(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 966 019**

(51) Int. Cl.:

G06F 21/53 (2013.01)
G06F 21/57 (2013.01)
G06F 9/455 (2008.01)
G06F 21/64 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **27.02.2020 PCT/EP2020/055155**

(87) Fecha y número de publicación internacional: **17.09.2020 WO20182482**

(96) Fecha de presentación y número de la solicitud europea: **27.02.2020 E 20707425 (3)**

(97) Fecha y número de publicación de la concesión europea: **15.11.2023 EP 3935537**

(54) Título: **Ejecución segura de controles ambientales del propietario de invitado**

(30) Prioridad:

08.03.2019 US 201916296498

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
17.04.2024

(73) Titular/es:

INTERNATIONAL BUSINESS MACHINES CORPORATION (100.0%)
New Orchard Road
Armonk, New York 10504, US

(72) Inventor/es:

BUENDGEN, REINHARD;
BRADBURY, JONATHAN y
HELLER, LISA

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Ejecución segura de controles ambientales del propietario de invitado

En los sistemas informáticos y las redes de transporte de información actuales, los elementos criptográficos son componentes tecnológicos importantes. La información puede almacenarse o transmitirse de forma criptográficamente segura para evitar el acceso no autorizado a la información almacenada o transmitida. En algunos casos, se pueden utilizar técnicas basadas en software puro y, en otros casos, se pueden utilizar soporte de hardware y elementos específicos de seguridad para realizar dicha protección de datos.

Un hipervisor o gestionador de máquinas virtuales puede controlar varios invitados *(p. ej.,* máquinas virtuales, servidores virtuales) con acceso a los recursos del sistema. Diferentes propietarios pueden generar diferentes invitados gestionados por un hipervisor común. De estos invitados, algunos pueden ser invitados seguros. Un hipervisor tradicional tiene control total sobre todos los invitados alojados. En particular, el hipervisor tiene la capacidad de inspeccionar e incluso modificar toda la memoria del invitado alojado. En un entorno de nube, dicha configuración requiere que el hipervisor y sus administradores sean totalmente confiables.

Un invitado seguro es un invitado que puede ser alojado por hipervisores que no son (totalmente) confiables. La imagen de dicho invitado estaría protegida cuando se carga y la protección de los contenidos de los recursos asignados al invitado *(p. ej., memoria,* registros de la CPU) se mantendría durante toda la vida útil del invitado. La protección del invitado comprende al menos la protección de la integridad *(p. ej.,* el hipervisor no puede cambiar maliciosamente ningún estado del invitado) y, además, puede comprender el mantenimiento de la confidencialidad de la imagen y el código iniciales y los datos que se ejecutan en el invitado. Estos servicios se pueden aplicar a cualquier interfaz entre una entidad segura y otra entidad no confiable que tradicionalmente permite el acceso a los recursos seguros por parte de esta otra entidad.

La solicitud de patente de EE. UU. 2018/019979 A1 divulga un método y un sistema para instanciar programas invitados, tales como máquinas virtuales, utilizando un componente confiable para autorizar la instanciación.

La solicitud de patente de EE. UU. 2012/054486 A1 divulga un sistema para asegurar máquinas virtuales en un entorno virtual, en el que se proporciona un servidor de autoridad de credenciales para gestionar las credenciales del entorno.

Sumario

Se superan las deficiencias de la técnica anterior y se proporcionan ventajas adicionales mediante la provisión de un método para permitir que un propietario controle la ejecución de un invitado seguro en un entorno técnico dado. El método incluye, por ejemplo: obtener, mediante un control de interfaz segura en un sistema informático, en el que el control de interfaz segura está acoplado comunicativamente a un hipervisor, en el que el hipervisor gestiona uno o más invitados, metadatos vinculados a una imagen de un invitado seguro de un propietario y gestionados por el hipervisor, en el que los metadatos comprenden uno o más controles, en el que cada control de uno o más controles indica al control de interfaz segura si se permite al hipervisor ejecutar una instancia de un invitado seguro generada con la imagen en el sistema informático en base a una presencia o ausencia de una o más configuraciones del sistema en el sistema informático; interceptar, mediante el control de interfaz segura, una orden del hipervisor para iniciar la instancia del invitado seguro a partir de la imagen del invitado seguro; determinar, mediante el control de interfaz segura, la presencia o ausencia de una o más configuraciones del sistema en el sistema informático; determinar, mediante el control de interfaz segura, en base a uno o más controles y la presencia o ausencia de la una o más configuraciones del sistema, si se permite al hipervisor ejecutar la instancia; en base a la determinación de que se permite al hipervisor ejecutar la instancia, permitir, mediante el control de interfaz segura, el inicio de la instancia por parte del hipervisor, en el sistema informático, en base a la transmisión de la orden interceptada al hipervisor; y en base a la determinación de que no se permite al hipervisor ejecutar la instancia, ignorar, mediante el control de interfaz segura, la orden.

Se superan las deficiencias de la técnica anterior y se proporcionan ventajas adicionales mediante la provisión de un producto de programa informático para permitir que un propietario controle la ejecución de un invitado seguro en un entorno técnico dado. El producto de programa informático comprende un medio de almacenamiento legible por un circuito de procesamiento y que almacena instrucciones para su ejecución por el circuito de procesamiento para realizar un método. El método incluye, por ejemplo: obtener, mediante el uno o más procesadores en un sistema informático, en el que el uno o más procesadores están acoplados comunicativamente a un hipervisor, en el que el hipervisor gestiona uno o más invitados, metadatos vinculados a una imagen de un invitado seguro de un propietario y gestionados por el hipervisor, en el que los metadatos comprenden uno o más controles, en el que cada control de uno o más controles indica al uno o más procesadores si se permite al hipervisor ejecutar una instancia de un invitado seguro generada con la imagen en el sistema informático en base a una presencia o ausencia de una o más configuraciones del sistema en el sistema informático; interceptar, mediante el uno o más procesadores, una orden del hipervisor para iniciar la instancia del invitado seguro a partir de la imagen del invitado seguro; determinar, mediante el uno o más procesadores, la presencia o ausencia de una o más configuraciones del sistema en el sistema informático; determinar, mediante el uno o más procesadores, en base al uno o más controles y la presencia o ausencia de la una o más configuraciones del sistema, si se permite al hipervisor ejecutar la instancia; en base a la determinación

de que se permite al hipervisor ejecutar la instancia, permitir, mediante el uno o más procesadores, el inicio de la instancia por parte del hipervisor, en el sistema informático, en base a la transmisión de la orden interceptada al hipervisor; y en base a la determinación de que no se permite al hipervisor ejecutar la instancia, ignorar, mediante el uno o más procesadores, la orden.

Se superan las deficiencias de la técnica anterior y se proporcionan ventajas adicionales mediante la provisión de un sistema para permitir que un propietario controle la ejecución de un invitado seguro en un entorno técnico dado. El sistema comprende una memoria, uno o más procesadores en comunicación con la memoria e instrucciones de programa ejecutables por el uno o más procesadores a través de la memoria para realizar un método. El método incluye, por ejemplo: obtener, mediante el uno o más procesadores en el sistema, en el que el uno o más procesadores están acoplados comunicativamente a un hipervisor, en el que el hipervisor gestiona uno o más invitados, metadatos vinculados a una imagen de un invitado seguro de un propietario y gestionados por el hipervisor, en el que los metadatos comprenden uno o más controles, en el que cada control de uno o más controles indica al uno o más procesadores si se permite al hipervisor ejecutar una instancia de un invitado seguro generada con la imagen en el sistema informático en base a una presencia o ausencia de una o más configuraciones del sistema en el sistema informático; interceptar, mediante el uno o más procesadores, una orden del hipervisor para iniciar la instancia del invitado seguro a partir de la imagen del invitado seguro; determinar, mediante el uno o más procesadores, la presencia o ausencia de la una o más configuraciones del sistema en el sistema informático; determinar, mediante el uno o más procesadores, en base al uno o más controles y la presencia o ausencia de la una o más configuraciones del sistema, si se permite al hipervisor ejecutar la instancia; en base a la determinación de que se permite al hipervisor ejecutar la instancia, permitir, mediante el uno o más procesadores, el inicio de la instancia por parte del hipervisor, en el sistema informático, en base a la transmisión de la orden interceptada al hipervisor; y en base a la determinación de que no se permite al hipervisor ejecutar la instancia, ignorar, mediante el uno o más procesadores, la orden.

También se describen y reivindican en el presente documento métodos y sistemas relacionados con uno o más aspectos. Además, también se describen y pueden reivindicarse en el presente documento servicios relacionados con uno o más aspectos.

Se obtienen características adicionales a través de las técnicas descritas en el presente documento. Se describen en detalle en el presente documento otras realizaciones y aspectos y se consideran parte de los aspectos reivindicados. Por ejemplo, en algunas realizaciones de la presente invención, el uno o más procesadores o el control de interfaz segura que obtienen los metadatos comprenden además: descifrar, mediante el uno o más procesadores o el control de interfaz segura, una parte de los metadatos vinculados a la imagen del invitado seguro, en donde los metadatos están protegidos en cuanto a su integridad y la parte que comprende una medida criptográfica de una imagen de arranque del invitado seguro se cifró mediante una clave obtenida usando una clave privada.

En algunas realizaciones de la presente invención, la parte cifrada de los metadatos comprende uno o más controles.

En algunas realizaciones de la presente invención, cada control del uno o más controles comprende una restricción ambiental.

En algunas realizaciones de la presente invención, las restricciones ambientales se seleccionan del grupo que consiste en: sistemas configurados para realizar mediciones de hardware y sistemas configurados para usar una clave de anfitrión no específica del sistema.

En algunas realizaciones de la presente invención, la clave privada es propiedad del control de interfaz segura y se usa exclusivamente por el control de interfaz segura.

En algunas realizaciones de la presente invención, la clave obtenida utilizando la clave privada se comparte entre el control de interfaz segura y el propietario.

En algunas realizaciones de la presente invención, los metadatos comprenden valores obtenidos de una imagen de arranque del invitado seguro calculada utilizando una función unidireccional resistente a colisiones.

En algunas realizaciones de la presente invención, el uno o más controles comprende cada uno una designación positiva o una designación negativa para unas configuraciones del sistema dadas, en donde la designación positiva indica que se permite a la instancia del invitado seguro ejecutarse en el sistema informático que comprende la configuración del sistema dada y la designación negativa indica que no se permite a la instancia del invitado seguro ejecutarse en el sistema informático que comprende la configuración del sistema dada.

En algunas realizaciones de la presente invención, los metadatos son inaccesibles para la instancia del invitado seguro.

En algunas realizaciones de la presente invención, determinar si se permite al hipervisor ejecutar la instancia comprende además: identificar, mediante el uno o más procesadores y/o el control de interfaz segura, en el uno o más controles, un control relevante para una configuración de la una o más configuraciones del sistema; y determinar, mediante el uno o más procesadores y/o el control de interfaz segura, si el control permite o restringe la ejecución de la instancia, en base al control.

En algunas realizaciones de la presente invención, el método comprende además: monitorizar, mediante el uno o más procesadores y/o el control de interfaz segura, la una o más configuraciones del sistema durante el tiempo de ejecución de la instancia; determinar, mediante el uno o más procesadores y/o el control de interfaz segura, que al menos una configuración de la una o más configuraciones cambió durante el tiempo de ejecución; identificar, mediante el uno o más procesadores y/o el control de interfaz segura, un control dado del uno o más controles relevantes para la al menos una configuración; y determinar, mediante el uno o más procesadores y/o el control de interfaz segura, en base a la al menos una configuración y el control dado, si se permite al hipervisor ejecutar la instancia. En base a la determinación de que no se permite al hipervisor ejecutar la instancia, terminar, mediante el uno o más procesadores, y/o el control de interfaz segura, a través del hipervisor, la instancia.

En algunas realizaciones de la presente invención, interceptar la orden del hipervisor para iniciar la instancia del invitado seguro a partir de la imagen del invitado seguro comprende además: realizar, mediante el uno o más procesadores y/o el control de interfaz segura, una verificación de integridad en los metadatos; y en base a completar con éxito la verificación de integridad, leer, mediante el uno o más procesadores y/o el control de interfaz segura, el uno o más controles en los metadatos.

Breve descripción de los dibujos

Uno o más aspectos se señalan particularmente y se reivindican claramente como ejemplos en las reivindicaciones al final de la memoria descriptiva. Lo anterior y objetos, características y ventajas de uno o más aspectos son evidentes a partir de la siguiente descripción detallada tomada junto con los dibujos adjuntos, en los que:

La FIG. 1 es un flujo de trabajo que ilustra ciertos aspectos de algunas realizaciones de la presente invención;

La FIG. 2 ilustra varios aspectos de algunas realizaciones de la presente invención;

La FIG. 3 es un flujo de trabajo que ilustra ciertos aspectos de algunas realizaciones de la presente invención;

La FIG. 4 representa una realización de un nodo informático que se puede utilizar en un entorno informático en la nube;

La FIG. 5 representa un entorno de informática en la nube de acuerdo con realización de la presente invención; y

La FIG. 6 representa capas de modelo de abstracción de acuerdo con una realización de la presente invención;

Descripción detallada

Las figuras adjuntas, en las que números de referencia similares se refieren a elementos idénticos o funcionalmente similares en todas las vistas separadas y que se incorporan y forman parte de la memoria descriptiva, ilustran además la presente invención y, junto con la descripción detallada de la invención, sirven para explicar los principios de la presente invención. Como entenderá un experto en la materia, las figuras adjuntas se proporcionan para facilitar la comprensión e ilustran aspectos de ciertas realizaciones de la presente invención. La invención no se limita a las realizaciones representadas en las figuras.

