(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 998 775**

(51) Int. Cl.:

G06F 21/12 (2013.01)
G06F 9/455 (2008.01)
G06F 21/51 (2013.01)
G06F 21/53 (2013.01)
G06F 21/62 (2013.01)
G06F 21/74 (2013.01)
G06F 21/57 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **28.02.2020 PCT/EP2020/055317**

(87) Fecha y número de publicación internacional: **17.09.2020 WO20182498**

(96) Fecha de presentación y número de la solicitud europea: **28.02.2020 E 20708469 (0)**

(97) Fecha y número de publicación de la concesión europea: **27.11.2024 EP 3935532**

(54) Título: **Intercepción de instrucciones de alto nivel de control de interfaz segura para la habilitación de interrupciones**

(30) Prioridad:

08.03.2019 US 201916296452

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
21.02.2025

(73) Titular/es:

INTERNATIONAL BUSINESS MACHINES CORPORATION (100.00%)
New Orchard Road
Armonk, New York 10504, US

(72) Inventor/es:

BORNTRAEGER, CHRISTIAN;
IMBRENDA, CLAUDIO;
BUSABA, FADI;
BRADBURY, JONATHAN y
HELLER, LISA

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Intercepción de instrucciones de alto nivel de control de interfaz segura para la habilitación de interrupciones

Antecedentes

La presente invención se refiere en general a la tecnología informática y, más específicamente, a la intercepción de instrucciones de alto nivel de control de interfaz segura para la habilitación de interrupciones.

La informática en la nube y el almacenamiento en la nube proporcionan a los usuarios capacidades para almacenar y procesar sus datos en centros de datos de terceros. La informática en la nube facilita la capacidad de aprovisionar una máquina virtual (VM) para un cliente de forma rápida y fácil, sin requerir que el cliente compre hardware o proporcione espacio para un servidor físico. El cliente puede expandir o contraer fácilmente la VM según las preferencias o requisitos cambiantes del cliente. Normalmente, un proveedor de informática en la nube aprovisiona la VM, que es físicamente residente en un servidor en el centro de datos del proveedor. Los clientes a menudo se preocupan por la seguridad de los datos en la VM, particularmente porque los proveedores informáticos suelen almacenar los datos de más de un cliente en el mismo servidor. Los clientes pueden desear seguridad entre su propio código/datos y el código/datos del proveedor informático en la nube, así como entre su propio código/datos y el de otras VM que se ejecutan en el sitio del proveedor. Además, el cliente puede desear seguridad de los administradores del proveedor, así como contra posibles violaciones de seguridad de otro código que se ejecuta en la máquina.

Para manejar dichas situaciones sensibles, los proveedores de servicios en la nube pueden implementar controles de seguridad para asegurar el aislamiento adecuado de los datos y la segregación de almacenamiento lógico. El uso extensivo de la virtualización en la implementación de la infraestructura en la nube da como resultado preocupaciones de seguridad únicas para los clientes de servicios en la nube, ya que la virtualización altera la relación entre un sistema operativo (SO) y el hardware subyacente, ya sea hardware informático, de almacenamiento o incluso de red. Esto introduce la virtualización como una capa adicional que, a su vez, debe configurarse, gestionarse y protegerse adecuadamente.

En general, una VM, que se ejecuta como un huésped bajo el control de un hipervisor de anfitrión, depende de ese hipervisor para proporcionar servicios de virtualización de forma transparente para ese huésped. Estos servicios incluyen la gestión de memoria, emulación de instrucciones y procesamiento de interrupciones.

En el caso de la gestión de memoria, la VM puede mover (paginación entrante) sus datos desde un disco para que residan en la memoria y la VM también puede mover sus datos de regreso (paginación saliente) al disco. Mientras la página reside en la memoria, la VM (huésped) usa la traducción dinámica de direcciones (DAT) para hacer corresponder las páginas en la memoria desde una dirección virtual de huésped a una dirección absoluta de huésped. Además, el hipervisor de anfitrión tiene su propia correspondencia DAT (desde la dirección virtual de anfitrión a la dirección absoluta de anfitrión) para las páginas de huésped en la memoria y puede, de forma independiente y transparente para el huésped, paginar las páginas del huésped dentro y fuera de la memoria. Es a través de las tablas DAT de anfitrión que el hipervisor proporciona aislamiento de memoria o uso compartido de memoria de huésped entre dos VM de huésped separadas. El anfitrión también puede acceder a la memoria del huésped para simular las operaciones del huésped, cuando sea necesario, en nombre del huésped.

En el caso de la emulación de instrucciones y el procesamiento de interrupciones, cuando el huésped ejecuta una instrucción en particular, con base en los controles establecidos por el hipervisor, la máquina devuelve el control al hipervisor para que el hipervisor pueda emular esa instrucción en particular en nombre del huésped. El hipervisor puede emular, por ejemplo, las instrucciones de palabra de estado de programa de carga (LPSW) o de control de carga (LCTL). de modo que pueda supervisar la activación de las interrupciones y presentar al huésped las interrupciones pendientes que mantiene el hipervisor con la prioridad adecuada. El documento US 2016/132345 se considera un estado de la técnica relevante relacionado con la protección de una máquina virtual de confianza frente a un hipervisor que no es de confianza mediante el control del acceso del hipervisor a través de un firmware a una memoria protegida de la máquina virtual.

