultado de la solicitud de verificación es devuelto al dispositivo 12 KSI.

Tal como se mencionó en la sección de antecedentes, la interfaz de radio de la medición inteligente podría no haber sido dimensionada para transmitir tamaños de paquetes grandes. Para evitar consumir recursos de radio enviando firmas de datos KSI de gran tamaño de un lado a otro sobre las interfaces de radio durante el proceso de lectura del medidor inteligente la firma de datos KSI se puede mantener en el almacenamiento de firmas basado en la red (por ejemplo, una base de datos de mediciones central) durante el proceso de lectura del medidor inteligente.

La figura 4 ilustra aspectos de las presentes enseñanzas implementados en un contexto de medidor inteligente. Dos medidores 12a, 12b inteligentes se ilustran y se pueden disponer para comunicarse sus datos, por ejemplo los valores de medición, sobre una red de comunicaciones inalámbrica. En la figura 4 la red de comunicaciones inalámbrica es ejemplificada mediante una red LTE. Los medidores 12a, 12b inteligentes pueden comprender de este modo una interfaz y otros medios para comunicarse con los nodos 15a, 15b de red, por ejemplo los eNodoB, de la red de comunicaciones inalámbrica. En otros ejemplos, los medidores 12a, 12b inteligentes de una red de detección se comunica sobre por ejemplo una WLAN con una puerta de enlace de la red de detección, que a su vez se comunica sobre por ejemplo la red LTE con la red 10 KSI, en concreto la puerta de enlace 13a KSI del mismo.

En la flecha A10, se inicia la lectura del medidor inteligente. La lectura puede ser iniciada de manera automática por un sistema que proporcionar una aplicación de medidor inteligente, por ejemplo, ser iniciada por una Puerta de Enlace de Acceso al Servicio (indicado en el número 22 de referencia). La lectura puede ser iniciada

alternativamente (flecha A10b) de manera manual mediante un administrador (indicado en el número 21 de referencia).

En la flecha A11, el medidor inteligente 12a (la aplicación de Medidor Inteligente del mismo) envía una solicitud de firma a la puerta de enlace 13a KSI. La solicitud de firma puede ir a través de un número de nodos antes de alcanzar la puerta de enlace 13a KSI, por ejemplo los nodos de la red de comunicaciones inalámbrica. Ejemplos de dichos nodos comprenden el eNodoB 15a, el Controlador de Red de Radio (RNC) 18, el nodo de soporte del Servicio GPRS (SGSN) 19, el nodo de soporte de la Puerta de enlace GPRS (GGSN) 20, etc.

La solicitud de firma comprende la nueva información: “solicitud de almacenamiento de firma basada en la red”, según las presentes enseñanzas. Esta nueva información, el indicador $I_{NW\_Almacenamiento}$ se describió con referencia a la figura 3, y la descripción es válida también para la presente realización y por lo tanto no se repite aquí.

En la flecha A12, la red 10 KSI procesa la solicitud de firma hacia arriba y hacia abajo como en una tecnología existente, esto es genera la firma de datos KSI, que obtiene la puerta de enlace 13a KSI.

En la flecha A13, la firma de datos KSI recibida puede, como se solicita, ser almacenada en un almacenamiento de firmas basado en la red. En el caso ilustrado la firma de datos KSI generada se almacena en una Base de Datos 14a de Mediciones Central.

En la flecha A14, ya que se establece el nuevo indicador $I_{NW\_Almacenamiento}$, la puerta de enlace 13a KSI devuelve al medidor 12a inteligente sólo una referencia a la firma en lugar de devolver la firma de datos KSI completa. De este modo se reduce el consumo de recursos de la interfaz de radio limitados en comparación al envío de la firma de datos KSI completa.

En la flecha A15 el medidor 12a inteligente, en concreto la aplicación KSI del mismo, combina la lectura actual (que es el activo de datos a proteger) con la referencia recibida y los envía, por ejemplo, como un valor hash, a la Puerta de Enlace 22 de Acceso al Servicio.

En la flecha A16, la puerta de enlace 13a KSI lee la firma de datos KSI desde el almacenamiento 14a de firmas, y compara el hash recibido con la firma de datos KSI leída.

En la flecha A17, si el hash recibido desde el medidor 12a inteligente y la firma de datos KSI coinciden, los resultados de lectura del medidor y la firma de datos KSI se almacenan en la Base de Datos 14a de Medición Central.

El medidor 12b inteligente más inferior puede usar también el esquema anterior para proteger su activo. Mientras que el medidor 12a inteligente más superior usa nodos de una red de núcleo del Servicio de Radio de Paquetes General (GPRS) para comunicarse con la Puerta de Enlace 22 de Acceso al Servicio, el medidor 12b inteligente más inferior se comunica con la Puerta de Enlace 22 de Acceso al Servicio usando la GW PDN 17 como la puerta de enlace de terminación hacia la red de datos de paquetes (PDN).

A continuación, se resumen la solicitud de firma y la solicitud de verificación.

Solicitud de firma

El propietario de los datos, por ejemplo el medidor 12a, 12b inteligente posee un activo de datos (por ejemplo, un valor de medición) a ser protegido. El dispositivo 12, 12a, 12b calcula un hash de los datos:

$$A = \text{hash}(\text{datos})$$

El valor de A se envía a la puerta de enlace 13, 13a KSI, en la solicitud de firma, incluyendo en la solicitud el indicador según las presentes enseñanzas, esto es indicando que se desea la firma almacenada en la red. La puerta de enlace 13, 13a KSI genera una firma B relacionada con A:

$$B = \text{firmar}(A)$$

A continuación, la puerta de enlace 13, 13a KSI genera una referencia C a la firma B:

$$C = \text{ref}(B)$$

Tal como se ha descrito, C puede ser un hash del activo de datos (C = A), C puede ser un hash de la firma, o un hash de los datos + una marca de tiempo, etc.

La puerta de enlace 13, 13a KSI puede calcular entonces un hash de la referencia C:

$$D = \text{hash}(C)$$

La puerta de enlace 13, 13a KSI almacena A, B, C y D.

La puerta de enlace 13, 13a devuelve la referencia C al dispositivo 12, 12a, 12b.

El dispositivo 12, 12a, 12b almacena el activo de datos y la referencia C recibida.

Solicitud de verificación:

El dispositivo 12, 12a, 12b calcula un hash A de los datos que se han de verificar:

A = hash(datos)

El dispositivo 12, 12a, 12b entonces envía los valores A y C a la puerta de enlace 13, 13a KSI en una solicitud de verificación.

La puerta de enlace 13, 13a KSI calcula un hash D' de la referencia C recibida y un hash A:

D' = hash(A, C)

A continuación, la puerta de enlace 13, 13a KSI usa la referencia C para encontrar A, B, C y D en el almacenamiento y comprueba que el valor D' calculado coincide con el calor D almacenado y que A en la solicitud de verificación coincide con el A en el almacenamiento.

Finalmente, la puerta de enlace 13, 13a verifica el hash A recibido contra la firma B almacenada.

Las diversas realizaciones y características que se han descrito se pueden combinar de diferentes maneras, ejemplos de las cuales se dan a continuación con referencia primero a la figura 5.

La figura 5 ilustra un diagrama de flujo sobre los pasos de una realización de un método en un primer dispositivo de acuerdo con las presentes enseñanzas. El método 30 de manejo de una firma de datos basada en árbol hash se puede realizar en un primer dispositivo 13, 13a tal como, por ejemplo, una puerta de enlace de una red 10 de infraestructura de firma sin clave.

El método 30 comprende recibir en 31, desde un segundo dispositivo 12, 12a, 12b (por ejemplo, un dispositivo de una red de detección), una solicitud de generación de firma de datos. La solicitud comprende una indicación sobre el tipo de almacenamiento de una firma de datos generada.