Como entenderá un experto en la materia, el código de programa, como se refiere a lo largo de esta solicitud, incluye tanto software como hardware. Por ejemplo, el código de programa en ciertas realizaciones de la presente invención incluye hardware de función fija, mientras que otras realizaciones utilizaron una implementación basada en software de la funcionalidad descrita. Ciertas realizaciones combinan ambos tipos de código de programa. Un ejemplo de código de programa, también referido como uno o más programas, se representa en la FIG. 4 como el programa/utilidad 40, que tiene un conjunto (al menos uno) de módulos de programa 42, puede almacenarse en la memoria 28.

Una máquina virtual (VM), que se ejecuta como invitado bajo el control de un hipervisor de anfitrión, depende de ese hipervisor para proporcionar servicios de virtualización de forma transparente para ese invitado. Estos servicios se pueden aplicar a cualquier interfaz entre una entidad segura y otra entidad no confiable que permite tradicionalmente el acceso a los recursos seguros por esta otra entidad. El término «sistema invitado» o «invitado» puede indicar, *p. ej.,* un sistema operativo que se ejecuta en una máquina virtual, VM, en un hipervisor. Se puede asignar un usuario al sistema invitado. Puede ser que se pueda asignar una clave criptográfica específica al sistema invitado. El hipervisor mencionado se puede utilizar para realizar dicha asignación. El sistema invitado puede, *p. ej.,* una ser máquina virtual, *es decir,* una VM, que hace funcionar o ejecuta un sistema operativo invitado.

El término «contenido» puede indicar cualquier cadena basada en caracteres. La cadena puede comprender texto legible o cualquier otro dato binario.

El término «vinculado criptográficamente», utilizado para expresar que un primer componente está vinculado criptográficamente a un segundo componente, significa que el primer componente está protegido en cuanto a su integridad y contiene una medida de todos los datos contenidos en el segundo componente. Siendo producida la medida por una función unidireccional resistente a colisiones. Los ejemplos de tales medidas incluyen, pero no se limitan a, hashes criptográficos, códigos de autenticación de mensajes y/o firmas criptográficas.

Las realizaciones de la presente invención incluyen un método implementado por ordenador, un producto de programa informático y un sistema informático que incluye un código de programa ejecutado en al menos un circuito de procesamiento y/o hardware que comprende un código de programa que permite a un propietario de un invitado de ejecución segura (SE) dar instrucciones a un control de interfaz segura que es hardware, firmware o una combinación de los mismos, para dar instrucciones al control de interfaz segura para permitir o no permitir que un invitado seguro se ejecute en un entorno anfitrión específico. En realizaciones de la presente invención, la instrucción del propietario de invitado es específica para una imagen de invitado particular. Por lo tanto, el propietario comunica una o más restricciones ambientales a la interfaz de control segura para cada invitado, de modo que el control de interfaz segura puede hacer cumplir una o más restricciones en un sistema dado. En las realizaciones de la presente invención, el propietario comunica las restricciones ambientales particulares para una imagen dada a través de metadatos que están vinculados criptográficamente a la imagen de arranque del invitado seguro y el control de interfaz segura hace cumplir los detalles especificados, en las realizaciones de la presente invención, para el invitado seguro dado. En algunas realizaciones de la presente invención, los metadatos asociados a un invitado seguro están protegidos en cuanto a su integridad *(p. ej.,* al menos parte de los metadatos se pueden cifrar). Los metadatos pueden contener valores obtenidos por el código de programa a partir de la imagen de arranque del invitado seguro que el código de programa calcula utilizando una función unidireccional resistente a colisiones *(p. ej.,* una función hash criptográfica). La función resistente a colisiones proporciona seguridad a los valores porque su aplicación hace que sea computacionalmente inviable construir una imagen alternativa para la que se puedan obtener los mismos valores. En algunas realizaciones de la presente invención, los metadatos comprenden designaciones positivas y negativas específicas relevantes para diversas restricciones ambientales. En algunas realizaciones de la presente invención, los metadatos comprenden datos de instalación del invitado seguro *(p. ej.,* el encabezado de ejecución segura (SE)).

En las realizaciones de la presente invención, el control de interfaz segura protege al sistema (anfitrión) de ejecutar un invitado seguro que entre en conflicto con las configuraciones del sistema. El control de interfaz segura evita este conflicto al rechazar las solicitudes para iniciar un invitado seguro en base a la determinación de que los controles de los metadatos vinculados criptográficamente a la imagen de invitado seguro designan al invitado seguro incompatible con el sistema, y al monitorizar el entorno técnico después de ejecutar el invitado seguro y determinar que un cambio en el entorno técnico hace que el invitado seguro ya no sea compatible con el entorno técnico. Si durante el tiempo de ejecución, un cambio en un entorno técnico en el que se ejecuta un invitado seguro crea un conflicto con los controles (cuando los controles de los metadatos vinculados criptográficamente a la imagen de los invitados seguros permitieron inicialmente que el control de interfaz segura iniciara los invitados seguros), el control de interfaz segura inicia una terminación del invitado seguro por el hipervisor. Por ejemplo, en realizaciones de la presente invención, cuando el control de interfaz segura evalúa si iniciar un invitado seguro: 1) el control de interfaz segura lee los controles individuales que indican restricciones ambientales en los metadatos que están vinculados criptográficamente a la imagen del invitado seguro; 2) el control de interfaz segura compara los controles con las configuraciones del sistema anfitrión en el que se ejecutaría la imagen, si se iniciara; 3) el control de interfaz segura inicia el invitado seguro si las configuraciones del sistema coinciden con los controles ambientales en los metadatos; y 4) el control de interfaz segura monitoriza las configuraciones del sistema del entorno técnico en el que se ejecuta el invitado seguro, y durante el tiempo de ejecución, si el control de interfaz segura determina que las configuraciones del sistema han cambiado de forma que entran en conflicto con los controles, el control de interfaz segura termina el invitado seguro. Los ejemplos de configuraciones del sistema que pueden incorporarse en controles que comprenden restricciones ambientales en los metadatos pueden incluir, pero no se limitan a: 1) sistemas configurados para realizar mediciones de hardware; y 2) sistemas configurados para usar una clave de anfitrión no específica del sistema.

La FIG. 1 es un flujo de trabajo 100 que ilustra cierta funcionalidad de algunas realizaciones de la presente invención. Como se ilustrará en la FIG. 1, en las realizaciones de la presente invención, para permitir una ejecución segura, una imagen de un invitado seguro se vincula criptográficamente a partir de metadatos, que se comunican de forma segura a un control de interfaz segura *(p. ej.,* un componente de hardware y/o software confiable, firmware confiable) en base a una clave de anfitrión privada, a la que solo puede acceder el control de interfaz segura. En las realizaciones de la presente invención, un propietario del invitado seguro determina el comportamiento de control de interfaz segura, permitiendo o prohibiendo la ejecución del invitado seguro en ciertos entornos técnicos. Una ventaja de las realizaciones de la presente invención sobre los enfoques existentes para garantizar la seguridad del invitado es que los controles granulares en los metadatos, tal como los impone el control de interfaz segura, permiten al propietario limitar la ejecución de los invitados seguros a fin de abordar problemas de seguridad específicos. Por ejemplo, el propietario puede controlar si el control de interfaz segura puede iniciar un invitado seguro en un sistema anfitrión configurado para realizar mediciones de hardware y/o en un sistema anfitrión configurado para usar una clave de anfitrión no específica del sistema. En algunas realizaciones de la presente invención, el código de programa que se ejecuta en al menos un circuito de procesamiento y/o proporcionado por hardware, restringe la ejecución de un invitado seguro.

Haciendo referencia a la FIG. 1, en algunas realizaciones de la presente invención, un control de interfaz segura *(p. ej.,* hardware, software, firmware, una combinación, etc.) intercepta una solicitud para iniciar un invitado seguro en base a una imagen (105) del invitado seguro. El control de interfaz segura obtiene metadatos vinculados al invitado seguro que se va a iniciar, de un propietario (110) del invitado seguro. En algunas realizaciones de la presente invención, los metadatos están protegidos en cuanto a su integridad y vinculados criptográficamente a una imagen de arranque del invitado seguro que se va a iniciar por un propietario (SE). Los metadatos comprenden controles y cada

control indica una restricción para el invitado seguro; estas restricciones comprenden restricciones ambientales.

Volviendo a la FIG. 1, en una realización de la presente invención, el control de interfaz segura realiza una verificación de integridad en los metadatos (120). Si la interfaz de control segura determina que los metadatos no pasan la verificación de integridad, la interfaz de control segura devuelve un error al propietario (u otro solicitante para el caso del invitado seguro) (125). Al devolver un error al propietario en base a una verificación de integridad fallida, el control de interfaz segura impide el inicio solicitado del invitado seguro. Incluso si los metadatos pasan la verificación de integridad, el control de interfaz segura puede abortar el inicio del invitado seguro.

Si el control de interfaz segura determina que los metadatos pasan la verificación de integridad, el control de interfaz segura obtiene los controles de los metadatos (130). El control de interfaz segura analiza la configuración del sistema del sistema anfitrión para determinar si la configuración del sistema es compatible con los controles en los metadatos, para determinar si se permite al invitado seguro iniciarse en el sistema (140). Si el control de interfaz segura determina que no se permite el inicio al invitado seguro, en base a los metadatos, el control de interfaz segura devuelve un error al propietario y/o a la entidad que solicitó el inicio del invitado seguro (145). Al devolver un error, el control de interfaz segura abortó un intento de iniciar un invitado seguro. En algunas realizaciones de la presente invención, el control de interfaz segura ignora la solicitud para iniciar el invitado seguro. Dependiendo de la configuración del control de interfaz segura, puede variar una respuesta a una orden de un hipervisor para iniciar un invitado seguro (desde una imagen), en el que los metadatos vinculados a la imagen indican que el invitado seguro no está permitido en el sistema dado, en base que a los controles son incompatibles con las configuraciones del sistema. Si el control de interfaz segura determina que el invitado seguro está permitido, en base a los metadatos y las configuraciones del sistema, el control de interfaz segura permite al hipervisor iniciar el invitado seguro, en base a la imagen (150). Debido a que el control de interfaz segura intercepta los intentos de iniciar un invitado seguro, el control de interfaz segura interrumpe una orden para iniciar un invitado seguro antes de que se pueda iniciar el invitado seguro. Por lo tanto, el control de interfaz segura rechaza de manera eficaz las solicitudes que entran en conflicto con las configuraciones del sistema, en base a los controles de los metadatos. La prevención y el permiso de varios invitados seguros se logran mediante el control de interfaz segura antes de que cualquier otro componente del sistema pueda iniciar los invitados seguros. Por ejemplo, en algunas realizaciones de la presente invención, los controles en los metadatos vinculados a una imagen de un invitado seguro pueden dar instrucciones al control de interfaz segura para restringir a un invitado seguro, en base a que el sistema anfitrión esté configurado para realizar mediciones de hardware y/o a un sistema anfitrión configurado para usar una clave de anfitrión no específica del sistema.

Volviendo a la FIG. 1, como se ha expuesto anteriormente, durante el tiempo de ejecución del invitado seguro, el control de interfaz segura puede monitorizar las configuraciones del sistema (160) y si el control de interfaz segura determina que las configuraciones del sistema han cambiado de una manera que entra en conflicto con los controles, lo que indicaba que se permitió al invitado seguro ejecutarse en el entorno, el control de interfaz segura puede terminar el invitado seguro (y/o hacer que el hipervisor termine el invitado seguro), durante el tiempo de ejecución (170). El control de interfaz segura puede continuar monitorizando las configuraciones del sistema en caso de que surja este tipo de cambio (165). En algunas realizaciones de la presente invención, el control de interfaz segura da instrucciones *(p. ej.,* obliga) al hipervisor a terminar el invitado seguro.

La FIG. 2 es un entorno técnico 200 que incluye varios aspectos de algunas realizaciones de la presente invención. Los componentes que comprenden el entorno técnico 200 de la FIG. 2 ilustran cómo un control de interfaz segura permite a un propietario de invitado controlar la funcionalidad de una imagen de invitado, mediante un cifrado único para cada propietario, y el control de interfaz segura habilita/deshabilita a un invitado seguro, en base a una imagen dada, para ejecutarse en entornos de anfitrión particulares. Para la Ejecución Segura, un invitado seguro 210 se vincula criptográficamente a los metadatos 240, que se comunican de forma segura a un control de interfaz segura 230 *(p. ej.,* un firmware confiable, un componente confiable, etc.), en base a una clave de anfitrión privada a la que solo puede acceder el control de interfaz segura 230. El propietario del invitado seguro 210 controla el control de interfaz segura 230 permitiendo o prohibiendo la ejecución del invitado seguro 210, en base a las restricciones ambientales impuestas por el propietario. Por ejemplo, las restricciones ambientales impuestas por el propietario, a través del control de interfaz segura 230, podrían limitar la ejecución de un invitado seguro en un sistema anfitrión configurado para realizar mediciones de hardware y/o en un sistema anfitrión configurado para usar una clave de anfitrión no específica del sistema. En algunas realizaciones de la presente invención, la clave para cifrar partes de los metadatos se obtiene utilizando una clave privada a la que solo puede acceder el control de interfaz segura 230. En base a las preferencias del propietario, como se señala en los metadatos 240 asociados con el invitado seguro 210, el control de interfaz segura 230 controla el inicio del invitado seguro 210, en base a las configuraciones del sistema 270. El control de interfaz segura 230 también monitoriza las configuraciones del sistema 270 durante el tiempo de ejecución, de modo que el control de interfaz segura 230 puede hacer que el hipervisor 220 termine el invitado seguro 210, siempre y cuando las configuraciones del sistema 270 cambien de tal manera que los controles 260 ya no indiquen que el invitado seguro 210 es compatible con el entorno técnico 200.