Compendio

De acuerdo con una o más realizaciones, un control de interfaz seguro de un ordenador proporciona un método que proporciona una interpretación parcial de instrucciones para una instrucción que permite una interrupción. El control de interfaz segura obtiene una palabra de estado del programa o un valor de registro de control de un almacenamiento huésped seguro. El control de interfaz segura notifica a una entidad que no es de confianza las actualizaciones de las máscaras de interrupción de huésped. La entidad que no es de confianza se ejecuta en el hardware del ordenador y en comunicación con él a través del control de interfaz segura para respaldar las operaciones de una entidad segura que se ejecuta en la entidad que no es de confianza. El control de interfaz segura recibe, de la entidad que no es de confianza, una solicitud para presentar una interrupción para huéspedes habilitada y de máxima prioridad en respuesta a la notificación de las actualizaciones de la máscara de interrupción de huésped. El control de interfaz segura mueve la información de interrupción a una página de prefijos de huésped e inyecta la interrupción en la entidad segura cuando se determina que una inyección de la interrupción es válida. Los efectos y beneficios técnicos de una o más realizaciones de la presente memoria incluyen la reducción de la complejidad y el riesgo al hacer que este código complejo resida en un solo lugar sin permitir el acceso de la entidad no confiable al estado de huésped seguro o a la memoria.

De acuerdo con una o más realizaciones o la realización del método anterior, el método puede incluir además emitir, por parte de la entidad segura, una palabra de estado de carga del programa o un control de carga que está siendo monitorizado por la entidad no confiable.

De acuerdo con una o más realizaciones o cualquiera de las realizaciones del método anteriores, el método puede incluir además cargar, mediante el control de interfaz segura, la palabra de estado del programa o el registro de control en respuesta a la búsqueda.

De acuerdo con una o más realizaciones o cualquiera de las realizaciones del método anteriores, el método puede incluir además priorizar, por parte de la entidad no confiable, las interrupciones pendientes y habilitadas para determinar la interrupción de huésped habilitada de mayor prioridad.

De acuerdo con una o más realizaciones o cualquiera de las realizaciones del método anteriores, el método puede incluir además almacenar, por parte de la entidad no confiable, la información de interrupción con la máxima prioridad, la interrupción de huésped habilitada en un almacenamiento no seguro.

De acuerdo con una o más realizaciones o cualquiera de las realizaciones del método anteriores, la entidad no confiable puede proporcionar la información de interrupción en una descripción de estado.

De acuerdo con una o más realizaciones o cualquiera de las realizaciones del método anteriores, la entidad no confiable puede emitir una instrucción para proporcionar la información de interrupción al control de interfaz segura y la información de interrupción se pasa como un parámetro para la instrucción.

De acuerdo con una o más realizaciones o cualquiera de las realizaciones del método anteriores, el método puede incluir además emitir, mediante el control de interfaz segura, una excepción a la entidad no confiable cuando se determine que la inyección de la interrupción no es válida. Los efectos y beneficios técnicos de una o más realizaciones de la presente memoria incluyen la reducción de la complejidad y el riesgo al hacer que este código complejo resida en un solo lugar sin permitir el acceso de la entidad no confiable al estado o memoria de huésped seguro.

De acuerdo con una o más realizaciones o cualquiera de las realizaciones del método anteriores, el método puede incluir además la ejecución, por parte de la entidad segura, de un gestor de interrupciones en respuesta a la recepción de la interrupción inyectada.

De acuerdo con una o más realizaciones o cualquiera de las realizaciones del método anteriores, la entidad segura puede incluir un huésped seguro y la entidad no confiable comprende un hipervisor.

De acuerdo con una o más realizaciones, cualquiera de las anteriores realizaciones del método puede implementarse como un sistema o un producto de programa informático.

Se obtienen características y ventajas adicionales a través de las técnicas de la presente descripción. Otras realizaciones y aspectos de la invención se describen en detalle en la presente memoria y se consideran una parte de la invención reivindicada. Para una mejor comprensión de la invención con ventajas y características, véanse la descripción y los dibujos.

Breve descripción de los dibujos

Los características específicas de los derechos exclusivos descritos en la presente memoria se señalan particularmente y se reivindican claramente en las reivindicaciones al final de la memoria descriptiva. Las anteriores y otras características y ventajas de las realizaciones de la invención son evidentes a partir de la siguiente descripción detallada tomada junto con los dibujos adjuntos, en los que:

La FIG. 1 representa una tabla para la seguridad de zona según una o más realizaciones de la presente invención;

la FIG. 2 representa espacios de direcciones virtuales y absolutas para realizar DAT según una o más realizaciones de la presente invención;

la FIG. 3 representa una DAT anidada de múltiples partes para soportar una máquina virtual (VM) que se ejecuta bajo un hipervisor según una o más realizaciones de la presente invención;

la FIG. 4 representa una correspondencia del almacenamiento de huésped seguro según una o más realizaciones de la presente invención;

la FIG. 5 representa un esquema de sistema de una operación de traducción dinámica de direcciones (DAT) según una o más realizaciones de la presente invención;

la FIG. 6 representa un esquema de sistema de una memoria de control de interfaz segura según una o más realizaciones de la presente invención;

la FIG. 7 representa un flujo de proceso de una operación de importación según una o más realizaciones de la presente invención;

la FIG. 8 representa un flujo de proceso de una operación de importación según una o más realizaciones de la presente invención;

la FIG. 9 representa un proceso de una operación de memoria donada según una o más realizaciones de la presente invención;

la FIG. 10 representa un flujo de proceso de una transición de páginas de hipervisor no seguras a páginas seguras de un control de interfaz segura según una o más realizaciones de la presente invención;