El método 30 comprende generar en 32, en respuesta a la solicitud de generación de firma de datos, una firma B de datos usando un método de firma basado en árbol hash.

El método 30 comprende proporcionar en 33, al segundo dispositivo 12, 12a, 12b, una referencia C, C = ref(B), a la firma B de datos generada, en donde la firma B de datos generada se puede obtener por medio de la referencia C.

En una realización, se establece la indicación del tipo de almacenamiento para indicar el almacenamiento mediante el primer dispositivo 13, 13a y el método 30 comprende almacenar la firma B de datos generada. El almacenamiento de la firma de datos generada puede comprender el primer dispositivo 13, 13a que almacena la firma de datos en un almacenamiento de datos que está disponible dentro del primer dispositivo 13, 13a o almacenar la firma de datos en un almacenamiento de datos externo al primer dispositivo 13, 13a accesible al primer dispositivo 13, 13a.

En una realización, el método 30 comprende:

- recibir, desde el segundo dispositivo 12, 12a, 12b, una solicitud de verificación de firma de datos, comprendiendo la solicitud un hash A de un activo de datos (A = hash(activo de datos)), y la referencia C a la firma B de datos generados (C = ref(B)).

- correlacionar la referencia C con una firma B de datos almacenados, y

- verificar la integridad del activo de datos tras correlacionar la referencia C con la firma B de datos almacenados.

En una variación de la realización anterior, la correlación comprende:

- calcular un valor hash D' (D' = hash(A, C)) del hash A recibido del activo de datos (A = hash (activo de datos)) y la referencia C a la firma B de datos generada (C = ref(B)),

- recuperar usando la referencia C recibida lo siguiente desde un almacenamiento de datos: un valor hash A del activo de datos y un valor D hash de la referencia C, y

- comparar el valor D' hash calculado con el valor D hash recuperado y comparar el valor A hash recibido del activo de datos con el valor A hash recuperado del activo de datos.

En una realización, el proporcionar en 33 comprende:

- calcular un valor E hash de la referencia C a la firma B de datos generada (E = hash(C), C=ref(B)),

- almacenar el valor E hash calculado de la referencia C (E = hash(C)) y almacenar también la firma B de datos generada, y

- proporcionar, al segundo dispositivo 12, 12a, 12b el valor E hash de la referencia C a la firma B de datos generada.

Proporcionando el valor E hash de la referencia C, en lugar de proporcionar la referencia C, se proporciona una seguridad incluso mejorada. Ya que el primer dispositivo 13, 13a almacena el valor E hash así como la correspondiente referencia C y la correspondiente firma B de datos, la firma B de datos puede ser fácilmente verificada por el primer dispositivo 13, 13a.

En una variación de la realización anterior, el método 30 comprende:

- recibir, desde el segundo dispositivo 12, 12a, 12b una solicitud de verificación de firma de datos, la solicitud comprende un hash A de un activo de datos (A = hash(activo de datos)) y el valor E hash de la referencia C a la firma B de datos generada (E = hash(C), C = ref(B)),

- correlacionar el valor E hash recibido de la referencia C (E = hash(C)) con un valor E' hash almacenado de la referencia C a la firma B de datos generada, y

- verificar la integridad del activo de datos tras correlacionar de manera exitosa el valor E hash recibido de la referencia C con el valor E' hash almacenado de la referencia a la firma de datos generada. Esto es, si la E (E = hash(C)), que el primer dispositivo 13, 13a recibe es igual al E' (E' = hash(C)), que ha almacenado anteriormente (en concreto cuando la haya calculado), entonces la integridad del activo de datos se verifica.

En diversas realizaciones, el método 30 comprende almacenar la firma B de datos generada junto con un activo de datos para el que la firma B de datos fue generada.

En diversos dispositivos el primer dispositivo 13, 13a comprende un dispositivo de una infraestructura 10 de firma sin clave, KSI, y la firma de datos basada en árbol hash comprende una firma KSI.

La figura 6 ilustra de manera esquemática un primer dispositivo y medio para implementar las realizaciones del método de acuerdo con las presentes enseñanzas.

El primer dispositivo 13, 13a comprende un procesador 40 que comprende cualquier combinación de uno o más de entre una unidad central de procesamiento (CPU), un multiprocesador, un microcontrolador, un procesador digital de señales (DSP), un circuito integrado específico de aplicación etc. capaz de ejecutar las instrucciones de software almacenadas en una memoria 41 que puede por tanto ser un producto 41 de programa informático. El procesador 40 se puede configurar para ejecutar cualquiera de las diversas realizaciones del método para el caso que se describe en relación con la figura 5.

La memoria 41 puede ser cualquier combinación de memoria de lectura y escritura (RAM) y memoria de sólo lectura (ROM), memoria Flash, cinta magnética, Disco Compacto (CD)-ROM, disco versátil digital (DVD), disco Blu-ray etc. La memoria 41 puede comprender también un almacenamiento persistente, que, por ejemplo, puede ser cualquier memoria magnética, memoria óptica, memoria de estado sólido o incluso memoria montada de manera remota única o en combinación

El primer dispositivo 13, 13a comprende también un dispositivo 43 de entrada/salida (indicado mediante I/O en la figura 6) para comunicarse con otras entidades, por ejemplo los diversos clientes que desean tener una firma generada verificada. Si el primer dispositivo 13, 13a es parte de la red 10 KSI, entonces el dispositivo 43 de entrada/salida se puede usar para comunicarse con otras entidades dentro de la red KSI. El primer dispositivo 13, 13a puede también, por medio del dispositivo 43 de entrada/salida, comunicarse con el segundo dispositivo 12, 12a, 12b. Dicho dispositivo 43 de entrada/salida del primer dispositivo 13, 13a puede comprender una interfaz de comunicación inalámbrica (por ejemplo, una interfaz de radio) y/o una interfaz de comunicación por cable.

El primer dispositivo 13, 13a puede comprender también circuitería de procesamiento adicional, indicada de manera esquemática en el número 44 de referencia para implementar las diversas realizaciones según las presentes enseñanzas.

Las presentes enseñanzas proporcionan programas 42 informáticos para el primer dispositivo 13, 13a. El programa 42 informático comprende código de programa informático, que, al ser ejecutado en al menos un procesador 40 del primer dispositivo 13, 13a, provoca que el primer dispositivo 13, 13a realice el método 30 según cualquiera de las realizaciones descritas del mismo.

La presente descripción también abarca los productos 41 de programa informático que comprenden un programa 42 informático para implementar las realizaciones del método tal como se describe, y un medio legible por ordenador sobre el que el programa 42 informático se almacena. El producto 41 de programa informático puede, tal como se mencionó anteriormente, ser una combinación de memoria de acceso aleatorio (RAM) o memoria de sólo lectura (ROM), memoria Flash, cinta magnética, Disco Compacto (CD)-ROM, disco versátil digital (DVD), disco Blu-ray etc.

Se proporciona un primer dispositivo 13, 13a para manejar una firma de datos basada en árbol hash. El primer dispositivo 13, 13a se configura para:

- recibir desde un segundo dispositivo 12, 12a, 12b, una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de una firma de datos generada,

- generar, en respuesta a la solicitud de generación de la firma de datos, la firma B de datos que usa un método de firma de datos basado en árbol hash, y

- proporcionar, al segundo dispositivo 12, 12a, 12b, una referencia C a la firma de datos generada, en donde la firma de datos generada se puede obtener por medio de la referencia C.

El primer dispositivo 13, 13a se puede configurar para realizar los pasos anteriores por ejemplo comprendiendo uno o más procesadores 40 y una memoria 41, la memoria 41 contiene instrucciones ejecutables por el procesador 40, a través de las cuales el primer dispositivo 13, 13a es operativo para realizar los pasos. En caso de varios procesadores 40 (no ilustrados) se pueden configurar para realizar todos los pasos del método 30 o sólo parte de los pasos.