En algunas realizaciones de la presente invención, un invitado seguro 210 *(p. ej.,* máquina virtual, servidor virtual) se controla por un hipervisor 220 *(p. ej.,* gestionador de máquina virtual). El control de interfaz segura 250 obtiene, de un propietario del invitado seguro 210, mediante el hipervisor 220, metadatos asociados con el invitado seguro 240. Estos metadatos 240 están protegidos en cuanto a su integridad (y parte de los metadatos 240, *es decir,* un secreto de invitado seguro 250, también está protegido por confidencialidad *(es decir,* cifrado)). En algunas realizaciones de la

presente invención, los metadatos 240 están vinculados criptográficamente a la imagen de arranque del invitado seguro 210. En algunas realizaciones de la presente invención, el secreto del invitado seguro 250 está contenido en los datos de instalación del invitado seguro *(p. ej.,* el encabezado de ejecución segura (SE)), que el control de interfaz segura 230 obtiene del propietario. En algunas realizaciones de la presente invención, los metadatos 240 contienen valores obtenidos de la imagen de arranque del invitado seguro 210 que se calculan utilizando una función unidireccional resistente a colisiones *(p. ej., una función hash criptográfica),* de modo que la construcción de una imagen alternativa para la que puedan obtenerse los mismos valores es computacionalmente inviable.

Como se ilustra en la FIG. 2, los metadatos vinculados al invitado seguro 240 comprenden un secreto 250. En algunas realizaciones de la presente invención, los metadatos del invitado están protegidos en cuanto a su integridad y el secreto se cifra mediante una clave obtenida utilizando una clave privada propiedad del control de interfaz segura 230. Los datos cifrados por la clave pueden comprender una medida criptográfica de una imagen de arranque del invitado dado. Por lo tanto, en algunas realizaciones de la presente invención, la parte de los metadatos 240 que contiene el secreto 250 se cifra mediante una clave que solo el control de interfaz segura 230 puede calcular. Los metadatos 240 no necesitan ser accesibles para el invitado seguro 210 en sí mismo.

Los metadatos 240 están vinculados a la imagen del invitado seguro 210, estando el secreto 250 vinculado criptográficamente a la imagen del invitado seguro, pero no forman parte del invitado. Más bien, como se describe en el presente documento, se transporta independientemente al control de interfaz segura *(p. ej.,* firmware, un componente confiable, software, hardware, etc.) *(p. ej.,* FIG. 1, 110). En las realizaciones de la presente invención, el secreto 250 se transporta a través de un canal seguro *(es decir,* cifrado) como parte de los metadatos del invitado y se vincula criptográficamente al invitado. En algunas realizaciones de la presente invención, los metadatos 240 están vinculados criptográficamente a un invitado*(p. ej.,* contienen una firma de la imagen del invitado), por lo que los metadatos de un invitado no pueden utilizarse indebidamente como metadatos de otro invitado. Por lo tanto, el control de interfaz segura 230 puede verificar que la imagen del invitado y los metadatos/secreto van juntos *(p. ej.,* FIG. 1, 120). Los metadatos 240 transferidos *(p. ej.,* de forma independiente, a través de un canal seguro) al control de interfaz segura 230 están protegidos en cuanto a su integridad y por confidencialidad.

Además de incluir un secreto 250, los metadatos 240 también incluyen los controles 260, que el control de interfaz segura 230 utiliza para habilitar y prohibir que el hipervisor 220 inicie/ejecute el invitado seguro 210. El control de interfaz segura 230 determina si se permite al invitado seguro 210 iniciar (y, en algunos casos, continuar ejecutándose) en el entorno técnico 200, en base a la comparación de los controles 260 en los metadatos 240 con las configuraciones del sistema 270. El control de interfaz segura 230 no permite que un hipervisor 220 inicie el invitado seguro 210 y, durante el tiempo de ejecución, puede hacer que el hipervisor 220 termine un invitado seguro 210 en base a las restricciones descritas por los metadatos 240 *(p. ej.,* los controles 260) que indican una incompatibilidad con el entorno técnico 200, en base a las configuraciones del sistema 270. En algunas realizaciones de la presente invención, los metadatos 240 contienen controles 260 que indican la compatibilidad con aspectos de un entorno técnico dado, si los controles son positivos, o restricciones para la ejecución en un entorno técnico dado, si los controles son negativos. Cuando las configuraciones 270 del sistema cambian después de que se haya iniciado previamente un invitado seguro 210 en base a los valores de las configuraciones 270 del sistema anteriores, el control 230 de interfaz segura puede hacer que el hipervisor 220 termine, el invitado seguro 210, durante el tiempo de ejecución, ya que en el control 230 de interfaz segura se determina que las configuraciones 270 del sistema modificadas son incompatibles con los controles 260, de modo que los controles 260 indican que el invitado seguro 210 está restringido para ejecutarse en el entorno técnico 200.

En las realizaciones de la presente invención, los controles 260 en los metadatos 240 vinculados a la imagen del invitado seguro 210 indican una o más restricciones ambientales. En algunas realizaciones de la presente invención, dependiendo de la configuración de un control dado de los controles 260, se puede iniciar o no iniciar un invitado seguro 210, si se cumple la restricción ambiental relevante *(p. ej.,* mediante las configuraciones del sistema 270). Por ejemplo, cuando dicho control es negativo, el control de interfaz segura 230 puede inhibir el inicio de un invitado seguro *(p. ej.,* mediante un hipervisor 220) cuando se cumple la restricción ambiental asociada. De lo contrario, si el control es positivo, el control de interfaz segura 230 puede permitir el inicio de un invitado seguro 210 *(p. ej.,* mediante el hipervisor 220) cuando se cumple la restricción ambiental asociada. Al evaluar los metadatos 240 para un invitado seguro 210, el control de interfaz segura 230 puede leer los controles 260 y, para cada control, el invitado seguro 210 determina si el entorno informático 200 cumple con la restricción ambiental asociada para cada control, en base a las configuraciones del sistema 270.

La FIG. 3 es un flujo de trabajo 300 que ilustra varios aspectos de algunas realizaciones de la presente invención; Como se ilustra en la FIG. 3, aspectos de varias realizaciones de la presente invención permiten al propietario de un invitado de ejecución segura dar instrucciones a un control de interfaz segura para permitir o no permitir que el invitado seguro del propietario se ejecute en un entorno de anfitrión específico. La decisión del propietario de invitado puede ser específica para una imagen de invitado en particular y el propietario de invitado comunica la decisión al sistema, en general, para que el sistema *(es decir,* mediante el control de interfaz segura) la haga cumplir, utilizando los metadatos vinculados criptográficamente al invitado seguro.

Volviendo a la FIG. 3, en una realización de la presente invención, un control de interfaz segura en un sistema informático, en el que el control de interfaz segura está acoplado comunicativamente a un hipervisor y el hipervisor

gestiona uno o más invitados, obtiene metadatos vinculados a una imagen de un invitado seguro para ser iniciado por el hipervisor, en el que los metadatos comprenden uno o más controles, en el que cada control del uno o más controles indica al control de interfaz segura si se permite a un invitado seguro ejecutarse en un entorno técnico (310) particular. Los metadatos en sí mismos pueden ser inaccesibles para el invitado seguro. Como parte de la obtención de los metadatos, en algunas realizaciones de la presente invención, el control de interfaz segura descifra una parte de los metadatos vinculados a una imagen de un invitado seguro. Los metadatos están protegidos en cuanto a su integridad y esta parte se cifró mediante una clave obtenida utilizando una clave privada que comprende una medida criptográfica de una imagen de arranque del invitado seguro, en algunas realizaciones de la presente invención. La clave privada puede ser propiedad del control de interfaz segura y utilizarse exclusivamente por el control de interfaz segura. Una clave obtenida utilizando la clave privada se puede compartir entre el control de interfaz segura y el propietario *(p. ej.,* únicamente). En algunas realizaciones de la presente invención, los metadatos comprenden valores obtenidos de una imagen de arranque del invitado seguro calculada utilizando una función unidireccional resistente a colisiones.

Como se ilustra en la FIG. 2, el uno o más controles *(p. ej.,* FIG. 2, 260) son parte de los metadatos *(p. ej.,* FIG. 2, 240). En algunas realizaciones de la presente invención, el uno o más controles están contenidos en una parte cifrada de los metadatos. El control de interfaz segura puede descifrar esta parte de los metadatos, que está vinculada a una imagen del invitado seguro, en la que los metadatos están protegidos en cuanto a su integridad y la parte cifrada comprende una medida criptográfica de una imagen de arranque del invitado seguro, la parte cifrada se cifra mediante una clave obtenida utilizando una clave privada. En algunas realizaciones de la presente invención, la clave privada es mediante una clave obtenida usando una clave privada que comprende una medida criptográfica de una imagen de arranque del invitado seguro. Cada uno de los controles puede comprender una designación positiva o una designación negativa para varias configuraciones del sistema, donde la designación positiva indica que se permite al invitado seguro ejecutarse en base a una configuración del sistema dada y la designación negativa indica que no se permite al invitado ejecutarse en base a la configuración del sistema dada.

Haciendo referencia a la FIG. 3, el control de interfaz segura intercepta una orden para iniciar un invitado seguro, en base a la imagen (320). En algunas realizaciones de la presente invención, la orden es por el hipervisor *(p. ej.,* una entidad no segura) que gestiona la ejecución del invitado del sistema anfitrión. El control de interfaz segura determina, en base al uno o más controles, si el invitado seguro puede ejecutarse en el entorno técnico, como se define por las configuraciones del sistema (330). Para determinar si se permite al invitado seguro ejecutarse, en algunas realizaciones de la presente invención, el control de interfaz segura identifica, en el uno o más controles, un control relevante para una o más de las configuraciones del sistema (334). El control de interfaz segura determina si el control permite o restringe la ejecución del invitado seguro en base a las configuraciones del sistema (336).

En base a la determinación de que se permite al invitado seguro ejecutarse, el control de interfaz segura permite al hipervisor comenzar la ejecución del invitado seguro, en base a la imagen, dentro del sistema informático (340). En base a la determinación de que no se permite al invitado seguro ejecutarse, el control de interfaz segura ignora la orden para iniciar el invitado seguro (345). Al ignorar la orden, la interfaz de control segura evita que el hipervisor comience la ejecución del invitado seguro.

En algunas realizaciones de la presente invención, el control de interfaz segura, durante el tiempo de ejecución del invitado seguro, determina que se ha realizado un cambio en las configuraciones (350). El control de interfaz segura determina, en base al uno o más controles, si se permite al invitado seguro ejecutarse en el entorno técnico, como se define por las configuraciones del sistema (basadas en el cambio) (330). Para determinar si se permite al invitado seguro ejecutarse, en algunas realizaciones de la presente invención, el control de interfaz segura identifica, en el uno o más controles, un control relevante para una o más de las configuraciones del sistema (334). El control de interfaz segura determina si el control permite o restringe la ejecución del invitado seguro en base a las configuraciones del sistema (336). En base a la determinación de que se permite al invitado seguro ejecutarse, el control de interfaz segura permite al hipervisor continuar la ejecución del invitado seguro, en base a la imagen, dentro del sistema informático (340). En base a la determinación de que no se permite al invitado seguro ejecutarse, el control de interfaz segura hace que el hipervisor termine el invitado seguro (335).

Las realizaciones de la presente invención incluyen un método implementado por ordenador, un producto de programa informático y un sistema informático, donde un control de interfaz segura en un sistema informático, donde el control de interfaz segura puede ser uno cualquiera, o una combinación, de firmware, hardware y software, obtiene metadatos vinculados a una imagen de un invitado seguro que ser iniciado por un propietario y gestionado por un hipervisor. El hipervisor gestiona uno o más invitados y los metadatos comprenden uno o más controles. Cada control del uno o más controles indica al control de interfaz segura si se permite a un invitado seguro generado con la imagen, iniciarse (o mantenerse) por un control de interfaz segura, en base a las configuraciones del sistema. Por ejemplo, en algunas realizaciones de la presente invención, el código de programa de control de interfaz segura (software, hardware y/o firmware, etc.) en un sistema informático obtiene metadatos vinculados a una imagen de un invitado seguro de un propietario y gestionada por el hipervisor, donde el control de interfaz segura se acopla comunicativamente a un hipervisor, donde el hipervisor gestiona uno o más invitados, donde cada control del uno o más controles indica al control de interfaz segura si se permite al hipervisor ejecutar una instancia de un invitado seguro generado con la imagen en el sistema informático en base a la presencia o ausencia de una o más configuraciones del sistema en el sistema informático. El código del programa intercepta una orden del hipervisor para iniciar la instancia del invitado seguro a partir de la imagen del invitado seguro. El código del programa determina la presencia o la ausencia de la

una o más configuraciones del sistema en el sistema informático. El código del programa determina, en base al uno o más controles y la presencia o la ausencia de la una o más configuraciones del sistema, si se permite al hipervisor ejecutar la instancia. En base a la determinación de que se permite al hipervisor ejecutar la instancia, el código del programa permite el inicio de la instancia por el hipervisor, en el sistema informático, en base a la transmisión de la orden interceptada al hipervisor. En base a la determinación de que no se permite al hipervisor ejecutar la instancia, el código del programa ignora la orden.

En algunas realizaciones de la presente invención, cuando el código del programa obtiene los metadatos, el código del programa descifra una parte de los metadatos vinculados a la imagen del invitado seguro, en donde los metadatos están protegidos en cuanto a su integridad y la parte que comprende una medida criptográfica de una imagen de arranque del invitado seguro se cifró mediante una clave obtenida utilizando una clave privada. En algunas realizaciones de la presente invención, la parte cifrada de los metadatos comprende uno o más controles. Cada control del uno o más controles puede comprender una restricción ambiental. En algunas realizaciones de la presente invención, las restricciones ambientales se seleccionan del grupo que consiste en: sistemas configurados para realizar mediciones de hardware y sistemas configurados para usar una clave de anfitrión no específica del sistema.