la FIG. 11 representa un flujo de proceso de un acceso de almacenamiento seguro realizado por el control de interfaz segura según una o más realizaciones de la presente invención;

la FIG. 12 representa un flujo de proceso de etiquetado de acceso mediante el control de interfaz segura y mediante hardware según una o más realizaciones de la presente invención;

la FIG. 13 representa un flujo de proceso de traducciones para soportar accesos seguros y no seguros mediante el programa y mediante el control de interfaz segura según una o más realizaciones de la presente invención;

la FIG. 14 representa un flujo de proceso de una DAT con protección de almacenamiento seguro mediante el programa y mediante el control de interfaz segura según una o más realizaciones de la presente invención;

la FIG. 15 representa un flujo de proceso para la intercepción de instrucciones de alto nivel de control de interfaz segura para habilitación de interrupciones según una o más realizaciones de la presente invención;

la FIG. 16 representa un flujo de proceso para la intercepción de instrucciones de alto nivel de control de interfaz segura para habilitación de interrupciones según una o más realizaciones de la presente invención;

la FIG. 17 representa un entorno de informática en la nube según una o más realizaciones de la presente invención;

la FIG. 18 representa capas de modelo de abstracción según una o más realizaciones de la presente invención;

la FIG. 19 representa un sistema según una o más realizaciones de la presente invención; y

la FIG. 20 representa un sistema de procesamiento según una o más realizaciones de la presente invención.

Los diagramas representados en la presente memoria son ilustrativos.

Por ejemplo, las acciones se pueden realizar en un orden diferente o se pueden añadir, eliminar o modificar acciones. Asimismo, el término "acoplado" y variaciones del mismo describen la existencia de una vía de comunicación entre dos elementos y no implica una conexión directa entre los elementos sin elementos/conexiones intermedios entre ellos.

Todas estas variaciones se consideran parte de la memoria descriptiva.

Descripción detallada

La invención está definida por la reivindicaciones 1, 11 y 21 independientes.

Una o más realizaciones de la presente invención aprovechan un control de interfaz segura ligero y eficiente entre el software y la máquina para proporcionar seguridad adicional. En este caso, esta interfaz se usa para permitir que un control de interfaz seguro emule la mayoría de las instrucciones de habilitación de interrupciones (por ejemplo, Palabra de Estado de Programa de Carga o Control de Carga) y, al mismo tiempo, permitir que una entidad que no sea de confianza mantenga las interrupciones pendientes en nombre de una entidad segura. Esta estructura de interrupciones pendientes es requerida por la entidad que no es de confianza para gestionar la priorización de las interrupciones con base en la entidad segura que no se envía al hardware. Los efectos y beneficios técnicos de una o más realizaciones de la presente memoria incluyen la reducción de la complejidad y el riesgo al hacer que este código complejo resida en un solo lugar sin permitir el acceso de la entidad no confiable al estado o memoria de huésped seguro.

Una máquina virtual (VM), que se ejecuta como huésped bajo el control de una entidad que no es de confianza (por ejemplo, un hipervisor de anfitrión), depende de ese hipervisor para proporcionar de forma transparente servicios de virtualización para ese huésped. 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. Como se mencionó anteriormente, estos servicios pueden incluir, pero no se limitan a, la gestión de memoria, emulación de instrucciones y procesamiento de interrupciones. Por ejemplo, para la inyección de interrupciones y excepciones, el hipervisor normalmente lee y/o escribe en un área de prefijo (núcleo bajo) del huésped. El término “máquina virtual” o “VM”, como se usa en la presente memoria, se refiere a una representación lógica de una máquina

física (dispositivo informático, procesador, etc.) y su entorno de procesamiento (sistema operativo (OS), recursos de software, etc.). La VM se mantiene como software que se ejecuta en una máquina de anfitrión subyacente (procesador físico o conjunto de procesadores). Desde la perspectiva de un usuario o un recurso de software, la VM parece ser su propia máquina física independiente. Los términos "hipervisor" y "monitor de VM (VMM)", como se usan en la presente memoria, se refieren a un entorno de procesamiento o servicio de plataforma que gestiona y permite que múltiples VM se ejecuten usando múltiples (y a veces diferentes) OS en una misma máquina de anfitrión. Debe apreciarse que la implementación de una VM incluye un proceso de instalación de la VM y un proceso de activación (o inicio) de la VM. En otro ejemplo, desplegar una VM incluye un proceso de activación (o inicio) de la VM (por ejemplo, en caso de que la VM se haya instalado previamente o ya exista).

En las soluciones técnicas disponibles actualmente, el hipervisor (por ejemplo, z/VM® de IBM® o una máquina Virtual Basada en Núcleo (KVM) de software de código abierto) distribuye una nueva CPU virtual (vCPU) de VM en una unidad de procesamiento física o servidor anfitrión, emitiendo una instrucción de Inicio de Ejecución Interpretativa (SIE) que hace que se invoque el milicódigo de Entrada SIE. El milicódigo es un firmware confiable que funciona como una extensión del hardware del procesador. El operando de la instrucción SIE es un bloque de control, denominado descripción de estado, que contiene el estado de huésped. Durante la Entrada SIE, este estado huésped (que incluye los registros de control y de uso general, la dirección de instrucciones del huésped y la palabra de estado de programa de huésped (PSW)) se cargan mediante milicódigo en el hardware. Esto permite que la vCPU invitada se ejecute en el procesador físico. Mientras la vCPU se ejecuta en el hardware, el estado de huésped se mantiene en el hardware. En algún momento, el hardware/millicódigo debe devolver el control al hipervisor. Esto se conoce con frecuencia como Salida SIE. Esto puede ser necesario, por ejemplo, si esta vCPU ejecuta una instrucción que requiere la emulación por parte del hipervisor o si el intervalo de tiempo de la vCPU (es decir, el tiempo asignado para que esta vCPU se ejecute en el procesador físico) caduca. Los hipervisores existentes se basan en el uso de dicha interfaz mediante la instrucción SIE para enviar vCPU.