En una realización, la indicación del tipo de almacenamiento se establece para indicar el almacenamiento por el primer dispositivo 13, 13a y el primer dispositivo 13, 13a se configura para almacenar la firma de datos generada.

En una realización, el primer dispositivo 13, 13a se configura para:

- recibir, desde el segundo dispositivo 12, 12a, 12b, una solicitud de verificación de firma de datos, comprendiendo la solicitud un hash A de un activo de datos y la referencia C a la firma de datos generada,

- correlacionar la referencia C con la firma B de datos almacenada, y

- verificar la integridad del activo de datos tras la correlación exitosa de la referencia C con la firma B de datos almacenada.

En una variación de la realización anterior, el primer dispositivo 13, 13a se configura para correlacionarse mediante:

- el cálculo de un valor D' de hash del hash A recibido del activo de datos y la referencia C a la firma de datos generada,

- la recuperación mediante el uso de la referencia C recibida de lo siguiente desde un almacenamiento de datos: un valor hash A del activo de datos y un valor D hash de la referencia C, y

- la comparación del valor D' hash calculado con el valor D recuperado y la comparación del valor A hash recibido del activo de datos con el valor A hash recuperado del activo de datos.

En diversas realizaciones, el primer dispositivo 13, 13a se configura para proporcionar mediante:

- el cálculo de un valor E hash de la referencia C a la firma B de datos generada.

- el almacenamiento del valor E hash calculado de la referencia C y la firma B de datos generada, y

- la proporción, al segundo dispositivo 12, 12a, 12b, del valor E hash de la referencia C a la firma B de datos generada.

En una variación de la realización anterior, el primer dispositivo 13, 13a se configura para:

- recibir, desde el segundo dispositivo 12, 12a, 12b, una solicitud de verificación de firma de datos, comprendiendo la solicitud un valor hash A de un activo de datos y el valor E hash de la referencia C a la firma B de datos generada.

- correlacionar el valor E hash recibido de la referencia C con un valor E' hash almacenado de la referencia a la firma de datos generada, y

- verificar la integridad del activo de datos tras la correlación exitosa del valor E hash recibido de la referencia C con el valor E hash almacenado de la referencia a la firma de datos generada.

En diversas realizaciones, el primer dispositivo 13, 13a se configura para almacenar la firma B de datos generada junto con un activo de datos para el que la firma B de datos fue generada.

En diversas realizaciones, el primer dispositivo 13, 13a comprende un dispositivo de una infraestructura 10 de firma sin clave, KSI, y en donde la firma de datos basada en árbol hash comprende una firma KSI.

La figura 7 ilustra un primer dispositivo que comprende módulos de función/módulos de software para implementar las realizaciones de las presentes enseñanzas.

En un aspecto, se proporcionan los medios, por ejemplo los módulos o unidades de función, que se pueden implementar usando instrucciones de software tales como un programa informático que se ejecuta en un procesador y/o que usa hardware, tal como circuitos integrados específicos de aplicación, matrices de puertas programables en campo, componentes lógicos discretos, etc., o cualquier combinación de los mismos.

Se proporciona un primer dispositivo para manejar una firma de datos basada en árbol hash. El primer dispositivo comprende una primera unidad 51 para recibir, desde un segundo dispositivo, una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de la firma de datos generada. Dicha primera unidad 51 puede comprender por ejemplo circuitería de procesamiento para recibir dicha solicitud y/o una interfaz de comunicación (por ejemplo, las unidades 44 y/o 43 descritas con referencia a la figura 6).

El primer dispositivo comprende una segunda unidad 52 para generar, en respuesta a la solicitud de generación de firma de datos, una firma de datos que usa un método de firma de datos basado en árbol hash. Dicha segunda unidad 52 puede comprender por ejemplo circuitería de procesamiento (por ejemplo, la unidad 44 de la figura 6) adaptada para generar firmas de datos usando un método de firma de datos basado en árbol hash.

El primer dispositivo comprende una tercera unidad 53 para proporcionar, al segundo dispositivo, una referencia a la firma de datos generada, en donde la firma de datos generada se puede obtener por medio de la referencia. Dicha tercera unidad 53 puede por ejemplo comprender circuitería de procesamiento para transmitir y/o una interfaz de comunicación (por ejemplo, las unidades 44 y/o 43 descritas con referencia a la figura 6).

La figura 8 ilustra un diagrama de flujo sobre los pasos de una realización de un método en un segundo dispositivo de acuerdo con las presentes enseñanzas.

El método 50 de manejo de una firma de datos basada en árbol hash se puede realizar en un segundo dispositivo 12, 12a, 12b, tal como, por ejemplo, un dispositivo inteligente. El método 50 comprende el envío 51, a un primer dispositivo 13, 13a, de una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de la firma de datos generada.

El método 50 comprende recibir en 52, desde el primer dispositivo 13, 13a, una referencia C a la firma B de datos generada en respuesta a la solicitud, en donde la firma B de datos generada se puede obtener por medio de la referencia C (C = ref(B)).

El método 50 permite el uso eficiente de los escasos recursos de la interfaz de radio incluso aunque la tecnología KSI sea usada por el primer dispositivo 13, 13a si se conecta, por ejemplo, a una red móvil (por ejemplo, 3G, 4G) o a tecnologías de acceso tales como las de las familias IEEE 802.11 (Red de Área Local Inalámbrica, WLAN) o 802.16 (WiMAX) u otros sistemas de comunicación inalámbricos. Además, ya que la firma se almacena fuera del segundo dispositivo, por ejemplo siendo un dispositivo inteligente, la firma es menos vulnerable a accesos ilegales. Los dispositivos inteligentes a menudo residen en áreas no seguras mientras que el almacenamiento de la firma se ubica normalmente dentro de una zona segura. Aún además, la verificación de la integridad de los datos a ser hecha por el extremo que recibe los datos resulta más simple cuando la firma se almacena fuera del dispositivo inteligente. En concreto, si el extremo receptor de datos hace la verificación, la firma no necesita ser enviada al extremo receptor de datos, pero en su lugar puede acceder de manera directa a ésta desde el almacenamiento basado en la red (central).

En una realización, el método 50 comprende:

- enviar, al primer dispositivo 13, 13a, una solicitud de verificación de firma de datos, comprendiendo la solicitud un hash de un activo de datos y la referencia a la firma de datos, y
- recibir, en respuesta a la solicitud, una verificación de la firma de datos.

En una realización, el método 50 comprende recibir desde el primer dispositivo 13, 13a un valor hash calculado como la referencia a la firma de datos generada.

La figura 9 ilustra de manera esquemática un segundo dispositivo 12, 12a, 12b y los medios para implementar las realizaciones del método de acuerdo con las presentes enseñanzas.

El segundo dispositivo 12, 12a, 12b comprende un procesador 60 que comprende cualquier combinación de uno o más de entre una unidad de procesamiento central (CPU), un multiprocesador, un microcontrolador, un procesador digital de señales (DSP), un circuito integrado para aplicaciones específicas etc. capaz de ejecutar instrucciones de software almacenadas en una memoria 61 que puede ser por tanto un producto 61 de programa informático. El

procesador 60 se puede configurar para ejecutar cualquier de las diversas realizaciones del método por ejemplo tal como se describe en relación a la figura 8.

La memoria 61 puede ser cualquier combinación de memoria de lectura y escritura (RAM) y memoria de sólo lectura (ROM), memoria Flash, cinta magnética, Disco Compacto (CD)-ROM, disco versátil digital (DVD), disco Blu-ray etc. La memoria 61 puede comprender también almacenamiento persistente, que, por ejemplo, puede ser cualquier memoria magnética, memoria óptica, memoria de estado sólido o incluso memoria montada de manera remota única o en combinación.