En algunas realizaciones de la presente invención, la clave privada es propiedad del control de interfaz segura y se usa exclusivamente por el control de interfaz segura.

En algunas realizaciones de la presente invención, la clave obtenida utilizando la clave privada se comparte entre el control de interfaz segura y el propietario.

En algunas realizaciones de la presente invención, los metadatos comprenden valores obtenidos de una imagen de arranque del invitado seguro calculada utilizando una función unidireccional resistente a colisiones.

En algunas realizaciones de la presente invención, el uno o más controles comprende cada uno una designación positiva o una designación negativa para unas configuraciones del sistema dadas, donde la designación positiva indica que se permite a la instancia del invitado seguro ejecutarse en el sistema informático que comprende la configuración del sistema dada, y la designación negativa indica que no se permite a la instancia del invitado seguro ejecutarse en el sistema informático que comprende la configuración del sistema dada.

En algunas realizaciones de la presente invención, los metadatos son inaccesibles para la instancia del invitado seguro.

En algunas realizaciones de la presente invención, el código de programa que determina si se permite al hipervisor ejecutar la instancia comprende además: el código de programa que identifica, en el uno o más controles, un control relevante para una configuración de la una o más configuraciones del sistema; y el código de programa que determina si el control permite o restringe la ejecución de la instancia, en base al control.

En algunas realizaciones de la presente invención, el código del programa monitoriza la uno o más configuraciones del sistema durante el tiempo de ejecución de la instancia. El código del programa determina que al menos una configuración de la una o más configuraciones cambió durante el tiempo de ejecución. El código del programa identifica un control dado del uno o más controles relevantes para la al menos una ejecución. El código del programa determina, en base a la al menos una configuración y al control dado, si se permite al hipervisor ejecutar la instancia. En base a la determinación de que no se permite al hipervisor ejecutar la instancia, el código del programa puede terminar, mediante el hipervisor, la instancia.

En algunas realizaciones de la presente invención, el código de programa que intercepta la orden por el hipervisor para iniciar la instancia del invitado seguro a partir de la imagen del invitado seguro comprende además: el código del programa que realiza una verificación de integridad en los metadatos; y, en base a completar con éxito la verificación de integridad, el código del programa que lee el uno o más controles en los metadatos.

Haciendo referencia ahora a la FIG. 4, un esquema de un ejemplo de un nodo informático, que puede ser un nodo informático 10 en la nube. El nodo informático 10 en la nube es solo un ejemplo de un nodo informático en la nube adecuado y no pretende sugerir ninguna limitación en cuanto al alcance de uso o la funcionalidad de las realizaciones de la invención descritas en el presente documento. Independientemente, el nodo 10 informático en la nube es capaz de implementarse y/o realizar cualquiera de las funcionalidades establecidas anteriormente. En una realización de la presente invención, el invitado seguro 210 (FIG. 2), el control de interfaz segura 230 (FIG. 2) y/o el hipervisor 220 (FIG. 2) puede entenderse cada uno como que se ejecutan en un nodo informático 10 en la nube (FIG. 4) y si no en un nodo informático 10 en la nube, entonces uno o más nodos informáticos generales que incluyen aspectos del nodo informático 10 en la nube.

En el nodo informático 10 en la nube hay un sistema/servidor informático 12, que es operativo con muchos otros entornos o configuraciones de sistemas informáticos de propósito general o de propósito especial. Ejemplos de sistemas, entornos y/o configuraciones informáticos bien conocidos que pueden ser adecuados para su uso con el sistema/servidor informático 12 incluyen, pero no se limitan a, sistemas informáticos personales, sistemas informáticos de servidor, clientes ligeros, clientes pesados, dispositivos de mano o portátiles, sistemas multiprocesadores, sistemas basados en microprocesadores, decodificadores, electrónica de consumo programable, ordenadores de red, sistemas

de miniordenadores, sistemas informáticos de ordenador central y entornos informáticos en la nube distribuidos que incluyen cualquiera de los sistemas o dispositivos anteriores., y similares.

El sistema/servidor informático 12 puede describirse en el contexto general de las instrucciones ejecutables por un sistema informático, tales como módulos de programa, que se ejecutan por un sistema informático. Generalmente, los módulos de programa pueden incluir rutinas, programas, objetos, componentes, lógica, estructuras de datos, etc., que realizan tareas particulares o implementan tipos de datos abstractos particulares. El sistema/servidor informático 12 se puede practicar en entornos informáticos en la nube distribuidos donde las tareas se realizan mediante dispositivos de procesamiento remotos que están vinculados a través de una red de comunicaciones. En un entorno informático en la nube distribuido, los módulos de programa pueden estar ubicados en medios de almacenamiento de sistemas informáticos tanto locales como remotos, incluidos dispositivos de almacenamiento de memoria.

Como se muestra en la FIG. 4, el sistema/servidor informático 12 que puede utilizarse como nodo informático 10 en la nube, se muestra en forma de un dispositivo informático de uso general. Los componentes del sistema/servidor informático 12 pueden incluir, pero no se limitan a, uno o más procesadores o unidades de procesamiento 16, una memoria del sistema 28 y un bus 18 que acopla varios componentes del sistema, incluida la memoria del sistema 28, al procesador 16.

El bus 18 representa uno o más de cualquiera de varios tipos de estructuras de bus, incluyendo un bus de memoria o un controlador de memoria, un bus periférico, un puerto de gráficos acelerado y un procesador o bus local que utiliza cualquiera de una variedad de arquitecturas de bus. A modo de ejemplo, y no de limitación, tales arquitecturas incluyen el bus de la Arquitectura Estándar de la Industria (ISA), el bus de Arquitectura Micro Canal (MCA), el bus de ISA mejorado (EISA), el bus local de la Asociación de Estándares de Electrónica de Vídeo (VESA) y el bus de Interconexión de Componentes Periféricos (PCI).

El sistema/servidor informático 12 normalmente incluye una variedad de medios legibles por el sistema informático. Dichos medios pueden ser cualquier medio disponible que sea accesible por el sistema/servidor informático 12, e incluyen medios volátiles y no volátiles, medios extraíbles y no extraíbles.

La memoria del sistema 28 puede incluir medios legibles por el sistema informático en forma de memoria volátil, tal como la memoria de acceso aleatorio (RAM) 30 y/o la memoria caché 32. El sistema/servidor informático 12 puede incluir además otros medios de almacenamiento de sistema informático extraíbles/no extraíbles, volátiles/no volátiles. Solo a modo de ejemplo, el sistema de almacenamiento 34 puede proporcionarse para leer y escribir en un medio magnético no extraíble y no volátil (no mostrado y normalmente denominado «disco duro»). Aunque no se muestra, se pueden proporcionar una unidad de disco magnético para leer de y escribir en un disco magnético extraíble no volátil *(p. ej.,* un «disco flexible») y una unidad de disco óptico para leer de o escribir en un disco óptico extraíble no volátil, tal como un CD-ROM, DVD-ROM u otros medios ópticos. En tales casos, cada uno puede conectarse al bus 18 mediante una o más interfaces de medios de datos. Como se representará y describirá adicionalmente a continuación, la memoria 28 puede incluir al menos un producto de programa que tiene un conjunto *(p. ej.,* al menos uno) de módulos de programa que están configurados para llevar a cabo las funciones de las realizaciones de la invención.

El programa/utilidad 40, que tiene un conjunto (al menos uno) de módulos de programa 42, puede almacenarse en la memoria 28 a modo de ejemplo, y no de limitación, así como un sistema operativo, uno o más programas de aplicaciones, otros módulos de programa y datos de programa. Cada uno del sistema operativo, uno o más programas de aplicaciones, otros módulos de programa y datos de programa o alguna combinación de los mismos, pueden incluir una implementación de un entorno de interconexión de redes. Los módulos de programa 42 generalmente llevan a cabo las funciones y/o metodologías de las realizaciones de la invención como se describe en el presente documento.

El sistema/servidor informático 12 también puede comunicarse con uno o más dispositivos externos 14, tales como un teclado, un dispositivo señalador, una pantalla 24, etc.; uno o más dispositivos que permiten a un usuario interactuar con el sistema/servidor informático 12; y/o cualquier dispositivo *(p. ej., tarjeta de red, módem,* etc.) que permita al sistema/servidor informático 12 comunicarse con uno o más de otros dispositivos informáticos. tal comunicación puede ocurrir a través de las interfaces 22 de Entrada/Salida (E/S). Aún así, el sistema/servidor informático 12 puede comunicarse con una o más redes, tales como una red de área local (LAN), una red de área amplia general (WAN) y/o una red pública (por ejemplo, Internet) a través del adaptador de red 20. Como se representa, el adaptador de red 20 se comunica con los otros componentes del sistema/servidor informático 12 a través del bus 18. Se debería entender que, aunque no se muestran, podrían usarse otros componentes de hardware y/o software junto con el sistema/servidor informático 12. Los ejemplos incluyen, pero no se limitan a: microcódigo, controladores de dispositivo, unidades de procesamiento redundantes, agrupaciones de unidades de disco externas, sistemas RAID, unidades de cinta y sistemas de almacenamiento de archivos de datos, etc.

Debe entenderse que, aunque esta divulgación incluye una descripción detallada de la informática en la nube, la implementación de las enseñanzas mencionadas en el presente documento no se limita a un entorno informático en la nube. Más bien, las realizaciones de la presente invención son capaces de ser implementadas junto con cualquier otro tipo de entorno informático conocido ahora o desarrollado posteriormente.

La informática en la nube es un modelo de prestación de servicios para permitir un acceso de red cómodo y bajo demanda a un conjunto compartido de recursos informáticos configurables (p. ej., redes, ancho de banda de red, servidores, procesamiento, memoria, almacenamiento, aplicaciones, máquinas virtuales y servicios) que se pueden aprovisionar y lanzar rápidamente con un mínimo esfuerzo de gestión o interacción con un proveedor del servicio. Este modelo en la nube puede incluir al menos cinco características, al menos tres modelos de servicio y al menos cuatro modelos de implementación.

Las características son como sigue:

Autoservicio bajo demanda: un consumidor en la nube puede aprovisionar de forma unilateral capacidades informáticas, tales como la hora del servidor y el almacenamiento de red, según sea necesario de forma automática, sin requerir interacción humana con el proveedor del servicio.

Amplio acceso a la red: las capacidades están disponibles sobre una red y se accede a ellas a través de mecanismos estándar que promueven el uso por plataformas heterogéneas de clientes ligeros o pesados *(p. ej.,* teléfonos móviles, ordenadores portátiles y PDA). Agrupación de recursos: los recursos informáticos del proveedor se agrupan para servir a múltiples consumidores usando un modelo de múltiples inquilinos, con diferentes recursos físicos y virtuales asignados dinámicamente y reasignados según la demanda. Existe una sensación de independencia de ubicación en el sentido de que el consumidor generalmente no tiene control ni conocimiento sobre la ubicación exacta de los recursos proporcionados, pero puede ser capaz de especificar la ubicación en un nivel de abstracción superior *(p. ej.,* país, estado o centro de datos). Elasticidad rápida: las capacidades se pueden aprovisionar rápida y elásticamente, en algunos casos de manera automática, para poner a escala rápidamente y liberarlas rápidamente para escalarlas rápidamente. Para el consumidor, las capacidades disponibles para el aprovisionamiento suelen parecer ilimitadas y pueden adquirirse en cualquier cantidad en cualquier momento.

Servicio medido: los sistemas en la nube controlan y optimizan automáticamente el uso de los recursos al aprovechar una capacidad de medición en algún nivel de abstracción apropiado para el tipo de servicio *(p. ej.,* almacenamiento, procesamiento, ancho de banda y cuentas de usuario activas). El uso de los recursos se puede monitorizar, controlar e informar, proporcionando transparencia tanto para el proveedor como para el consumidor del servicio utilizado.

Los modelos de servicio son como sigue:

Software como servicio (SaaS): la capacidad que se proporciona al consumidor es para utilizar las aplicaciones del proveedor que se ejecutan en una infraestructura en la nube. Las aplicaciones son accesibles desde varios dispositivos cliente a través de una interfaz de cliente ligero, tal como un navegador web *(p. ej.,* correo electrónico basado en web). El consumidor no gestiona ni controla la infraestructura de nube subyacente, incluida la red, los servidores, los sistemas operativos, el almacenamiento o incluso las capacidades de aplicaciones individuales, con la posible excepción de ajustes limitados de configuración de aplicaciones específicas del usuario.

Plataforma como servicio (PaaS): la capacidad proporcionada al consumidor es para implementar en la infraestructura en la nube, aplicaciones aF1 creadas o adquiridas por el consumidor, creadas usando lenguajes de programación y herramientas soportadas por el proveedor. El consumidor no gestiona ni controla la infraestructura en la nube subyacente, incluidas las redes, los servidores, los sistemas operativos o el almacenamiento, pero tiene control sobre las aplicaciones implementadas y, posiblemente, sobre las configuraciones del entorno del alojamiento de aplicaciones.

Infraestructura como Servicio (IaaS): la capacidad proporcionada al consumidor es para aprovisionar procesamiento, almacenamiento, redes y otros recursos informáticos fundamentales donde el consumidor es capaz de implementar y ejecutar software arbitrario, que puede incluir sistemas operativos y aplicaciones. El consumidor no gestiona ni controla la infraestructura en la nube subyacente, pero tiene control sobre los sistemas operativos, el almacenamiento, las aplicaciones implementadas y, posiblemente, un control limitado de componentes de interconexión de redes seleccionados *(p. ej.,* los firewalls del anfitrión).