Para facilitar y dar soporte a huéspedes seguros (por ejemplo, entidad segura), existe un reto técnico en el que se requiere seguridad adicional entre el hipervisor y los huéspedes seguros sin depender del hipervisor, de manera que el hipervisor no pueda acceder a los datos de la VM y, por lo tanto, no pueda proporcionar servicios de la manera descrita anteriormente.

La ejecución segura descrita en la presente memoria proporciona un mecanismo de hardware para garantizar el aislamiento entre el almacenamiento seguro y el almacenamiento no seguro, así como entre el almacenamiento seguro que pertenece a diferentes usuarios seguros. Para huéspedes seguros, se proporciona seguridad adicional entre el hipervisor no seguro "no confiable" y los huéspedes seguros. Para hacer esto, muchas de las funciones que el hipervisor normalmente hace en nombre de los huéspedes deben incorporarse en la máquina. En la presente memoria se describe un nuevo control de interfaz segura, también denominado "UV" en la presente memoria, para proporcionar una interfaz segura entre el hipervisor y los huéspedes seguros. Los términos control de interfaz segura y UV se usan indistintamente en la presente memoria. El control de interfaz segura funciona en colaboración con el hardware para proporcionar esta seguridad adicional.

El control de interfaz segura, en un ejemplo, se implementa en hardware y/o firmware interno, seguro y confiable. Para un huésped o entidad seguros, el control de interfaz segura proporciona la inicialización y el mantenimiento del entorno seguro, así como la coordinación del envío de estas entidades seguras en el hardware. Mientras el huésped seguro usa activamente los datos y es residente en el almacenamiento del anfitrión, se mantiene "a salvo" en almacenamiento seguro. Ese huésped seguro único puede acceder al almacenamiento seguro de huésped; el hardware hace cumplir esto estrictamente. Es decir, el hardware impide a cualquier entidad no segura (incluyendo el hipervisor u otros huéspedes no seguros) o huésped seguro diferente el acceso a esos datos. En este ejemplo, el control de interfaz segura se ejecuta como una parte confiable de los niveles más bajos de firmware. El nivel más bajo, o milicódigo, es realmente una extensión del hardware y se usa para implementar las instrucciones y funciones complejas definidas, por ejemplo, en zArchitecture® de IBM. Milicódigo tiene acceso a todas las partes del almacenamiento, lo que, en el contexto de la ejecución segura, incluye su propio almacenamiento UV seguro, almacenamiento de hipervisor no seguro, almacenamiento de huésped seguro y almacenamiento compartido. Esto le permite proporcionar cualquier función que necesite el huésped seguro o el hipervisor en apoyo de ese huésped. El control de interfaz segura también tiene acceso directo al hardware, lo que permite que el hardware proporcione controles de seguridad de manera eficiente bajo el control de las condiciones establecidas por el control de interfaz segura.

De acuerdo con una o más realizaciones de la presente invención, se proporciona un bit de almacenamiento seguro en el hardware para marcar una página segura. Cuando se establece este bit, el hardware evita que cualquier huésped o hipervisor no seguro acceda a esta página. Además, cada página segura o compartida se registra en una tabla de seguridad de zona y se etiqueta con una identificación (ID) de dominio de huésped seguro. Cuando la página no es segura, se marca como tal en la tabla de seguridad de zona. Esta tabla de seguridad de zona se mantiene mediante el control de interfaz segura por partición o zona. Hay una entrada por página absoluta del anfitrión que es usada por el hardware en cualquier traducción DAT realizada por una entidad segura para verificar que acceden a la página solo el huésped seguro o la entidad que la posee.

Según una o más realizaciones de la presente invención, el software usa una instrucción de Llamada UV (UVC) para solicitar el control de interfaz segura para realizar una acción específica. Por ejemplo, la instrucción UVC puede ser

usada por el hipervisor para inicializar el control de interfaz segura, crear el dominio de huésped seguro (por ejemplo, configuración de huésped segura) y crear las CPU virtuales dentro de esa configuración segura. También se puede usar para importar (descifrar y asignar al dominio de huésped seguro) y exportar (cifrar y permitir el acceso del anfitrión) una página de huésped segura como parte de las operaciones de paginación entrante o paginación saliente del hipervisor. Además, el huésped seguro tiene la capacidad de definir el almacenamiento compartido con el hipervisor, hacer que el almacenamiento seguro sea compartido y hacer que el almacenamiento compartido sea seguro.

Para proporcionar seguridad, cuando el hipervisor página de forma transparente la entrada y salida de los datos de huéspedes seguros, el control de la interfaz segura, que trabaja con el hardware, proporciona y garantiza el descifrado y el cifrado de los datos. Para lograr esto, se requiere que el hipervisor emita nuevas UVC al enviar y recibir los datos seguros de huésped. El hardware, basado en controles configurados por el control de interfaz segura durante estas nuevas UVC, garantizará que estas UVC sean emitidas por el hipervisor.