El segundo dispositivo 12, 12a, 12b comprende también un dispositivo 63 de entrada/salida (indicado por I/O en la figura 9) para comunicarse con otras entidades. Por ejemplo, en caso del segundo dispositivo 12, 12a, 12b es un dispositivo sensor y parte de una red de detección, el segundo dispositivo 12, 12a, 12b puede comunicarse con otras entidades dentro de la red de detección. El segundo dispositivo 12, 12a, 12b puede también, por medio del dispositivo 63 de entrada/salida, comunicarse con el primer dispositivo 13, 13a. Dicho dispositivo 63 de entrada/salida del segundo dispositivo 12, 12a, 12b puede comprender una interfaz de comunicación inalámbrica (por ejemplo, una interfaz de radio) y/o una interfaz de comunicación por cable.

El segundo dispositivo 12, 12a, 12b puede comprender también circuitería de procesamiento adicional, indicada de manera esquemática en el número 64 de referencia, para implementar las diversas realizaciones según las presentes enseñanzas.

Las presentes enseñanzas proporcionan programas 62 informáticos para el segundo dispositivo 12, 12a, 12b. El programa 62 informático comprende código de programa informático, que, al ser ejecutado en al menos un procesador 60 del segundo dispositivo 12, 12a 12b provoca que el segundo dispositivo 12, 12a, 12b realice el método 50 según cualquiera de las realizaciones descritas del mismo.

La presente descripción abarca también los productos 61 de programa informático que comprenden un programa 62 informático para implementar las realizaciones del método tal como se describe, y medios legibles por ordenador sobre los que se almacena el programa 62 informático. El producto 61 de programa informático puede, tal como se mencionó anteriormente, ser cualquier combinación de una memoria de acceso aleatorio (RAM) o una memoria de sólo lectura (ROM), una memoria Flash, una cinta magnética, un Disco Compacto (CD)-ROM, un disco Versátil digital (DVD), un disco Blu-ray etc.

Un segundo dispositivo 12, 12a, 12b se proporciona para manejar una firma de datos basada en árbol hash. El segundo dispositivo 12, 12a, 12b se configura para:

- enviar, a un primer dispositivo 13, 13a, una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de la firma de datos generada, y

- recibir, desde el primer dispositivo 13, 13a, una referencia a una firma de datos generada en respuesta a la solicitud, en donde la firma de datos generada se puede obtener por medio de la referencia.

El segundo dispositivo 12, 12a, 12b se puede configurar para realizar los pasos anteriores, por ejemplo comprendiendo uno o más procesadores 60 y una memoria 61, conteniendo la memoria 61 instrucciones ejecutables por el procesador 60, a través del cual el segundo dispositivo 12, 12a, 12b es operativo para realizar los pasos. En el caso de varios procesadores 60 (no ilustrado) se pueden configurar para realizar todos los pasos del método 50 o sólo parte de los pasos.

En una realización, el segundo dispositivo 12, 12a, 12b se configura para:

- enviar, a un primer dispositivo 13, 13a, una solicitud de generación de firma de datos, comprendiendo la solicitud un hash de un activo de datos y la referencia a la firma de datos, y

- recibir, en respuesta a la solicitud, una verificación de la firma de datos.

En una realización, el segundo dispositivo 12, 12a, 12b se configura para recibir, desde el primer dispositivo 13, 13a, un valor hash calculado como la referencia a la firma de datos generada. Calculando un hash de la referencia a la firma de datos, la referencia es también protegida en su integridad proporcionando aún una integridad de datos aumentada.

La figura 10 ilustra un segundo dispositivo que comprende módulos de función/módulos de software para implementar realizaciones de las presentes enseñanzas.

En un aspecto, se proporcionan los medios, por ejemplo los módulos o unidades de función, que se pueden implementar usando instrucciones de software tales como un programa informático que se ejecuta en un procesador y/o usando hardware tal como circuitos integrados para aplicaciones específicas, matrices de puertas programables en campo, componentes de lógica discreta etc., o cualquier combinación de los mismos.

Se proporciona un segundo dispositivo para manejar una firma de datos basada en árbol hash. El segundo dispositivo comprende una primera unidad 71 para enviar, a un primer dispositivo, una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de la firma de datos generada. Dicha primera unidad 71 puede por ejemplo comprender circuitería de procesamiento para enviar dicha solicitud y/o una interfaz de comunicación (por ejemplo, las unidades 64 y/o 63 descritas con referencia a la figura 9).

El segundo dispositivo comprende una segunda unidad 72 para recibir, desde el primer dispositivo, una referencia a una firma de datos generada en respuesta a la solicitud, en donde la firma de datos generada se puede obtener por medio de la referencia. Dicha segunda unidad 72 puede por ejemplo comprender la circuitería de procesamiento para recibir dicha referencia y/o una interfaz de comunicación (por ejemplo, las unidades 64 y/o 63 descritas con referencia a la figura 9).

La invención se ha descrito principalmente en la presente memoria con referencia a unas pocas realizaciones. Sin embargo, como será apreciado por una persona experta en la técnica, otras realizaciones distintas de las concretas descritas en la presente memoria son igualmente posibles dentro del alcance de la invención, tal como se define por las reivindicaciones de patente adjuntas.

REIVINDICACIONES

1. Un método (30) de manejo de una firma de datos basada en árbol hash, estando el método (30) realizado en un primer dispositivo (13, 13a) y comprendiendo:

- recibir (31), desde un segundo dispositivo (12, 12a, 12b), una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de una firma de datos generada.

- generar (32), en respuesta a la solicitud de generación de firma de datos, una firma B de datos que usa un método de firma de datos basado en árbol hash, y

- proporcionar (33), al segundo dispositivo (12, 12a, 12b), en base a la indicación del tipo de almacenamiento, una referencia C a la firma B de datos generada, en donde la firma B de datos generada se puede obtener por medio de la referencia C.

2. El método (30) tal como se reivindica en la reivindicación 1, en donde la indicación del tipo de almacenamiento se establece para indicar el almacenamiento por parte del primer dispositivo (13, 13a) y en donde el método (30) comprende el almacenamiento de la firma B de datos generada.

3. El método (30) tal como se reivindica en la reivindicación 1 o 2, que comprende:

- recibir, desde el segundo dispositivo (12, 12a, 12b), una solicitud de verificación de firma de datos, comprendiendo la solicitud un hash A de un activo de datos y la referencia C a la firma de datos generada,

- correlacionar la referencia C con una firma B de datos almacenada, y

- verificar la integridad del activo de datos tras la correlación exitosa de la referencia C con la firma B de datos almacenada.

4. El método (30) tal como se reivindica en la reivindicación 3, en donde la correlación comprende:

- calcular un valor D' hash del A hash recibido del activo de datos y la referencia C a la firma de datos generada,

- recuperar usando la referencia C recibida lo siguiente desde un almacenamiento de datos: un valor A hash del activo de datos y un valor D hash de la referencia C, y

- comparar el valor D hash calculado con el valor D hash recuperado y comparar el valor A hash recibido del activo de datos con el valor A hash recuperado del activo de datos.

5. El método (30) tal como se reivindica en la reivindicación 1 o 2, en donde el proporcionar (33) comprende:

- calcular un valor E hash de la referencia C a la firma B de datos generada,

- almacenar el valor E hash calculado de la referencia C y la firma B de datos generada, y

- proporcionar, al segundo dispositivo (12, 12a, 12b), el valor E hash de la referencia C a la firma B de datos generada.

6. El método (30) tal como se reivindica en la reivindicación 5 que comprende:

- recibir, desde el segundo dispositivo (12, 12a, 12b), una solicitud de verificación de firma de datos, comprendiendo la solicitud un hash A de un activo de datos y el valor E hash de la referencia C a la firma B de datos generada,

- correlacionar el valor E hash recibido de la referencia C con un valor E' hash almacenado de la referencia a la firma de datos generada, y

- verificar la integridad del activo de datos tras la correlación exitosa del valor E hash recibido de la referencia C con el valor E' hash almacenado de la referencia a la firma de datos generada.