Los modelos de implementación son como sigue:

Nube privada: la infraestructura en la nube se opera únicamente para una organización. Puede estar gestionada por la organización o por un tercero y puede existir en las instalaciones o fuera de las instalaciones.

Nube comunitaria: la infraestructura en la nube es compartida por varias organizaciones y soporta una comunidad específica que tiene intereses compartidos (p. ej., misión, requisitos de seguridad, políticas y consideraciones de cumplimiento). Se puede gestionar por las organizaciones o un tercero y puede existir en las instalaciones o fuera de las instalaciones.

Nube pública: la infraestructura en la nube se pone a disposición del público en general o de un gran grupo industrial y es propiedad de una organización que vende servicios en la nube.

Nube híbrida: la infraestructura de en la nube es una composición de dos o más nubes (privada, comunitaria o pública) que siguen siendo entidades únicas, pero están unidas entre sí por una tecnología estandarizada o propietaria que

permite la portabilidad de datos y aplicaciones (p. ej., la ráfaga en la nube para equilibrar la carga entre nubes).

Un entorno informático en la nube está orientado a servicio con un enfoque en ausencia de estado, bajo acoplamiento, modularidad e interoperabilidad semántica. En el corazón de la informática en la nube hay una infraestructura que incluye una red de nodos interconectados.

Haciendo referencia ahora a la FIG. 5, se representa el entorno informático en la nube 50 ilustrativo. Como se muestra, el entorno informático en la nube 50 incluye uno o más nodos informáticos en la nube 10 con los que pueden comunicarse los dispositivos informáticos locales utilizados por los consumidores en la nube, tales como, por ejemplo, un asistente digital personal (PDA) o un teléfono celular 54A, un ordenador de sobremesa 54B, un ordenador portátil 54C y/o un sistema informático de automóvil 54N. Los nodos 10 pueden comunicarse unos con otros. Se pueden agrupar (no mostrado) física o virtualmente, en una o más redes, tales como nubes privadas, comunitarias, públicas o híbridas, como se ha descrito anteriormente, o una combinación de las mismas. Esto permite que el entorno informático en la nube 50 ofrezca infraestructura, plataformas y/o software como servicios para los que un consumidor en la nube no necesita mantener los recursos en un dispositivo informático local. Se entiende que los tipos de dispositivos informáticos 54A-N mostrados en la FIG. 5 se pretende que sean únicamente ilustrativos y que los nodos informáticos 10 y el entorno informático en la nube 50 pueden comunicarse con cualquier tipo de dispositivo informatizado sobre cualquier tipo de red y/o conexión direccionable de red (p. ej., utilizando un navegador web).

Haciendo referencia ahora a la FIG. 6, se muestra un conjunto de capas de abstracción funcionales proporcionadas por el entorno informático en la nube 50 (FIG. 5). Se debería entender de antemano que los componentes, las capas y las funciones mostrados en la FIG. 6 se pretende que sean únicamente ilustrativos y las realizaciones de la invención no se limitan a los mismos. Como se representa, se proporcionan las siguientes capas y funciones correspondientes:

La capa de hardware y software 60 incluye componentes de hardware y software. Ejemplos de componentes de hardware incluyen: ordenadores centrales 61; servidores basados en la arquitectura RISC (Ordenador de Conjunto de Instrucciones Reducido) 62; servidores 63; servidores blade 64; dispositivos de almacenamiento 65; y redes y componentes de interconexión de redes 66. En algunas realizaciones, los componentes de software incluyen software de servidor de aplicaciones de red 67 y el software de base de datos 68.

La capa de virtualización 70 proporciona una capa de abstracción a partir de la que se pueden proporcionar los siguientes ejemplos de entidades virtuales: servidores virtuales 71; almacenamiento virtual 72; redes virtuales 73, incluyendo redes privadas virtuales; aplicaciones virtuales y sistemas operativos 74; y clientes virtuales 75.

En un ejemplo, la capa de gestión 80 puede proporcionar las funciones descritas a continuación. El aprovisionamiento de recursos 81 proporciona una adquisición dinámica de recursos informáticos y otros recursos que se utilizan para realizar tareas dentro del entorno informático en la nube. La Medición y Fijación de precios 82 proporcionan un seguimiento de los costes a medida que se utilizan los recursos dentro del entorno informático en la nube, y el cobro o facturación por el consumo de estos recursos. En un ejemplo, estos recursos pueden incluir licencias de software de aplicaciones. La seguridad proporciona verificación de identidad para los consumidores y las tareas en la nube, así como protección para los datos y otros recursos. El portal de usuario 83 proporciona acceso al entorno informático en la nube para consumidores y administradores de sistemas. La gestión de nivel de servicio 84 proporciona la asignación y gestión de recursos informáticos en la nube de manera que se cumplan los niveles de servicio requeridos. La planificación y el cumplimiento del Acuerdo de Nivel de Servicio (SLA) 85 proporcionan disposición previa para, y adquisición de, recursos informáticos en la nube para los que se anticipa un requisito futuro de acuerdo con un SLA.

La capa de cargas de trabajo 90 proporciona ejemplos de funcionalidad para lo cual se puede utilizar el entorno informático en la nube. Ejemplos de cargas de trabajo y funciones que se pueden proporcionar desde esta capa incluyen: mapeo y navegación 91; desarrollo de software y gestión del ciclo de vida 92; distribución de educación en el aula virtual 93; procesamiento de análisis de datos 94; procesamiento de transacciones 95; y control de la ejecución de un invitado seguro, en base a los factores ambientales, a través de un control de interfaz segura 96.

La presente invención puede ser un sistema, un método y/o un producto de programa informático en cualquier posible nivel de detalle técnico de integración. El producto de programa informático puede incluir un medio (o medios) de almacenamiento legible por ordenador que tiene instrucciones de programa legibles por ordenador en el mismo para hacer que un procesador lleve a cabo aspectos de la presente invención.

El medio de almacenamiento legible por ordenador puede ser un dispositivo tangible que puede retener y almacenar instrucciones para su uso por un dispositivo de ejecución de instrucciones. El medio de almacenamiento legible por ordenador puede ser, por ejemplo, pero no se limita a, un dispositivo de almacenamiento electrónico, un dispositivo de almacenamiento magnético, un dispositivo de almacenamiento óptico, un dispositivo de almacenamiento electromagnético, un dispositivo de almacenamiento de semiconductores o cualquier combinación adecuada de los anteriores. Una lista no exhaustiva de ejemplos más específicos del medio de almacenamiento legible por ordenador incluye lo siguiente: un disquete de ordenador portátil, un disco duro, una memoria de acceso aleatorio (RAM), una memoria de solo lectura (ROM), una memoria de solo lectura programable y borrable (EPROM o memoria Flash), una memoria estática de acceso aleatorio (SRAM), un disco compacto portátil de memoria de solo lectura (CD-ROM), un disco versátil digital (DVD), una tarjeta de memoria, un disco flexible, un dispositivo codificado mecánicamente, tal

como tarjetas perforadas o estructuras en relieve en un surco que tiene instrucciones grabadas en el mismo, y cualquier combinación adecuada de los anteriores. Un medio de almacenamiento legible por ordenador, como se usa en el presente documento, no debe interpretarse como señales transitorias en sí, tales como ondas de radio u otras ondas electromagnéticas que se propagan libremente, ondas electromagnéticas que se propagan a través de una guía de ondas u otros medios de transmisión *(p. ej.,* pulsos de luz que pasan a través de un cable de fibra óptica) o señales eléctricas transmitidas a través de un cable.

Las instrucciones del programa legibles por ordenador descritas en el presente documento pueden descargarse a los respectivos dispositivos informáticos/de procesamiento desde un medio de almacenamiento legible por ordenador o a un ordenador externo o dispositivo de almacenamiento externo a través de una red, por ejemplo, Internet, una red de área local, una red de área amplia y/o una red inalámbrica. La red puede comprender cables de transmisión de cobre, fibras de transmisión óptica, transmisión inalámbrica, enrutadores, firewalls, conmutadores, ordenadores de puerta de enlace y/o servidores periféricos. Una tarjeta adaptadora de red o interfaz de red en cada dispositivo informático/de procesamiento recibe instrucciones de programa legibles por ordenador desde la red y envía las instrucciones de programa legibles por ordenador para su almacenamiento en un medio de almacenamiento legible por ordenador dentro del dispositivo informático/de procesamiento respectivo.

Las instrucciones de programa legibles por ordenador para llevar a cabo las operaciones de la presente invención pueden ser instrucciones de ensamblador, instrucciones de arquitectura de conjunto de instrucciones (ISA), instrucciones de máquina, instrucciones dependientes de la máquina, microcódigo, instrucciones de firmware, datos de ajuste de estado, datos de configuración para circuitería integrada o código fuente o código objeto escritos en cualquier combinación de uno o más lenguajes de programación, incluido un lenguaje de programación orientado a objetos tales como Smalltalk, C++ o similares, y lenguajes de programación procedimentales, tal como el lenguaje de programación «C» o lenguajes de programación similares. Las instrucciones del programa legibles por ordenador pueden ejecutarse completamente en el ordenador del usuario, en parte en el ordenador del usuario, como un paquete de software independiente, en parte en el ordenador del usuario y en parte en un ordenador remoto o completamente en el ordenador o servidor remoto. En el último escenario, el ordenador remoto puede estar conectado al ordenador del usuario a través de cualquier tipo de red, incluida una red de área local (LAN) o una red de área amplia (WAN), o la conexión puede realizarse a un ordenador externo (por ejemplo, a través de Internet utilizando un proveedor de servicios de Internet). En algunas realizaciones, la circuitería electrónica que incluye, por ejemplo, circuitería lógica programable, matrices de puertas programables en campo (FPGA) o matrices lógicas programables (PLA) pueden ejecutar las instrucciones del programa legibles por ordenador utilizando la información de estado de las instrucciones del programa legibles por ordenador para personalizar la circuitería electrónica, con el fin de realizar aspectos de la presente invención.

Los aspectos de la presente invención se describen en el presente documento con referencia a ilustraciones de diagrama de flujo y/o diagramas de bloques de métodos, aparatos (sistemas) y productos de programas informáticos de acuerdo con las realizaciones de la invención. Se entenderá que cada bloque de las ilustraciones del diagrama de flujo y/o los diagramas de bloques, y las combinaciones de bloques en las ilustraciones del diagrama de flujo y/o los diagramas de bloques, pueden implementarse mediante instrucciones de programa legibles por ordenador.

Estas instrucciones de programa legibles por ordenador pueden proporcionarse a un procesador de un ordenador de propósito general, ordenador de propósito especial u otro aparato de procesamiento de datos programable para producir una máquina, de modo que las instrucciones, que se ejecutan a través del procesador del ordenador u otro aparato de procesamiento de datos programable, creen medios para implementar las funciones/actos especificados en el diagrama de flujo y/o bloque o bloques del diagrama de bloques. Estas instrucciones de programa legibles por ordenador también pueden almacenarse en un medio de almacenamiento legible por ordenador que puede dirigir un ordenador, un aparato de procesamiento de datos programable y/u otros dispositivos para que funcionen de una manera particular, de modo que el medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo comprende un artículo de fabricación que incluye instrucciones que implementan aspectos de la función/acto especificados en el diagrama de flujo y/o bloque o bloques del diagrama de bloques.

Las instrucciones del programa legibles por ordenador también pueden cargarse en un ordenador, otro aparato de procesamiento de datos programable u otro dispositivo para provocar que se realicen una serie de etapas operativas en el ordenador, otro aparato programable u otro dispositivo para producir un proceso implementado por ordenador, de modo que las instrucciones que se ejecutan en el ordenador, otro aparato programable u otro dispositivo implementan las funciones/actos especificados en el diagrama de flujo y/o bloque o bloques del diagrama de bloques.

El diagrama de flujo y los diagramas de bloques de las Figuras ilustran la arquitectura, la funcionalidad y la operación de posibles implementaciones de sistemas, métodos y productos de programas informáticos de acuerdo con diversas realizaciones de la presente invención. A este respecto, cada bloque en el diagrama de flujo o los diagramas de bloques puede representar un módulo, segmento o porción de instrucciones, que comprende una o más instrucciones ejecutables para implementar la función o funciones lógicas especificadas. En algunas implementaciones alternativas, las funciones indicadas en los bloques pueden ocurrir fuera del orden indicado en las Figuras. Por ejemplo, dos bloques mostrados en sucesión pueden, de hecho, ejecutarse de manera sustancialmente simultánea, o los bloques pueden ejecutarse a veces en orden inverso, dependiendo de la funcionalidad implicada. También se observará que cada bloque de los diagramas de bloques y/o la ilustración del diagrama de flujo, y las combinaciones de bloques en los

diagramas de bloques y/o la ilustración del diagrama de flujo, pueden implementarse mediante sistemas basados en hardware de propósito especial que realizan las funciones o actos especificados o llevan a cabo combinaciones de hardware de propósito especial e instrucciones de ordenador.

La terminología utilizada en el presente documento tiene el propósito de describir las realizaciones particulares únicamente y no se pretende que sea limitante. Como se usa en el presente documento, las formas singulares «un», «una» y «el» se pretende que incluyan también las formas plurales, a menos que el contexto indique claramente lo contrario. Se entenderá además que los términos «comprende» y/o «que comprende», cuando se usan en esta memoria descriptiva, especifican la presencia de características, números enteros, etapas, operaciones, elementos y/o componentes indicados, pero no excluyen la presencia o adición de una o más características, números enteros, etapas, operaciones, elementos, componentes y/o grupos de los mismos.