En este nuevo entorno seguro, cada vez que el hipervisor realiza paginación saliente a una página segura, se requiere emitir una nueva UVC de conversión de almacenamiento seguro (exportación). El UV, o control de interfaz segura, en respuesta a esta exportación de UVC, 1) indicará que la página está "bloqueada" por el UV, 2) cifrará la página, 3) establecerá la autoridad a un dominio de huésped seguro particular y 4) restablecerá el bloqueo UV. Una vez que se completa la UVC de exportación, el hipervisor ahora puede eliminar la página de huésped cifrada.

Además, cuando el hipervisor realiza paginación entrante a una página segura, se requiere emitir una nueva conversión a UVC de almacenamiento seguro (importación). El UV, o control de interfaz segura, en respuesta a esta Importación de UVC, 1) marcará la página como segura en el hardware, 2) indicará que la página está "bloqueada" por el UV, 3) descifrará la página, 4) establecerá la autoridad a un dominio de huésped seguro particular y 5) restablecerá el bloqueo UV. Cuando un acceso sea hecho por una entidad segura, el hardware realiza comprobaciones de autorización en esa página durante la traducción. Estas comprobaciones incluyen 1) una comprobación para verificar que la página realmente pertenece al dominio de huésped seguro que está tratando de acceder a ella y 2) una comprobación para asegurarse de que el hipervisor no ha cambiado la correspondencia de anfitrión de esta página mientras esta página ha sido residente en la memoria de huésped. Una vez que una página se marca como segura, el hardware evita el acceso a cualquier página segura por parte del hipervisor o de una VM huésped no segura. Los pasos de traducción adicionales impiden el acceso por otra VM segura y evitan el volver a realizar la correspondencia por parte del hipervisor.

Hay casos en los que el hipervisor emula las instrucciones en nombre de un huésped no seguro. Sin embargo, para un huésped seguro, el control de la interfaz segura debe intervenir y proporcionar cualquier función que pueda permitir que un hipervisor “no confiable” comprometa el estado de huésped seguro. Esta intervención puede ser necesaria por varias razones. Por ejemplo, la emulación de esta instrucción puede requerir el acceso a una memoria segura para los huéspedes o para proteger las instalaciones de los huéspedes. En algunos casos, el UV emulará completamente la instrucción. En otros casos, el control de interfaz segura completará la emulación de la instrucción, pero notificará al hipervisor alguna actualización del estado de huésped. En otros casos, se usa una interfaz para pasar información limitada entre el control de la interfaz segura y el hipervisor sin comprometer el estado del huésped. Este enfoque aprovecha la interfaz ligera entre el control de la interfaz segura y el hipervisor, lo que nos permite minimizar, en este caso, la duplicación de la estructura de interrupción pendiente en el control de la interfaz segura. Esta estructura de interrupción pendiente es volviendo ahora a la FIG. 1, una tabla 100 para seguridad de zona generalmente mostrada de acuerdo con una o más realizaciones de la presente invención. La tabla 100 de seguridad de zona mostrada en la FIG. 1 se mantiene mediante el control de interfaz segura y es usada por el control de interfaz segura y el hardware para garantizar el acceso seguro a cualquier página a la que acceda una entidad segura. La tabla 100 de seguridad de zona está indexada por la dirección 110 absoluta de anfitrión. Es decir, hay una entrada para cada página de almacenamiento absoluto de anfitrión. Cada entrada incluye información que se usa para verificar que la entrada pertenece a la entidad segura que realiza el acceso.

Además, como se muestra en la FIG. 1, la tabla 100 de seguridad de zona incluye un ID 120 de dominio seguro (identifica el dominio seguro asociado a esta página); un bit 130 UV (indica que esta página se donó al control de interfaz segura y es propiedad del control de interfaz segura); un bit 140 de deshabilitación de comparación de direcciones (DA) (usado para deshabilitar la comparación de pares de direcciones de anfitrión en determinadas circunstancias, tal como cuando una página de control de interfaz segura que se define como absoluta de anfitrión no tiene una dirección virtual de anfitrión asociada); un bit 150 (SH) compartido (indica que la página es compartida con el hipervisor no seguro) y una dirección 160 virtual de anfitrión (indica la dirección virtual de anfitrión registrada para esta dirección absoluta de anfitrión, que se denomina como par de direcciones de anfitrión). Se observa que un par de direcciones de anfitrión indica una dirección virtual de anfitrión registrada, absoluta y asociada de anfitrión. El par de direcciones de anfitrión representa la correspondencia de esta página, una vez importada por el hipervisor, y la comparación garantiza que el anfitrión no vuelva a hacer corresponder esa página mientras la esté usando el huésped.

La traducción dinámica de direcciones (DAT) se usa para hacer corresponder el almacenamiento virtual al almacenamiento real. Cuando una VM invitada se ejecuta como un huésped paginable bajo el control de un hipervisor, el huésped usa DAT para gestionar las páginas que residen en su memoria. Además, el anfitrión, de forma independiente, usa DAT para gestionar aquellas páginas de huésped (junto con sus propias páginas) cuando las páginas residen en su memoria. El hipervisor usa DAT para proporcionar aislamiento y/o compartir almacenamiento

entre diferentes VM, así como para evitar que el huésped acceda al almacenamiento de hipervisor. El hipervisor tiene acceso a todo el almacenamiento de los huéspedes cuando los huéspedes se ejecutan en modo no seguro.