7. Un programa (42) informático para un primer dispositivo (13, 13a) para manejar las firmas de datos basadas en árbol hash, comprendiendo el programa (42) informático código de programa informático, que, al ser ejecutado en al menos un procesador en el primer dispositivo (13, 13a) provoca que el primer dispositivo (13, 13a) realice el método (30) según cualquier de las reivindicaciones 1-6.

8. Un producto (41) de programa informático que comprende un programa (42) informático tal como se reivindica en la reivindicación 7 y unos medios legibles por ordenador en los que se almacena el programa (42) informático.

9. Un primer dispositivo (13, 13a) para manejar una firma de datos basada en árbol hash, estando el primer dispositivo (13, 13a) configurado para:

- recibir, desde un segundo dispositivo (12, 12a, 12b), una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de la firma de datos generada,

- generar, en respuesta a la solicitud de generación de la firma de datos, la firma B de datos usando un método de firma de datos basado en árbol hash, y

- proporcionar, al segundo dispositivo (12, 12a, 12b), en base a la indicación del tipo de almacenamiento, una referencia C a la firma de datos generada, en donde la firma de datos generada se puede obtener por medio de la referencia C.

10. El primer dispositivo (13, 13a) tal como se reivindica en la reivindicación 9, en donde la indicación sobre el tipo de almacenamiento se establece para indicar el almacenamiento por parte del primer dispositivo (13, 13a) y en donde el primer dispositivo (13, 13a) se configura para almacenar la firma de datos generada.

11. El primer dispositivo (13, 13a) tal como se reivindica en la reivindicación 9 o 10, configurado para:

- recibir, desde el segundo dispositivo (12, 12a, 12b), una solicitud de verificación de firma de datos, comprendiendo la solicitud un hash A de un activo de datos y la referencia C a la firma de datos generada,

- correlacionar la referencia C con una firma B de datos almacenada, y

- verificar la integridad del activo de datos tras correlacionar exitosamente la referencia C con la firma B de datos almacenada.

12. El primer dispositivo (13, 13a) tal como se reivindica en la reivindicación 11, configurado para correlacionarse mediante:

- el cálculo de un valor D' hash del A hash recibido del activo de datos y la referencia C a la firma de datos generada,

- la recuperación usando la referencia C recibida de lo siguiente desde un almacenamiento de datos: un valor A hash del activo de datos y un valor D hash de la referencia C, y

- la comparación del valor D' hash calculado con el valor D hash recuperado y comparar el valor A hash recibido del activo de datos con el valor A hash recuperado del activo de datos.

13. El primer dispositivo (13, 13a) tal como se reivindica en la reivindicación 9 o 10, configurado para comparar mediante:

- el cálculo de un valor E hash de la referencia C a la firma B de datos generada,

- almacenar el valor E hash calculado de la referencia C y la firma B de datos, y

- proporcionar, al segundo dispositivo (12, 12a, 12b), el valor E hash de la referencia C a la firma B de datos generada.

14. El primer dispositivo (13, 13a) tal como se reivindica en la reivindicación 13, configurado para:

- recibir, desde el segundo dispositivo (12, 12a, 12b) una solicitud de verificación de firma de datos que comprende un A hash de un activo de datos y el valor E hash de la referencia C a la firma B de datos generada,

- correlacionar el valor E hash recibido de la referencia C con un valor E' hash almacenado de la referencia a la firma de datos generada, y

- verificar la integridad del activo de datos tras la correlación exitosa del valor E hash recibido de la referencia C con el valor E' hash almacenado de la referencia a la firma de datos generada.

15. El primer dispositivo (13, 13a) tal como se reivindica en cualquiera de las reivindicaciones 9-14, configurado para almacenar la firma B de datos generada junto con un activo de datos para el que la firma B de datos fue generada.

16. El primer dispositivo (13, 13a) tal como se reivindica en cualquiera de las reivindicaciones 9-15, que comprende un dispositivo de una infraestructura (10) de firma sin clave, KSI, y en donde la firma de datos basada en árbol hash comprende una firma KSI.

17. Un método (50) de manejo de una firma de datos basada en árbol hash, estando el método (50) realizado en un segundo dispositivo (12, 12a, 12b) y comprendiendo:

- enviar (51), a un primer dispositivo (13, 13a), una solicitud de generación de firma de datos, comprendiendo la solicitud una indicación del tipo de almacenamiento de la firma de datos generada, que indica una solicitud de una referencia a una firma de datos, y

- recibir (52), desde el primer dispositivo (13, 13a), una referencia a una firma de datos generada en respuesta a la solicitud, en donde la firma de datos generada se puede obtener por medio de la referencia.

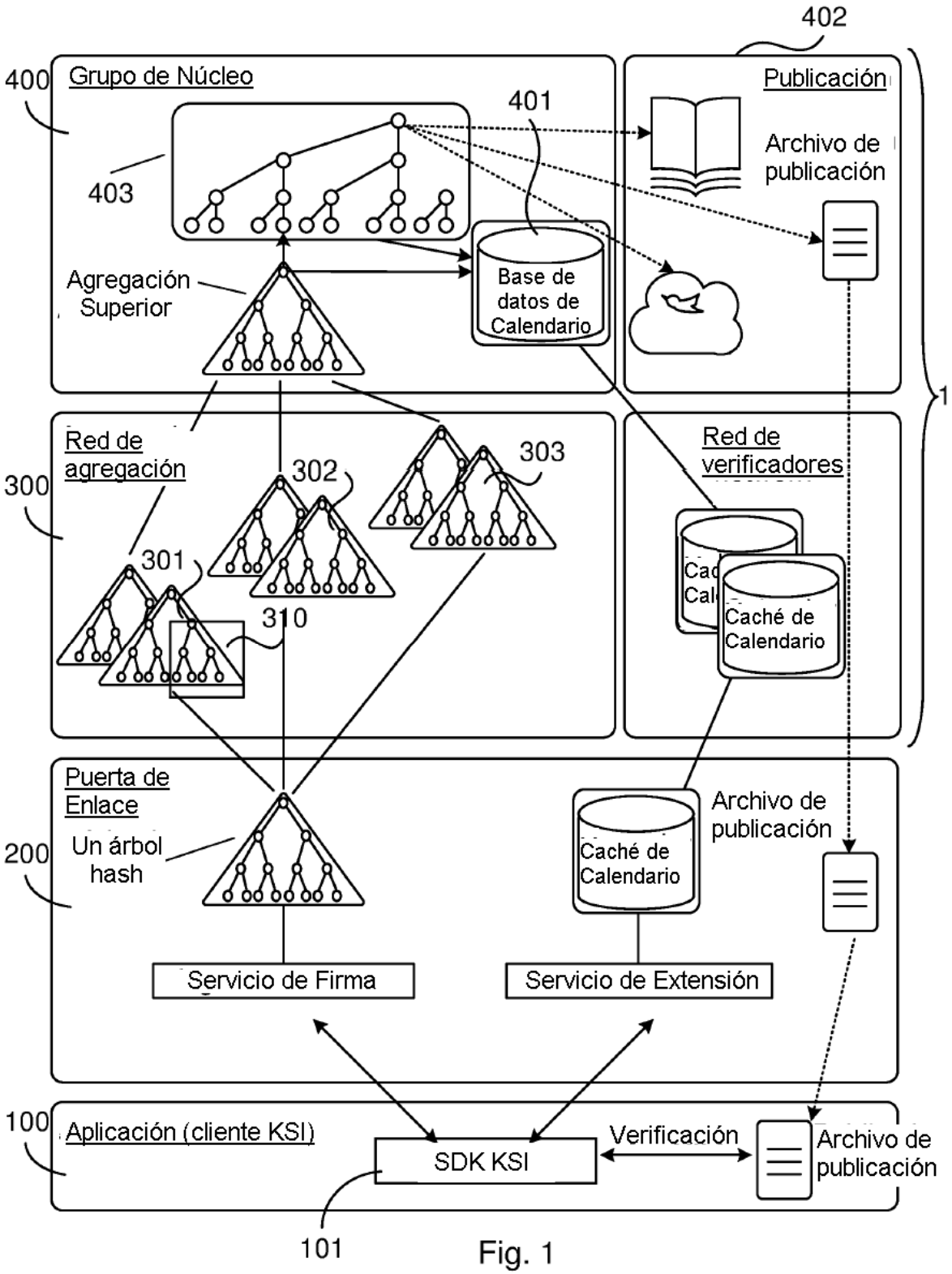
402
400
Grupo de Núcleo
401
Publicación
Archivo de publicación
403
Agregación Superior
Base de datos de Calendario
1
Red de agregación
300
302
303
301
310
Red de verificadores
Caché de Calendario
Puerta de Enlace
Un árbol hash
200
Caché de Calendario
Archivo de publicación
Servicio de Firma
Servicio de Extensión
100
Aplicación (cliente KSI)
SDK KSI
Verificación
Archivo de publicación
101