Las correspondientes estructuras, materiales, actos y equivalentes de todos los medios o etapas más elementos de función en las reivindicaciones siguientes, si los hay, se pretende que incluyan cualquier estructura, material o acto para realizar la función en combinación con otros elementos reivindicados, como se reivindica específicamente. La descripción de una o más realizaciones se ha presentado con fines ilustrativos y descriptivos, pero no se pretende que sea exhaustiva ni se limite a la forma divulgada. Muchas modificaciones y variaciones resultarán evidentes para los expertos en la técnica. La realización se eligió y describió con el fin de explicar mejor diversos aspectos y la aplicación práctica, y para permitir que otros expertos en la técnica comprendan diversas realizaciones con diversas modificaciones que sean adecuadas para el uso particular contemplado.

REIVINDICACIONES

1. Un método implementado por ordenador, que comprende:

obtener, mediante un control de interfaz segura (230) en un sistema informático, en el que el control de interfaz segura (230) está acoplado comunicativamente a un hipervisor (220), en el que el hipervisor (220) gestiona uno o más invitados, metadatos (240) vinculados a una imagen de un invitado seguro (210) de un propietario y gestionados por el hipervisor (220), en el que los metadatos (240) comprenden uno o más controles (260), en el que cada control del uno o más controles (260) indica al control de interfaz segura (230) si se permite al hipervisor (220) ejecutar una instancia de un invitado seguro (210) generado con la imagen en el sistema informático en base a la presencia o ausencia de una o más configuraciones del sistema (270) en el sistema informático;

interceptar, mediante el control de interfaz segura (230), una orden por el hipervisor (220) para iniciar la instancia del invitado seguro (210) a partir de la imagen del invitado seguro (210);

determinar, mediante el control de interfaz segura (230), la presencia o la ausencia de una o más configuraciones del sistema (270) en el sistema informático;

determinar, mediante el control de interfaz segura (230), en base a uno o más controles (260) y la presencia o ausencia de una o más configuraciones del sistema (270), si se permite al hipervisor (220) ejecutar la instancia;

en base a la determinación de que se permite al hipervisor (220) ejecutar la instancia;

permitir, mediante el control de interfaz segura (230), el inicio de la instancia por el hipervisor (220), en el sistema informático, en base a la transmisión de la orden interceptada al hipervisor (220),

monitorización, mediante el control de interfaz segura (230), durante el tiempo de ejecución de la instancia

iniciada por el hipervisor (220) en respuesta a la habilitación, de la una o más

configuraciones de sistema (270);

determinar, mediante el control de interfaz segura (230), que al menos una configuración de

la una o más configuraciones del sistema cambió durante el tiempo de ejecución;

identificar, mediante el control de interfaz segura (230), en los metadatos (240), un control dado del uno o más controles (260) relevantes para la al menos una configuración que cambió durante el tiempo de ejecución;

determinar, mediante el control de interfaz segura (230), en base a la al menos una configuración que cambió durante el tiempo de ejecución y el control dado, si ya no se permite al hipervisor (220) ejecutar la instancia; y

en base a la determinación de si ya no se permite al hipervisor (220) ejecutar la instancia, provoca, mediante el control de interfaz segura (230), durante el tiempo de ejecución de la instancia, que el hipervisor (220) termine la instancia; y

en base a la determinación de si no se permite al hipervisor (220) ejecutar la instancia;

ignorar, mediante el control de interfaz segura (230), la orden.

2. El método implementado por ordenador de la reivindicación 1, en el que la obtención de los metadatos (240) comprende además:

descifrar, mediante el control de interfaz segura (230), una parte de los metadatos (240) vinculados a la imagen del invitado seguro (210), en donde los metadatos (240) están protegidos en cuanto a su integridad y la parte que comprende una medida criptográfica de una imagen de arranque del invitado seguro (210) se cifró mediante una clave obtenida usando una clave privada.

3. El método implementado por ordenador de la reivindicación 1 o 2, en el que la parte cifrada de los metadatos (240) comprende el uno o más controles (260).

4. El método implementado por ordenador de una de las reivindicaciones 1 a 3, en el que cada control del uno o más controles (260) comprende una restricción ambiental.

5. El método implementado por ordenador de la reivindicación 4, en el que las restricciones ambientales se seleccionan del grupo que consiste en: sistemas configurados para realizar mediciones de hardware, y sistemas configurados para usar una clave de anfitrión no específica del sistema.

6. El método implementado por ordenador de la reivindicación 2, en el que la clave privada es propiedad del control de interfaz segura (230) y se usa exclusivamente por el control de interfaz segura (230).

7. El método implementado por ordenador de la reivindicación 6, en el que la clave obtenida utilizando la clave privada se comparte entre el control de interfaz segura (230) y el propietario.

8. El método implementado por ordenador de una de las reivindicaciones 1 a 7, en el que los metadatos (240) comprenden valores obtenidos de una imagen de arranque del invitado seguro (210) calculados utilizando una función unidireccional resistente a colisiones.

9. El método implementado por ordenador de una de las reivindicaciones 1 a 8, en el que el uno o más controles (260) comprende cada uno una designación positiva o una designación negativa para unas configuraciones del sistema (270) dadas, en donde la designación positiva indica que se permite a la instancia del invitado seguro (210) ejecutarse en el sistema informático que comprende la configuración del sistema (270) dada, y la designación negativa indica que no se permite a la instancia del invitado seguro (210) ejecutarse en el sistema informático que comprende la configuración del sistema (270) dada.

10. El método implementado por ordenador de una de las reivindicaciones 1 a 9, en el que los metadatos (240) son inaccesibles para la instancia del invitado seguro (210).

11. El método implementado por ordenador de una de las reivindicaciones 1 a 10, en el que la determinación de si se permite al hipervisor (220) ejecutar la instancia comprende además:

identificar, mediante el control de interfaz segura (230), en el uno o más controles (260), un control relevante para una configuración de la una o más configuraciones del sistema (270); y

determinar, mediante el control de interfaz segura (230), si el control permite o restringe la ejecución de la instancia, en base al control.

12. El método implementado por ordenador de una de las reivindicaciones 1 a 11, en el que la intercepción de la orden por el hipervisor (220) para iniciar la instancia del invitado seguro (210) a partir de la imagen del invitado seguro (210) comprende además:

realizar, mediante un control de interfaz segura (230), una verificación de integridad de los metadatos (240); y

en base a completar con éxito la verificación de integridad, leer, mediante el control de interfaz segura (230), el uno o más controles (260) en los metadatos (240).

13. Un producto de programa informático que comprende:

un medio de almacenamiento legible por ordenador legible por uno o más procesadores (16) y que almacena instrucciones para su ejecución por uno o más procesadores (16) para realizar un método que comprende:

obtener, mediante el uno o más procesadores (16) en un sistema informático, en el que el uno o más procesadores (16) están acoplados comunicativamente a un hipervisor (220), en el que el hipervisor (220) gestiona uno o más invitados, metadatos (240) vinculados a una imagen de un invitado seguro (210) de un propietario y gestionados por el hipervisor (220), en el que los metadatos (240) comprenden uno o más controles (260), en el que cada control del uno o más controles (260) indica al uno o más procesadores (16) si se permite al hipervisor (220) ejecutar una instancia de un invitado seguro (210) generado con la imagen en el sistema informático en base a la presencia o ausencia de una o más configuraciones del sistema (270) en el sistema informático;

interceptar, mediante uno o más procesadores (16), una orden por el hipervisor (220) para iniciar la instancia del invitado seguro (210) a partir de la imagen del invitado seguro (210);

determinar, mediante el uno o más procesadores (16), la presencia o la ausencia de la una o más configuraciones del sistema (270) en el sistema informático;

determinar, mediante el uno o más procesadores (16), en base al uno o más controles (260) y la presencia o ausencia de la una o más configuraciones del sistema (270), si se permite al hipervisor (220) ejecutar la instancia;

en base a la determinación de que se permite al hipervisor (220) ejecutar la instancia;

permitir, mediante el uno o más procesadores (16), el inicio de la instancia por el hipervisor (220), en el sistema informático, en base a la transmisión de la orden interceptada al hipervisor (220);

monitorizar, mediante el control de interfaz segura (230), durante el tiempo de ejecución de la instancia

iniciada por el hipervisor (220) en respuesta a la habilitación, la una o más configuraciones

del sistema (270);

determinar, mediante el control de interfaz segura (230), que al menos una configuración de la una o más configuraciones del sistema cambió durante el tiempo de ejecución;

identificar, mediante el control de interfaz segura (230), en los metadatos (240), un control dado del uno o más controles (260) relevantes para la al menos una configuración que cambió durante el tiempo de ejecución;

determinar, mediante el control de interfaz segura (230), en base a la al menos una configuración que cambió durante el tiempo de ejecución y el control dado, si ya no se permite al hipervisor (220) ejecutar la instancia; y

en base a la determinación de si ya no se permite al hipervisor (220) ejecutar la instancia, provocar, mediante el control de interfaz segura (230), durante el tiempo de ejecución de la instancia, el que hipervisor (220) termine la instancia; y

en base a la determinación de si no se permite al hipervisor (220) ejecutar la instancia;

ignorar, por el uno o más procesadores (16), la orden.

14. El producto de programa informático de la reivindicación 13, en el que la obtención de los metadatos (240) comprende además:

descifrar, mediante el uno más procesadore (16), una parte de los metadatos (240) vinculados a la imagen del invitado seguro (210), en donde los metadatos (240) están protegidos en cuanto a su integridad y la parte que comprende una medida criptográfica de una imagen de arranque del invitado seguro (210) se cifró mediante una clave obtenida usando una clave privada.

15. El producto de programa informático de la reivindicación 14, en el que la parte cifrada de los metadatos (240) comprende el uno o más controles (260).

16. El producto de programa informático de una de las reivindicaciones 13 a 15, en el que cada control del uno o más controles (260) comprende una restricción ambiental.

17. El producto de programa informático de la reivindicación 16, en el que las restricciones ambientales se seleccionan del grupo que consiste en: sistemas configurados para realizar mediciones de hardware y sistemas configurados para usar una clave de anfitrión no específica del sistema.

18. Un sistema informático que comprende:

una memoria (28);

uno o más procesadores (16) en comunicación con la memoria (28);

instrucciones de programa ejecutables por el uno o más procesadores (16) a través de la memoria (28) para realizar un método, comprendiendo el método:

obtener, mediante el uno o más procesadores (16) en el sistema informático, en el que el uno o más procesadores (16) están acoplados comunicativamente a un hipervisor (220), en el que el hipervisor (220) gestiona uno o más invitados, metadatos (240) vinculados a una imagen de un invitado seguro (210) de un propietario y gestionados por el hipervisor (220), en el que los metadatos (240) comprenden uno o más controles (260), en el que cada control del uno o más controles (260) indica al uno o más procesadores (16) si se permite al hipervisor (220) ejecutar una instancia de un invitado seguro (210) generado con la imagen en el sistema informático

en base a la presencia o ausencia de una o más configuraciones del sistema (270) en el sistema informático;

interceptar, mediante el uno o más procesadores (16), una orden por el

hipervisor (220) para iniciar la instancia del invitado seguro (210) a partir de la imagen del invitado seguro (210);

determinar, mediante el uno o más procesadores (16), la presencia o la ausencia de la una o más configuraciones del sistema (270) en el sistema informático;

determinar, mediante el uno o más procesadores (16), en base al uno o más controles (260) y la presencia o ausencia de la una o más configuraciones del sistema (270), si se permite al hipervisor (220) ejecutar la instancia;

en base a la determinación de que se permite al hipervisor (220) ejecutar la instancia;

permitir, mediante el uno o más procesadores (16), el inicio de la

instancia por el hipervisor (220), en el sistema informático, en base a la transmisión de la orden interceptada al hipervisor (220);

monitorizar, mediante el control de interfaz segura (230), durante el tiempo de ejecución de la instancia iniciada por el hipervisor (220) en respuesta a la habilitación, la una o más configuraciones del sistema (270);

determinar, mediante el control de interfaz segura (230), que al menos una

configuración de la una o más configuraciones del sistema cambió durante el tiempo de ejecución;

identificar, mediante el control de interfaz segura (230), en los metadatos (240), un control dado del uno o más controles (260) relevantes para la al menos una configuración que cambió durante el tiempo de ejecución;

determinar, mediante el control de interfaz segura (230), en base a la al

menos una configuración que cambió durante el tiempo de ejecución y el control dado, si ya no se permite al hipervisor (220) ejecutar la instancia; y

en base a la determinación de que ya no se permite al hipervisor (220) ejecutar la instancia, provocar, mediante el control de interfaz segura (230), durante el tiempo de ejecución de la instancia, que el hipervisor (220) termine la instancia; y

en base a la determinación de que no se permite al hipervisor (220) ejecutar la instancia;

ignorar, por el uno o más procesadores (16), la orden.