La DAT permite el aislamiento de una aplicación de otra al mismo tiempo que les permite compartir recursos comunes. Además, permite la implementación de VM, que pueden ser usadas en el diseño y prueba de nuevas versiones de sistemas operativos junto con el procesamiento simultáneo de programas de aplicación. Una dirección virtual identifica una ubicación en el almacenamiento virtual. Un espacio de direcciones es una secuencia consecutiva de direcciones virtuales, junto con los parámetros de transformación específicos (que incluyen las tablas DAT) que permiten traducir cada dirección virtual a una dirección absoluta asociada que identifica esa dirección con una ubicación de bytes en el almacenamiento.

La DAT usa una búsqueda en múltiples tablas para traducir la dirección virtual a la dirección absoluta asociada. Esta estructura de tabla es normalmente definida y mantenida por un gestor de almacenamiento. Este gestor de almacenamiento comparte de forma transparente el almacenamiento absoluto entre varios programas mediante la paginación saliente de una página, por ejemplo, para incluir otra página. Cuando la página esté paginada hacia fuera, el gestor de almacenamiento establecerá un bit no válido en la tabla de página asociada, por ejemplo. Cuando un programa intenta acceder a una página que estaba paginada hacia fuera, el hardware presentará una interrupción del programa, a menudo denominada fallo de página, al gestor de almacenamiento. En respuesta, el gestor de almacenamiento paginará hacia dentro en la página solicitada y restablecerá el bit no válido. Todo esto se hace de forma transparente para el programa y permite al gestor de almacenamiento virtualizar el almacenamiento y compartirlo entre varios usuarios diferentes.

Cuando una dirección virtual es usada por una CPU para acceder al almacenamiento principal, primero se convierte, mediante DAT, en una dirección real y, a continuación, mediante prefijado, en una dirección absoluta. La designación (origen y longitud) de la tabla de nivel más alto para un espacio de direcciones específico se denomina elemento de control de espacio de direcciones (ASCE) y define el espacio de direcciones asociado.

Volviendo ahora a la FIG. 2, se muestran de manera general los espacios 202 y 204 de direcciones virtuales de ejemplo y un espacio 206 de direcciones absoluto para realizar la DAT de acuerdo con una o más realizaciones de la presente invención. En el ejemplo mostrado en la FIG. 2, hay dos espacios de direcciones virtuales: el espacio 202 de direcciones virtual (definido por el elemento 208 de control de espacio de direcciones (ASCE) A) y el espacio 204 de direcciones virtual (definido por ASCE 210 B). El gestor de almacenamiento hace corresponder las páginas A1.V 212a1, A2.V 212a2 y A3.V 212a3 virtuales en una búsqueda de múltiples tablas (segmento 230 y tablas 232a, 232b de páginas), usando el ASCE 208 A, a las páginas A1.A 220a1, A2.A 220a2 y A3.A 220a3 absolutas. De manera similar, las páginas B1.V 214b1 y B2.V 214b2 virtuales se hacen corresponder en una búsqueda de dos tablas 234 y 236, usando el ASCE 210 B, a las páginas absolutas B1.A 222b1 y B2.A 222b2, respectivamente.

Volviendo ahora a la FIG. 3, se muestra generalmente un ejemplo de una traducción DAT anidada de múltiples partes usada para soportar una VM que se ejecuta bajo un hipervisor de acuerdo con una o más realizaciones de la presente invención. En el ejemplo mostrado en la FIG. 3, el espacio 302 de direcciones virtuales A del huésped A (definido por el ASCE (GASCE) 304 A de huésped) y el espacio 306 B de direcciones virtuales del huésped B (definido por el GASCEB 308) residen ambos en un espacio 325 de direcciones virtuales de anfitrión (hipervisor) compartido. Como se muestra, las páginas A1.GV 310a1, A2.GV 310a2 y A3.GV 310a3 virtuales, que pertenecen al huésped A, se hacen corresponder, por el gestor de almacenamiento del huésped A, usando GASCEA 304 a las páginas A1.HV 340a1, A2.HV 340a2 y A3.HV 340a3 absolutas de huésped, respectivamente; las páginas B1.GV 320b1 y B2.GV 320b2 virtuales, que pertenecen al huésped B, se hacen corresponder, independientemente, por el gestor de almacenamiento del huésped B, usando GASCEB 308 a las páginas B1.HV 360b1 y B2.HV 360b2 absolutas de huésped, respectivamente. En este ejemplo, estas páginas absolutas de huésped se hacen corresponder directamente al espacio 325 de direcciones virtuales de anfitrión compartido y, posteriormente, pasan a través de una traducción DAT de anfitrión adicional a un espacio 330 de direcciones absolutas de anfitrión. Como se muestra, las direcciones A1.HV 340a1, A3.HV 340a3 y B1.HV 360b1 virtuales de anfitrión se hacen corresponder, por el gestor de almacenamiento de anfitrión usando el anfitrión 350 ASCE (HASCE) para A1.HA 370a1, A3.HA 370a3 y B1.HA 370b1. La dirección A2.HV 340a2 virtual de anfitrión, que pertenece al huésped A, y la B2.HV 360b2, que pertenece al huésped B, ambas se hacen corresponder a la misma página AB2.HA 380 absoluta de anfitrión. Esto permite que los datos se compartan entre estos dos huéspedes. Durante la traducción DAT de huésped, cada una de las direcciones de la tabla de huésped se trata como una absoluta de huésped y se somete a una traducción DAT de anfitrión anidada adicional.