Fig. 1

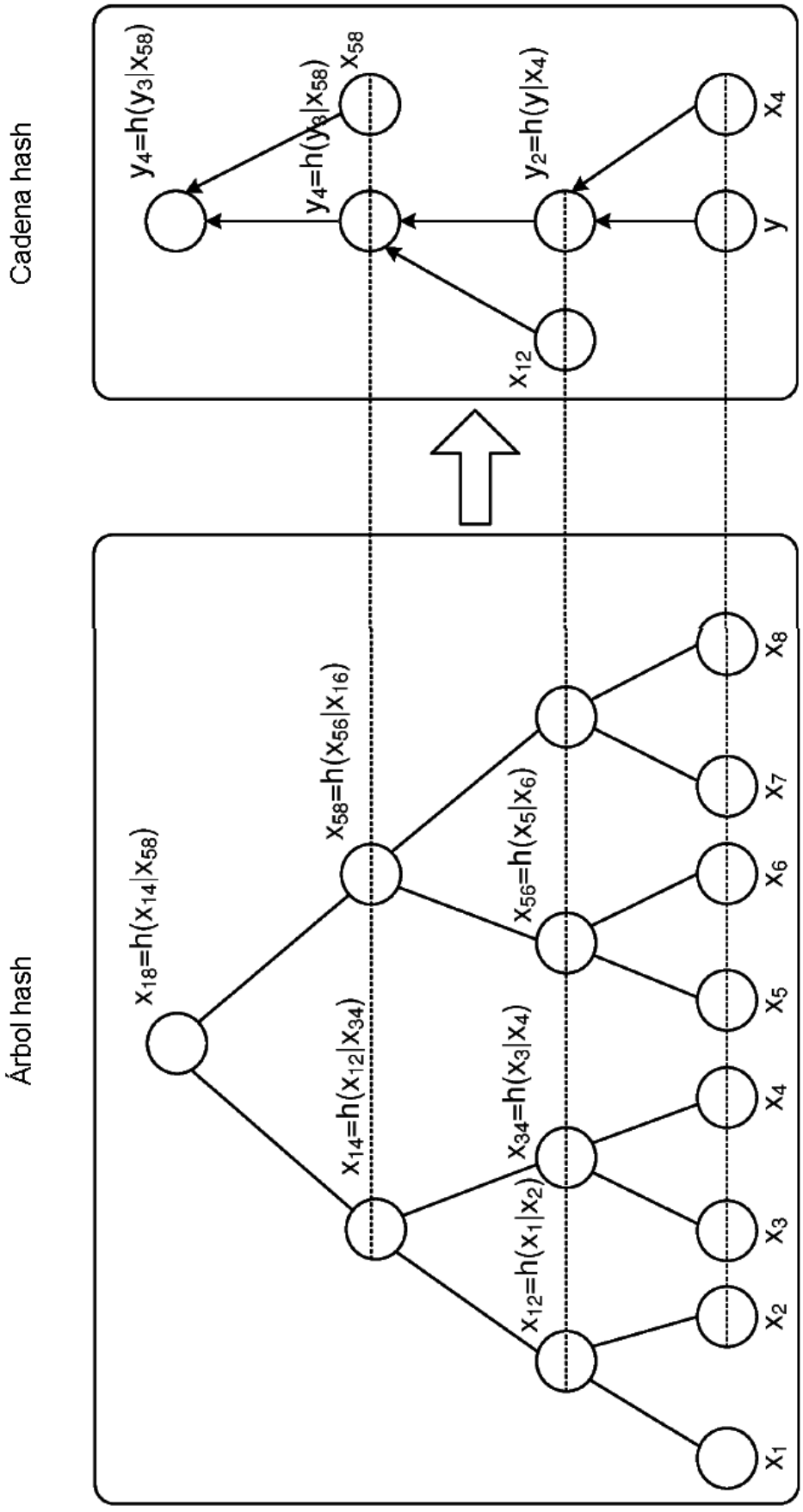
Árbol hash
Cadena hash
x18=h(x14|x58)
x14=h(x12|x34)
x58=h(x56|x16)
x12=h(x1|x2)
x34=h(x3|x4)
x56=h(x5|x6)
x1
x2
x3
x4
x5
x6
x7
x8
y4=h(y3|x58)
y4=h(y8|x58)
y2=h(y|x4)
x58
x12
y
x4