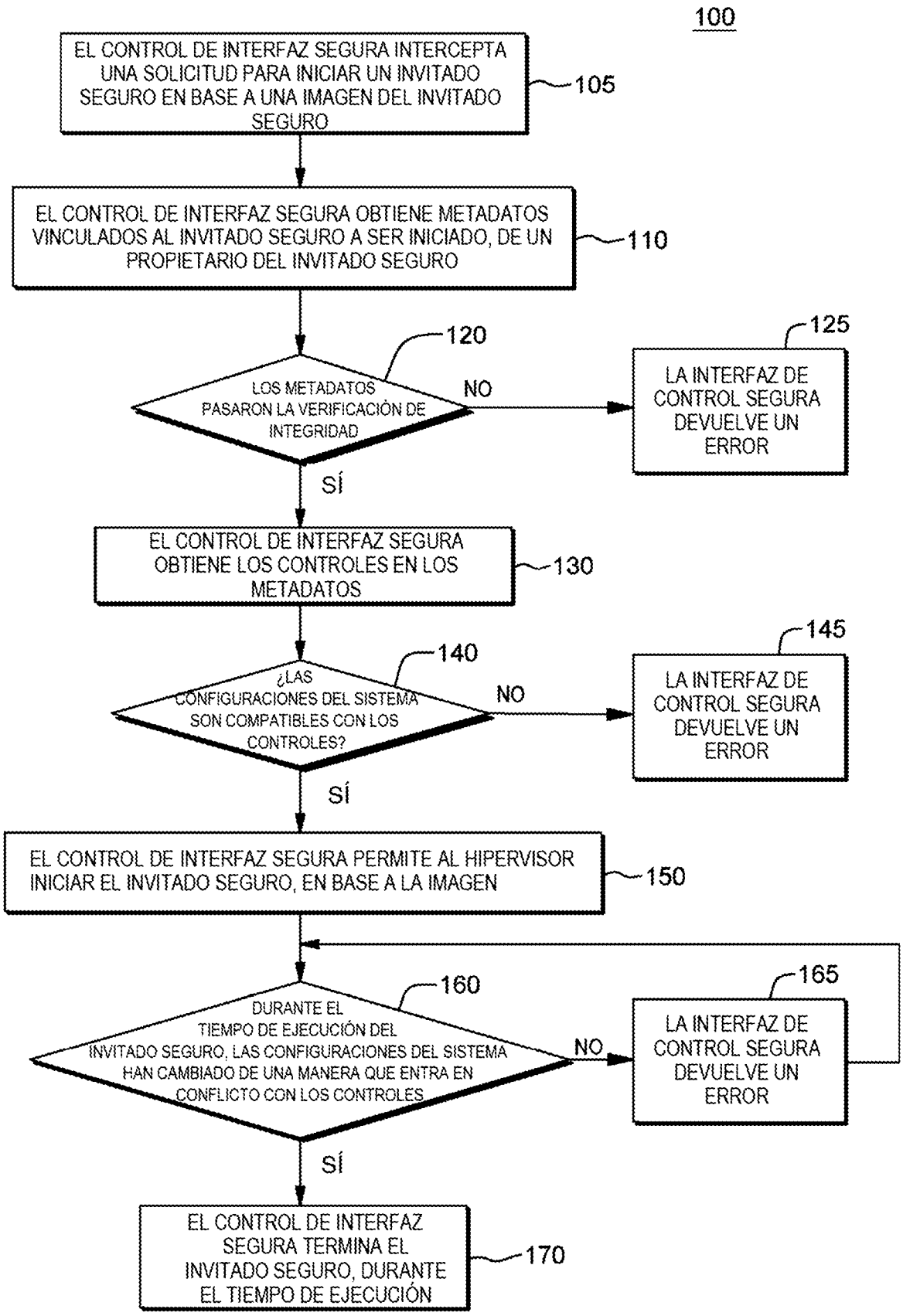
100
EL CONTROL DE INTERFAZ SEGURA INTERCEPTA UNA SOLICITUD PARA INICIAR UN INVITADO SEGURO EN BASE A UNA IMAGEN DEL INVITADO SEGURO
105
EL CONTROL DE INTERFAZ SEGURA OBTIENE METADATOS VINCULADOS AL INVITADO SEGURO A SER INICIADO, DE UN PROPIETARIO DEL INVITADO SEGURO
110
120
LOS METADATOS PASARON LA VERIFICACIÓN DE INTEGRIDAD
NO
125
LA INTERFAZ DE CONTROL SEGURA DEVUELVE UN ERROR
SÍ
EL CONTROL DE INTERFAZ SEGURA OBTIENE LOS CONTROLES EN LOS METADATOS
130
140
¿LAS CONFIGURACIONES DEL SISTEMA SON COMPATIBLES CON LOS CONTROLES?
NO
145
LA INTERFAZ DE CONTROL SEGURA DEVUELVE UN ERROR
SÍ
EL CONTROL DE INTERFAZ SEGURA PERMITE AL HIPERVISOR INICIAR EL INVITADO SEGURO, EN BASE A LA IMAGEN
150
160
DURANTE EL TIEMPO DE EJECUCIÓN DEL INVITADO SEGURO, LAS CONFIGURACIONES DEL SISTEMA HAN CAMBIADO DE UNA MANERA QUE ENTRA EN CONFLICTO CON LOS CONTROLES
NO
165
LA INTERFAZ DE CONTROL SEGURA DEVUELVE UN ERROR
SÍ
EL CONTROL DE INTERFAZ SEGURA TERMINA EL INVITADO SEGURO, DURANTE EL TIEMPO DE EJECUCIÓN
170

FIG. 1

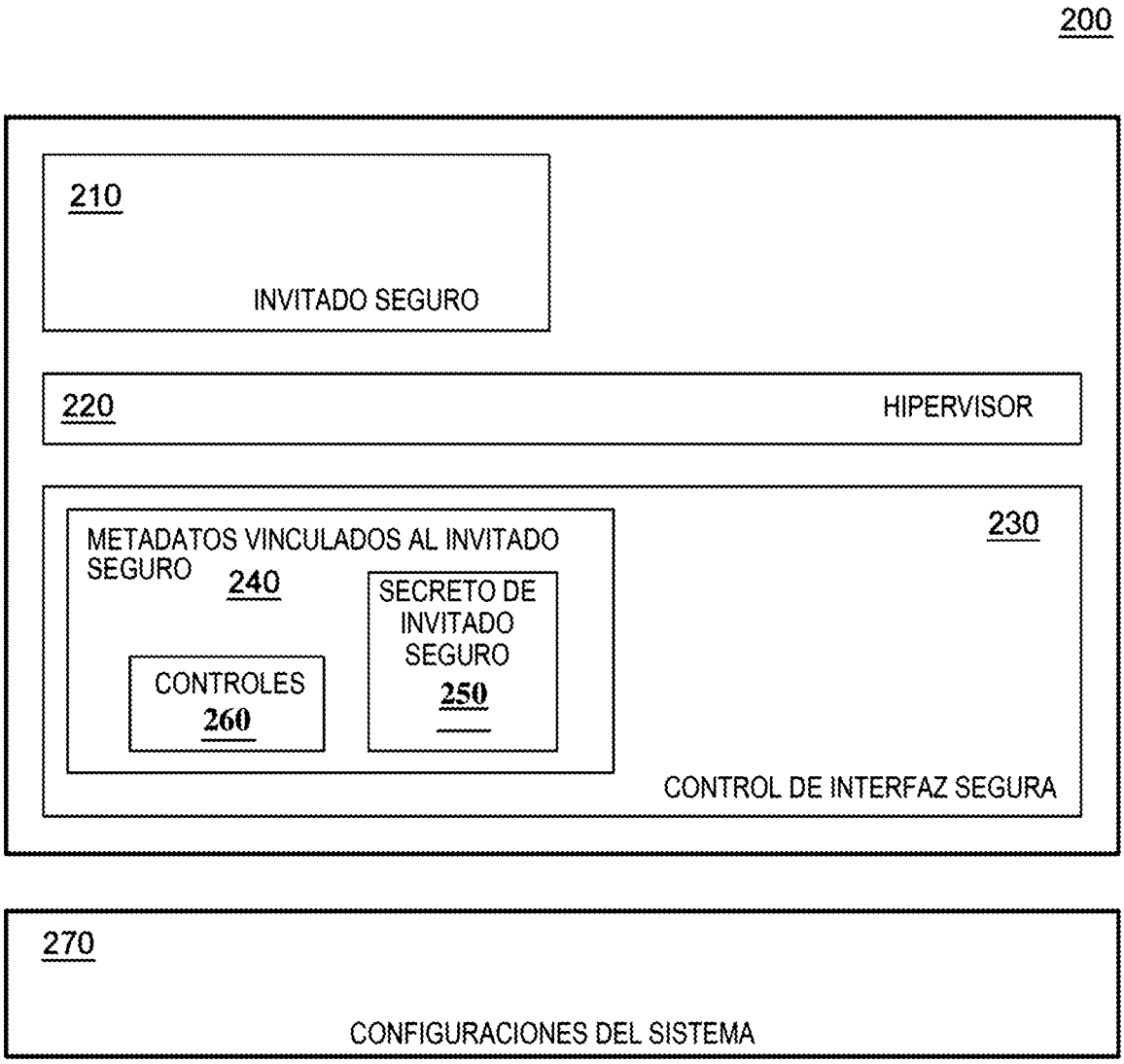
200
210
INVITADO SEGURO
220
HIPERVISOR
230
METADATOS VINCULADOS AL INVITADO SEGURO
240
SECRETO DE INVITADO SEGURO
250
CONTROLES
260
CONTROL DE INTERFAZ SEGURA
270
CONFIGURACIONES DEL SISTEMA

FIG. 2

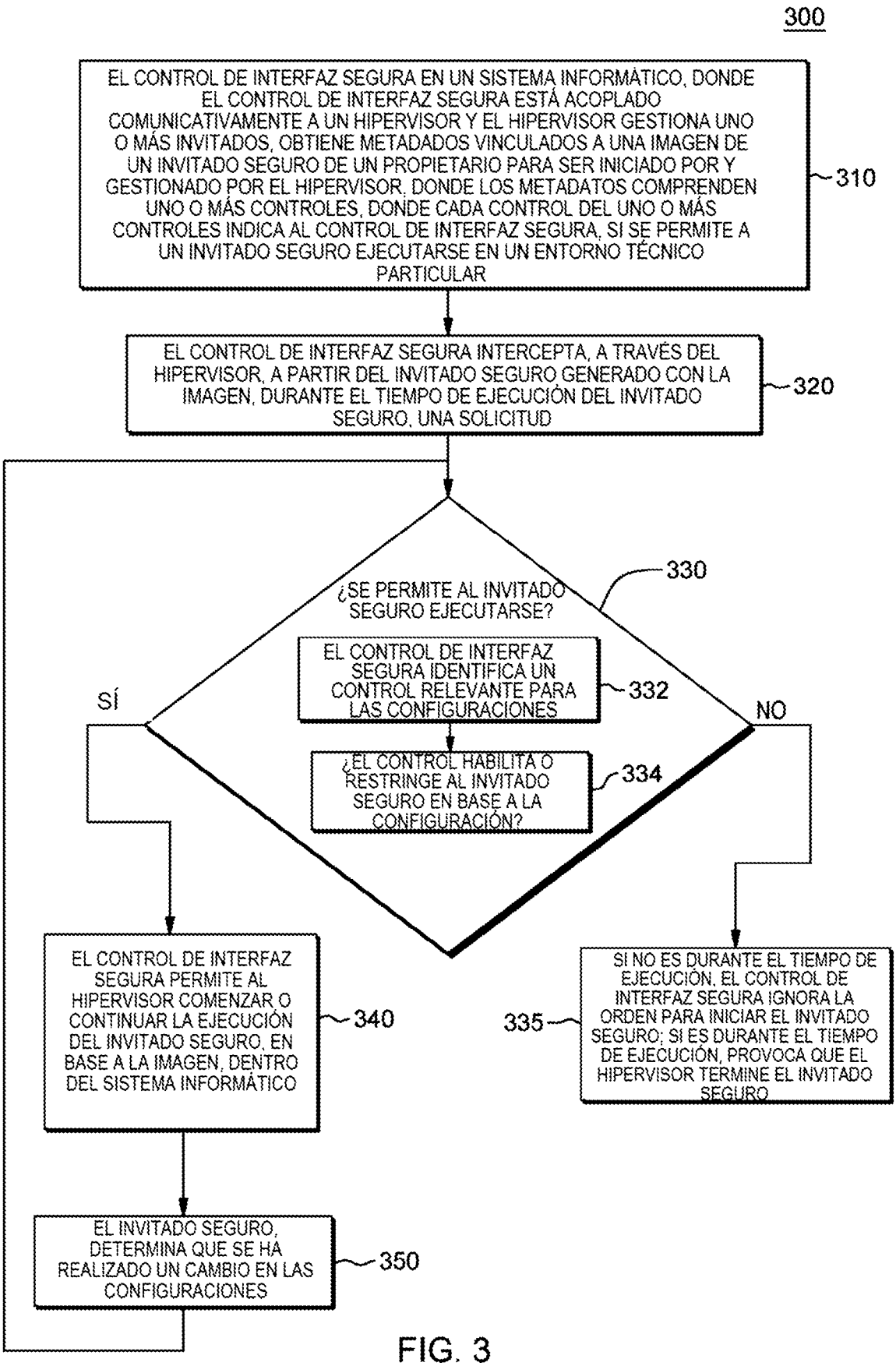
300
EL CONTROL DE INTERFAZ SEGURA EN UN SISTEMA INFORMÁTICO, DONDE EL CONTROL DE INTERFAZ SEGURA ESTÁ ACOPLADO COMUNICATIVAMENTE A UN HIPERVISOR Y EL HIPERVISOR GESTIONA UNO O MÁS INVITADOS, OBTIENE METADADOS VINCULADOS A UNA IMAGEN DE UN INVITADO SEGURO DE UN PROPIETARIO PARA SER INICIADO POR Y GESTIONADO POR EL HIPERVISOR, DONDE LOS METADATOS COMPRENDEN UNO O MÁS CONTROLES, DONDE CADA CONTROL DEL UNO O MÁS CONTROLES INDICA AL CONTROL DE INTERFAZ SEGURA, SI SE PERMITE A UN INVITADO SEGURO EJECUTARSE EN UN ENTORNO TÉCNICO PARTICULAR
310
EL CONTROL DE INTERFAZ SEGURA INTERCEPTA, A TRAVÉS DEL HIPERVISOR, A PARTIR DEL INVITADO SEGURO GENERADO CON LA IMAGEN, DURANTE EL TIEMPO DE EJECUCIÓN DEL INVITADO SEGURO, UNA SOLICITUD
320
¿SE PERMITE AL INVITADO SEGURO EJECUTARSE?
330
EL CONTROL DE INTERFAZ SEGURA IDENTIFICA UN CONTROL RELEVANTE PARA LAS CONFIGURACIONES
332
¿EL CONTROL HABILITA O RESTRINGE AL INVITADO SEGURO EN BASE A LA CONFIGURACIÓN?
334
SÍ
NO
EL CONTROL DE INTERFAZ SEGURA PERMITE AL HIPERVISOR COMENZAR O CONTINUAR LA EJECUCIÓN DEL INVITADO SEGURO, EN BASE A LA IMAGEN, DENTRO DEL SISTEMA INFORMÁTICO
340
SI NO ES DURANTE EL TIEMPO DE EJECUCIÓN, EL CONTROL DE INTERFAZ SEGURA IGNORA LA ORDEN PARA INICIAR EL INVITADO SEGURO; SI ES DURANTE EL TIEMPO DE EJECUCIÓN, PROVOCA QUE EL HIPERVISOR TERMINE EL INVITADO SEGURO
335
EL INVITADO SEGURO, DETERMINA QUE SE HA REALIZADO UN CAMBIO EN LAS CONFIGURACIONES
350