Las realizaciones de la presente invención descritas en la presente memoria proporcionan una protección de almacenamiento UV y huésped seguro. El acceso al almacenamiento seguro por parte de los huéspedes no seguros y al hipervisor está prohibido. El hipervisor proporciona que, para una página de huésped segura residente dada, ocurra lo siguiente. La dirección absoluta de anfitrión asociada solo es accesible a través de una única correspondencia DAT de hipervisor (anfitrión). Es decir, hay una sola dirección virtual de anfitrión que se hace corresponder a cualquier dirección absoluta de anfitrión asignada a un huésped seguro. La correspondencia DAT de hipervisor (virtual de anfitrión a absoluta de anfitrión) asociada a una página de huésped segura dada no cambia mientras se página hacia dentro. La página absoluta de anfitrión asociada a una página de huésped segura se hace corresponder para un único huésped seguro.

También está prohibido compartir el almacenamiento entre huéspedes seguros según una o más realizaciones de la presente invención. El almacenamiento se comparte entre un único huésped seguro y el hipervisor bajo el control del huésped seguro. El almacenamiento UV es un almacenamiento seguro y es accesible mediante la interfaz de control segura, pero no por los huéspedes/anfitriones. El hipervisor asigna el almacenamiento a la interfaz de control segura. Según una o más realizaciones de la presente invención, cualquier intento de violación de estas reglas está prohibido por el hardware y la interfaz de control segura.

Volviendo ahora a la FIG. 4, se muestra generalmente un ejemplo de correspondencia de almacenamiento de huésped seguro según una o más realizaciones de la presente invención. La FIG. 4 se parece a la FIG. 3, excepto que el ejemplo de la FIG. 4 no permite compartir el almacenamiento entre el huésped A seguro y el huésped B seguro. En el ejemplo no seguro de la FIG. 3, tanto la dirección A2.HV 340a2 virtual de anfitrión, que pertenece al huésped A, como la B2.HV 360b2, que pertenece al huésped B, se hacen corresponder a la misma página AB2.HA 380 absoluta de anfitrión. En el ejemplo de almacenamiento de huésped seguro de la FIG. 4, la dirección A2.HV 340a2 virtual de anfitrión, que pertenece al huésped A, se hace corresponder a la dirección A2.HA 490a absoluta de anfitrión, mientras que la B2.HV 360b2, que pertenece al huésped B, se hace corresponder a su propia B2.HA 490b. En este ejemplo, no se comparte nada entre huéspedes seguros.

Mientras la página de huésped segura reside en el disco, está cifrada. Cuando el hipervisor realiza paginación entrante de una página de huésped segura, emite una llamada UV (UVC), lo que hace que la interfaz de control segura marque la página como segura (a menos que se comparta), la descifre (a menos que se comparta) y la registre (en la tabla de seguridad de zona) como perteneciente al huésped seguro apropiado (el huésped A, por ejemplo). Además, registra la dirección virtual de anfitrión asociada (A3.HV 340a3, por ejemplo) en esa página absoluta de anfitrión (denominada par anfitrión-dirección). Si el hipervisor no emite la UVC correcta, recibe una excepción cuando intenta acceder a la página de huésped segura. Cuando el hipervisor realiza paginación saliente de una página de huésped, se emite una UVC similar que cifra la página de huésped (a menos que se comparta) antes de marcar la página de huésped como no segura y registrarla en la tabla de seguridad de zona como no segura.

En un ejemplo que tiene cinco páginas K, P, L, M y N absolutas de anfitrión dadas, la interfaz de control segura marca cada una de las páginas absolutas de anfitrión como segura cuando el hipervisor realiza paginación entrante en ellas. Esto impide que los huéspedes no seguros y el hipervisor accedan a ellas. Las páginas K, P y M absolutas de anfitrión se registran como pertenecientes al huésped A cuando el hipervisor realiza paginación entrante en ellas; las páginas L y N absolutas de anfitrión se registran al huésped B cuando el hipervisor realiza paginación entrante en ellas. Las páginas compartidas, páginas compartidas entre un único huésped seguro y el hipervisor, no se cifran ni descifran durante la paginación. No se marcan como seguras (permiten el acceso por el hipervisor), pero se registran con un único dominio de huésped seguro en la tabla de seguridad de zona.

De acuerdo con una o más realizaciones de la presente invención, cuando un huésped no seguro o el hipervisor intentan acceder a una página que es propiedad de un huésped seguro, el hipervisor recibe una excepción de acceso al almacenamiento seguro (PIC3D). No se requiere ningún paso de traducción adicional para determinar esto.

De acuerdo con una o más realizaciones, cuando una entidad segura intenta acceder a una página, el hardware realiza una comprobación de traducción adicional que verifica que el almacenamiento pertenece realmente a ese huésped seguro particular. De lo contrario, se presenta una excepción de acceso no seguro (PIC3E) al hipervisor. Además, si la dirección virtual de anfitrión que se está traduciendo no coincide con la dirección virtual de anfitrión del par anfitrión-dirección registrado en la tabla de seguridad de zona, se reconoce una excepción de violación de almacenamiento seguro ('3F'x). Para habilitar el uso compartido con el hipervisor, un huésped seguro puede acceder al almacenamiento que no esté marcado como seguro siempre y cuando las comprobaciones de traducción permitan el acceso.