Fig. 2

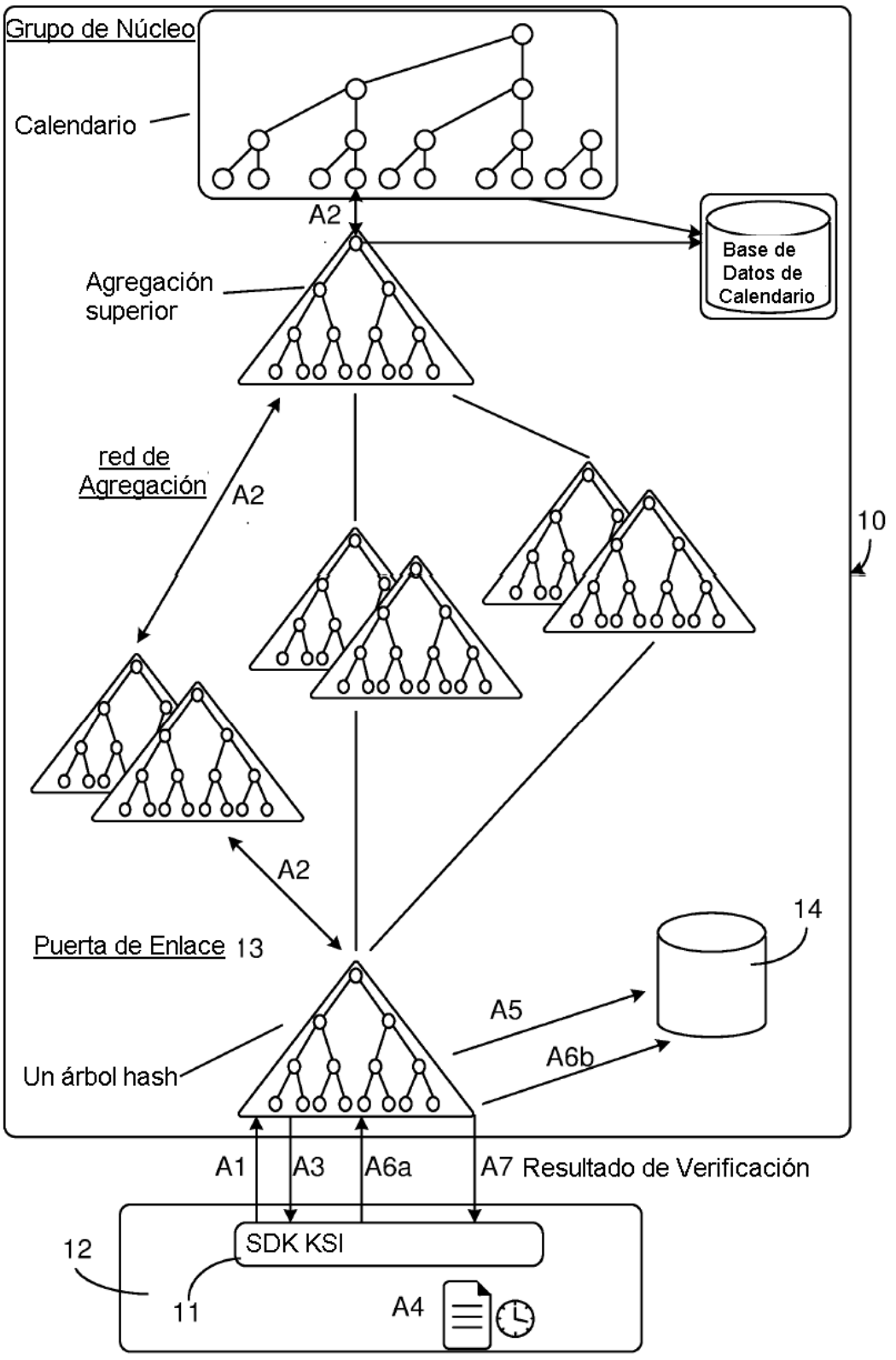
Grupo de Núcleo
Calendario
A2
Base de Datos de Calendario
Agregación superior
red de Agregación
A2
10
A2
Puerta de Enlace 13
14
A5
A6b
Un árbol hash
A1
A3
A6a
A7 Resultado de Verificación
12
SDK KSI
11
A4

Fig. 3

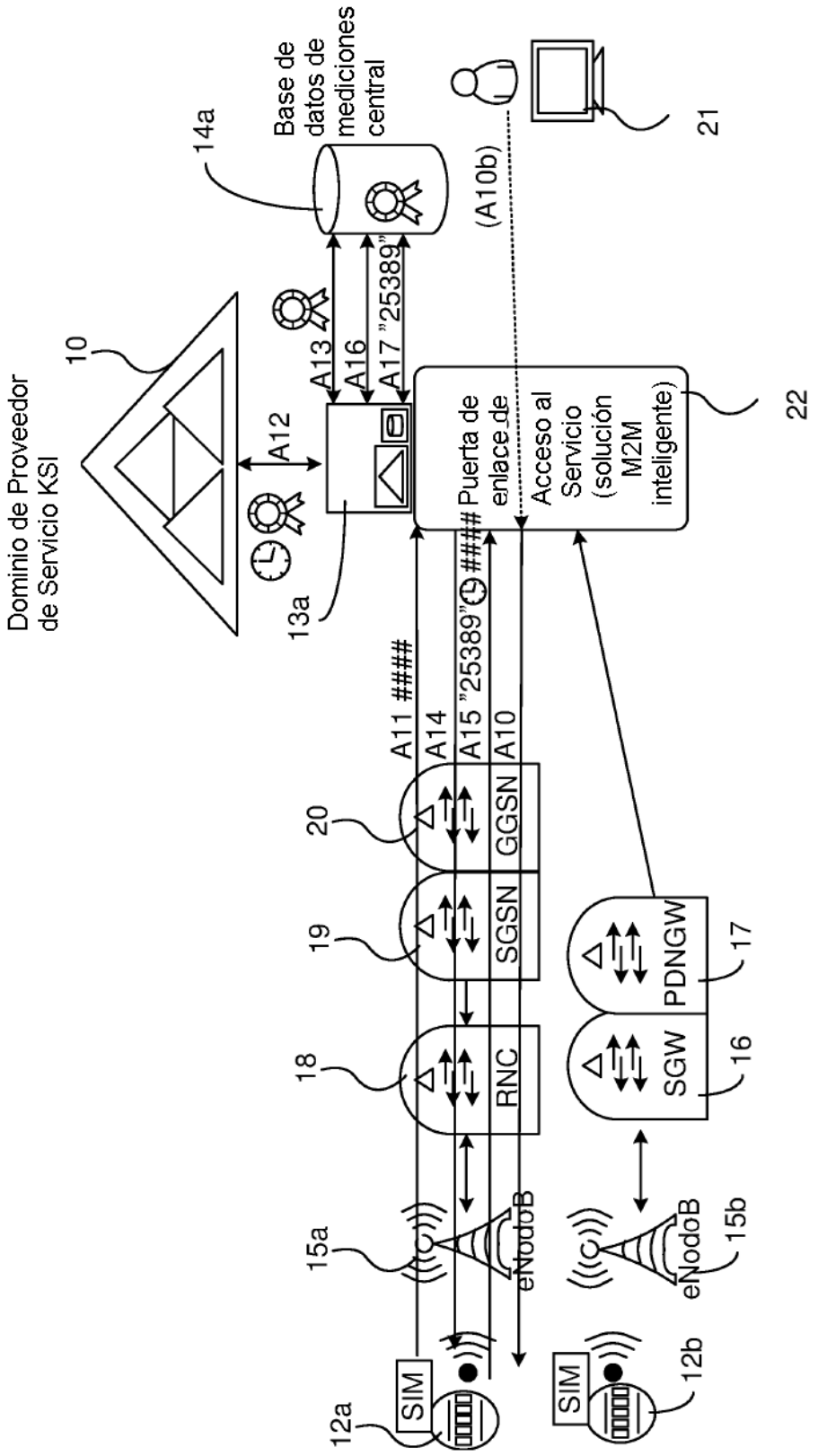
Dominio de Proveedor de Servicio KSI
10
14a
Base de datos de mediciones central
(A10b)
21
A13
A16
A17 "25389"
A12
13a
Puerta de enlace de Acceso al Servicio (solución M2M inteligente)
22
A11 ####
A14
A15 "25389" ####
A10
20
19
18
GGSN
SGSN
RNC
PDNGW
SGW
17
16
15a
15b
eNodoB
eNodoB
SIM
SIM
12a
12b