FIG. 3

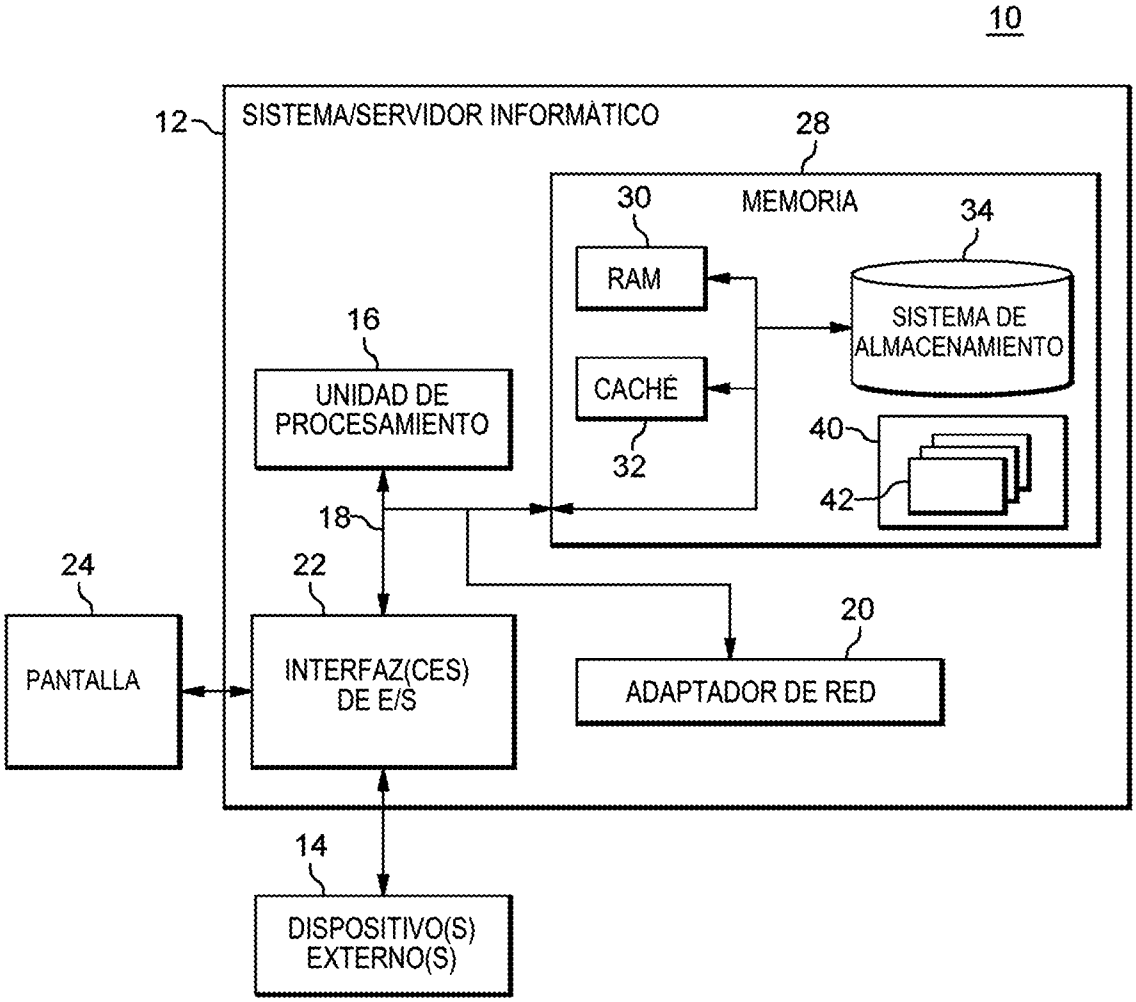
10
12
SISTEMA/SERVIDOR INFORMÁTICO
28
30
MEMORIA
34
RAM
SISTEMA DE ALMACENAMIENTO
16
UNIDAD DE PROCESAMIENTO
CACHÉ
32
40
42
18
24
22
20
PANTALLA
INTERFAZ(CES) DE E/S
ADAPTADOR DE RED
14
DISPOSITIVO(S) EXTERNO(S)

FIG. 4

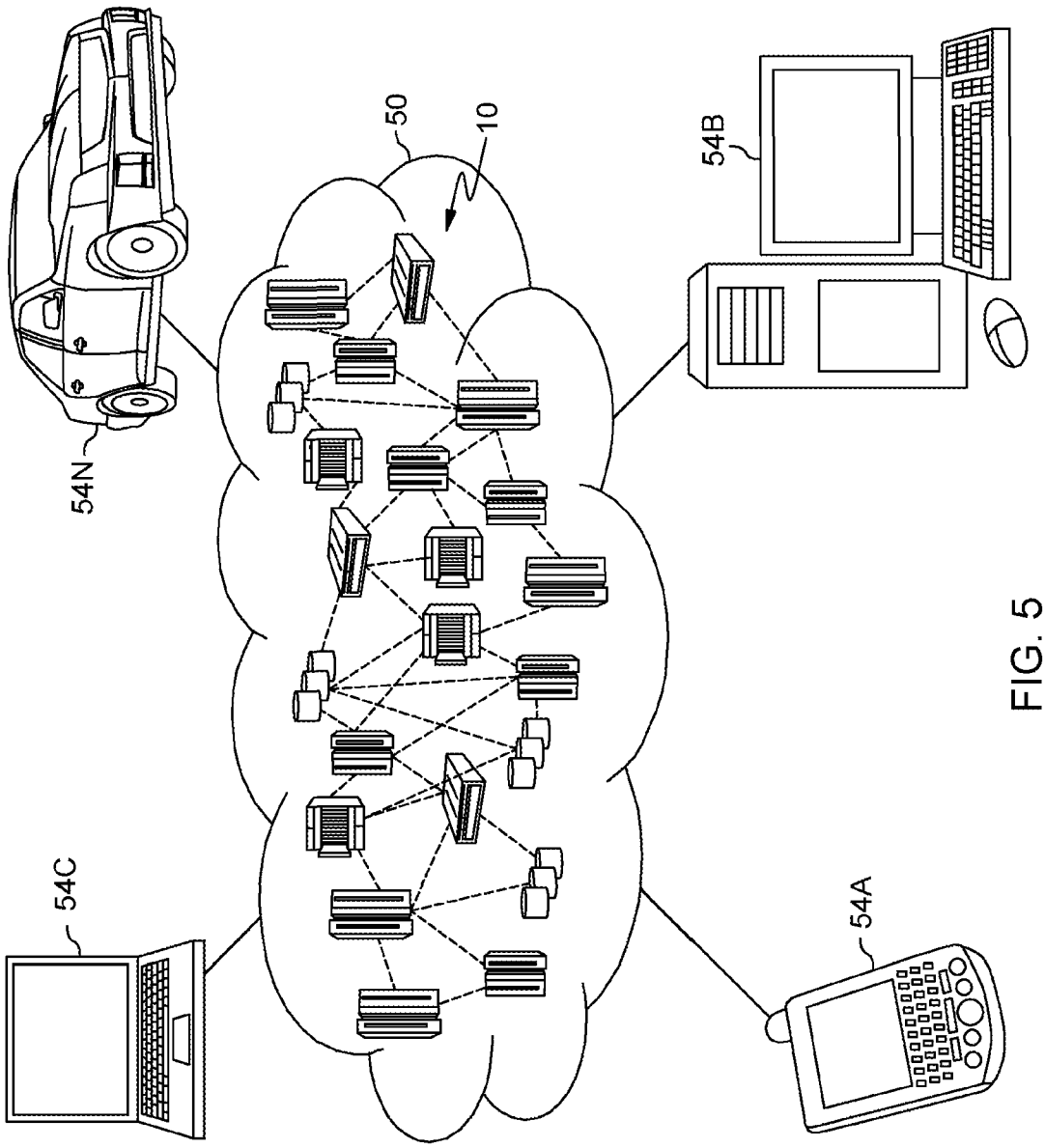

FIG. 5

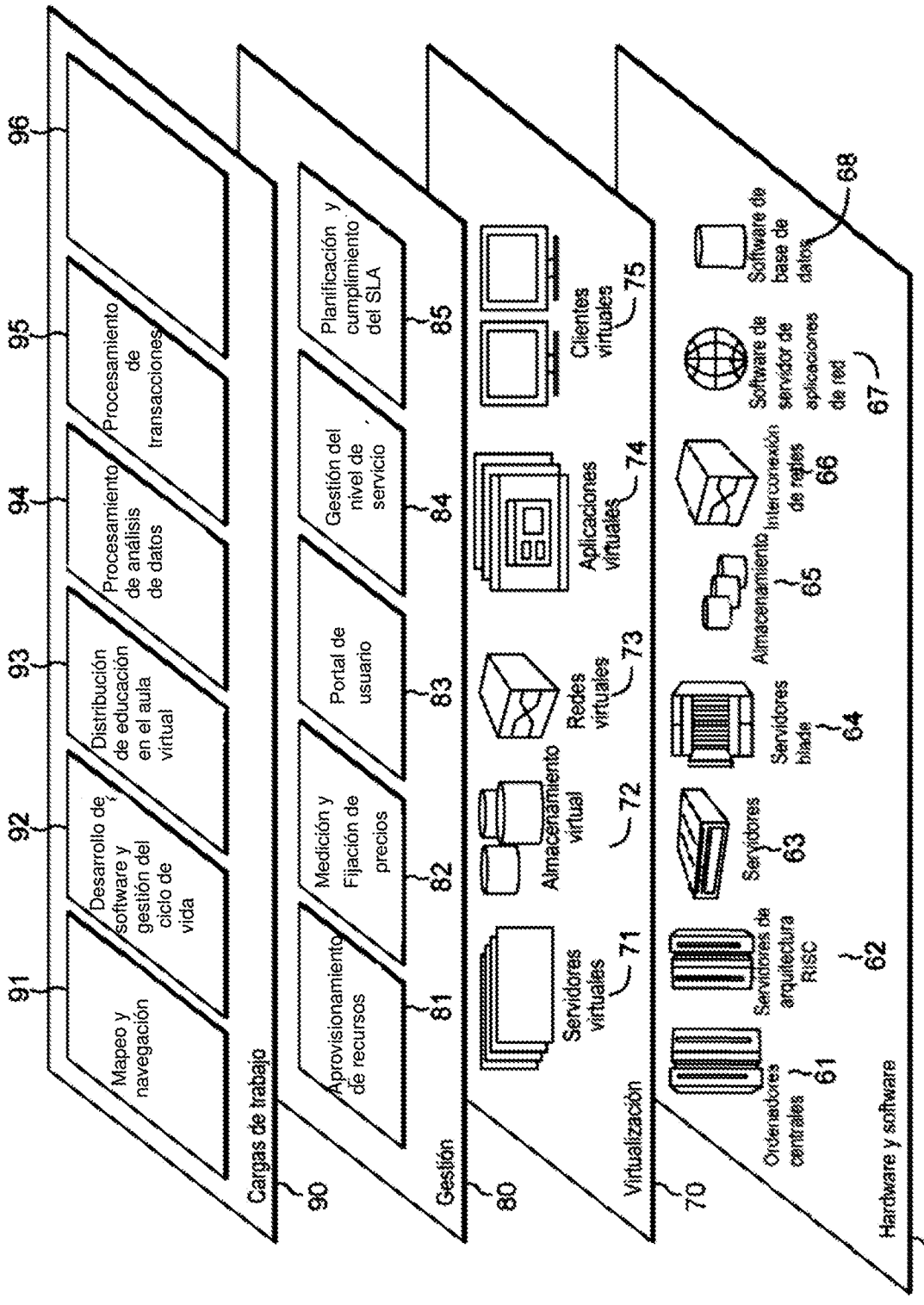
Hardware y software
60
Ordenadores centrales
61
Servidores de arquitectura RISC
62
Servidores
63
Servidores blade
64
Almacenamiento
65
Interconexión de redes
66
Software de servidor de aplicaciones de red
67
Software de base de datos
68
Virtualización
70
Servidores virtuales
71
Almacenamiento virtual
72
Redes virtuales
73
Aplicaciones virtuales
74
Clientes virtuales
75
Gestión
80
Aprovisionamiento de recursos
81
Medición y Fijación de precios
82
Portal de usuario
83
Gestión del nivel de servicio
84
Planificación y cumplimiento del SLA
85
Cargas de trabajo
90
Mapeo y navegación
91
Desarrollo de software y gestión del ciclo de vida
92
Distribución de educación en el aula virtual
93
Procesamiento de análisis de datos
94
Procesamiento de transacciones
95
96

FIG. 6