Fig. 4

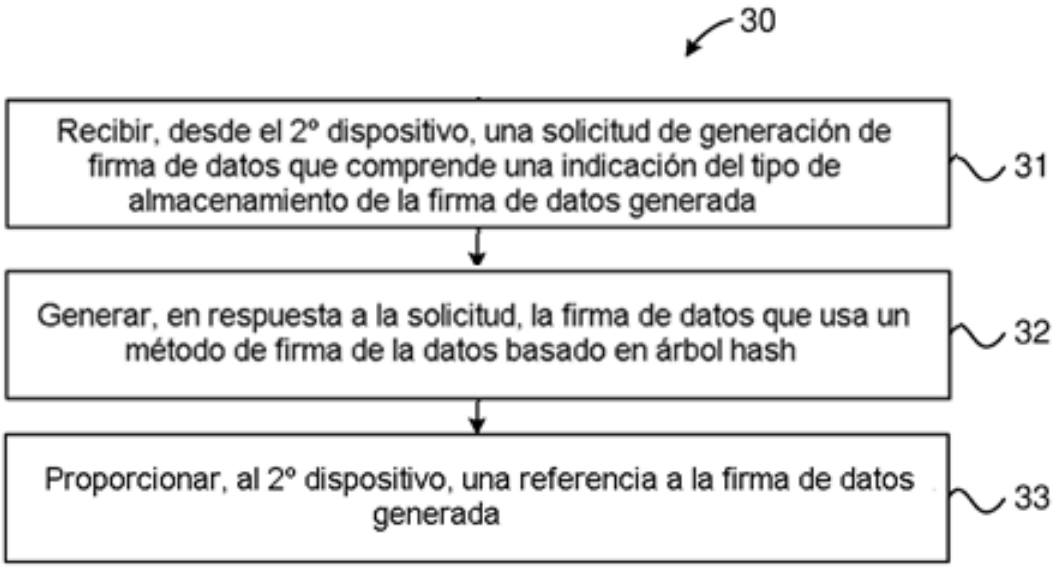

Fig. 5

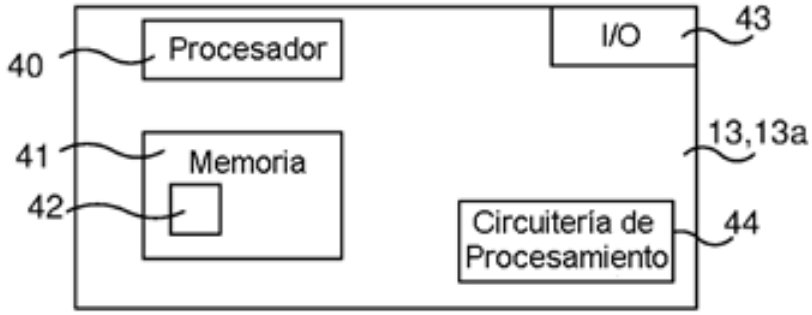

Fig. 6

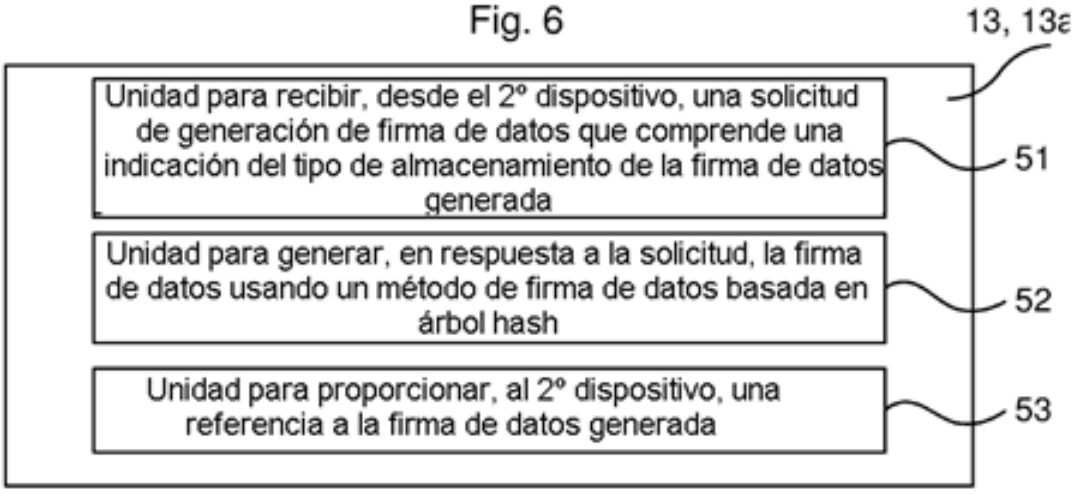

Fig. 7

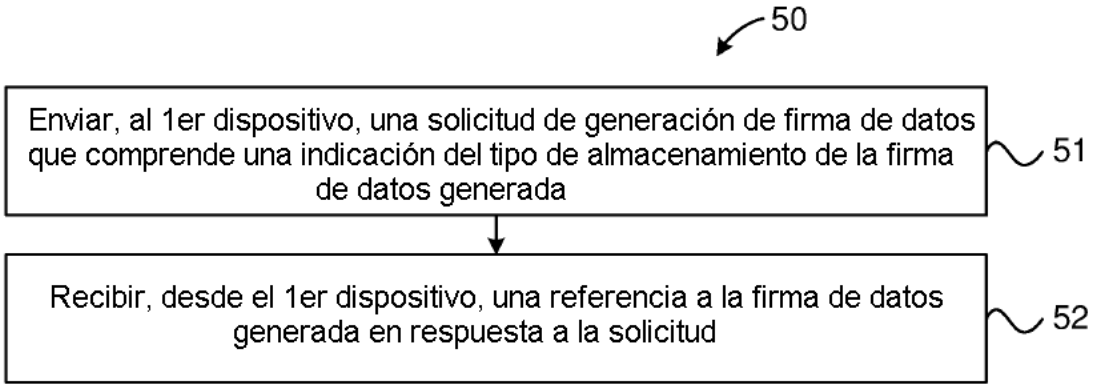

Fig. 8

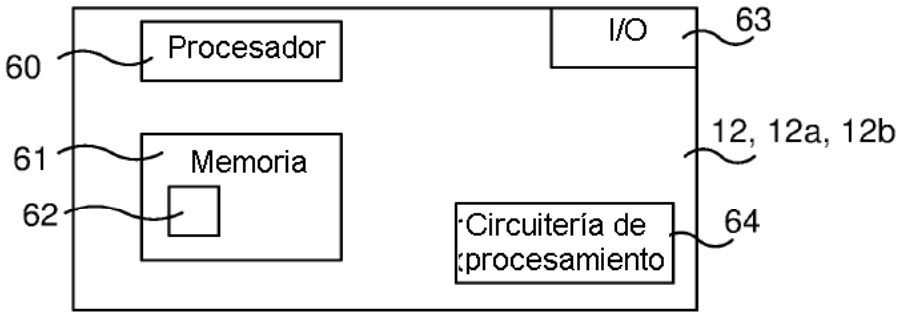

Fig. 9

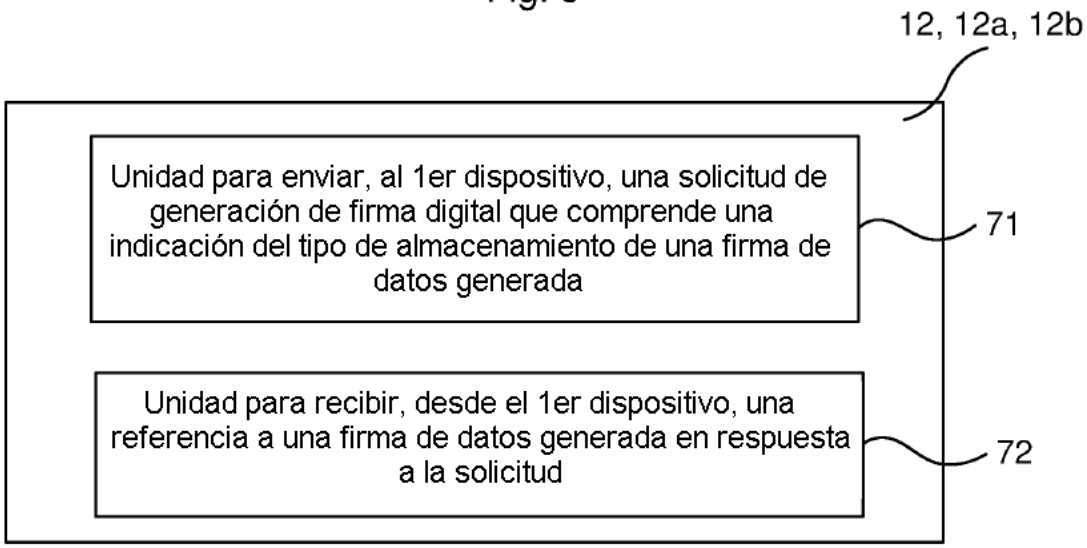

Fig. 10