Volviendo ahora a la FIG. 5, se muestra generalmente un esquema 500 de sistema de una operación DAT según una o más realizaciones de la presente invención. El esquema 500 de sistema incluye un espacio 510 de direcciones virtuales primario de anfitrión y un espacio 520 de direcciones virtuales de inicio de anfitrión, desde los que se traducen las páginas (por ejemplo, véase la traducción 525 DAT de anfitrión; se observa que las líneas punteadas representan la correspondencia a través de la traducción 525 DAT) a un espacio 530 de direcciones absolutas del hipervisor (anfitrión). Por ejemplo, la FIG. 5 ilustra la compartición de almacenamiento absoluto de anfitrión por dos espacios de direcciones virtuales de anfitrión diferentes y también la compartición de una de esas direcciones virtuales de anfitrión no solo entre dos huéspedes sino, además, con el propio anfitrión. En este sentido, el espacio 510 de direcciones virtuales primario de anfitrión y el espacio 520 de direcciones virtuales de inicio de anfitrión son ejemplos de dos espacios de direcciones virtuales de anfitrión, cada uno de los cuales se direcciona por un ASCE separado, el ASCE 591 primario de anfitrión (HPASCE) y el ASCE 592 de inicio de anfitrión (HHASCE), respectivamente. Se observa que todo el almacenamiento de control de interfaz segura (tanto virtual como real) lo dona el hipervisor y lo marca como seguro. Una vez donado, el almacenamiento de control de interfaz segura solo puede ser accedido por el control de interfaz segura mientras exista una entidad segura asociada.

Como se ilustra, el espacio 510 de direcciones virtuales primario de anfitrión incluye una página A1.HV absoluta de huésped A, una página A2.HV absoluta de huésped A, una página B1.HV absoluta de huésped B y una página H3.HV virtual de anfitrión. El espacio 520 de direcciones virtuales de inicio de anfitrión incluye una página U1.HV virtual de control de interfaz segura, una página H1.HV virtual de anfitrión y una página H2.HV virtual de anfitrión.

De acuerdo con una o más realizaciones de la presente invención, se registra todo el almacenamiento de huésped seguro (por ejemplo, huésped A seguro e huésped B seguro) en la tabla de seguridad de zona descrita en la presente memoria como perteneciente a una configuración de huésped segura, y la dirección virtual de anfitrión asociada (por ejemplo, A1.HV, A2.HV, B1.HV) también se registra como parte de un par anfitrión-dirección En una o más realizaciones, todo el almacenamiento de huésped seguro se hace corresponder en el espacio virtual primario de anfitrión. Además, se registra todo el almacenamiento de control de interfaz segura, también en la tabla de seguridad de zona, como perteneciente al control de interfaz segura y se puede diferenciar adicionalmente en la tabla de seguridad de zona con base en el dominio de huésped seguro asociado. De acuerdo con una o más realizaciones de la presente invención, el almacenamiento virtual UV se hace corresponder en el espacio virtual de inicio de anfitrión y la dirección virtual de anfitrión asociada se registra como parte del par anfitrión-dirección. De acuerdo con una o más realizaciones, el almacenamiento real UV no tiene una correspondencia virtual de anfitrión asociada, y el bit DA en la tabla de seguridad de zona (que indica que la comparación de direcciones virtuales está deshabilitada) se establece para indicar esto. El almacenamiento de anfitrión se marca como no seguro y también se registra en la tabla de seguridad de zona como no seguro.

Por lo tanto, en el caso donde "absoluta de huésped = virtual de anfitrión", las tablas DAT primarias del hipervisor (anfitrión) (definidas por el HPASCE 591) traducen las páginas del espacio 510 de direcciones virtuales primario de anfitrión de la siguiente manera: la Página A1.HV Absoluta de Huésped A se hace corresponder a una Absoluta A1.HA de Anfitrión que pertenece a un Huésped A Seguro; la Página A2.HV Absoluta de Huésped A se hace corresponder a una Absoluta A2.HA de Anfitrión que pertenece al Huésped A Seguro; la Página B1.HV Absoluta de Huésped B se hace corresponder a una Absoluta B1.HA de Anfitrión que pertenece al Huésped B Seguro; y la Página H3.HV Virtual de Anfitrión se hace corresponder a una Página H3.HA Absoluta de Anfitrión de Anfitrión No Seguro (y no hay un par anfitrión-dirección ya que no es seguro). Además, las tablas DAT de inicio del hipervisor (anfitrión) (definidas por el HHASCE 592) traducen las páginas del espacio 520 de direcciones virtuales de inicio de anfitrión de la siguiente manera: la Página U1.HV Virtual de Control de Interfaz Segura se hace corresponder a una Página U1.HA Absoluta de Anfitrión definida como Virtual UV Segura; la Página H1.HV Virtual de Anfitrión se hace corresponder a una Página H1.HA Absoluta de Anfitrión definida como No Segura; y la Página H2.HV Virtual de Anfitrión se hace corresponder a una Página H2.HA Absoluta de Anfitrión definida como No Segura. No hay un par anfitrión-dirección asociado a H1.HA ni a H2.HA, ya que no son seguras.

En funcionamiento, si un huésped seguro intenta acceder a una página segura asignada al control de interfaz segura, el hardware presenta al hipervisor una excepción de violación de almacenamiento seguro ('3F'X). Si un huésped no seguro o el hipervisor intentan acceder a una página segura (incluidos las asignadas al control de interfaz segura), el hardware presenta al hipervisor una excepción de acceso al almacenamiento seguro ('3D'X). De manera alternativa, se puede presentar una condición de error para los intentos de acceso realizados al espacio de control de interfaz segura. Si el hardware detecta una disparidad en la asignación segura (por ejemplo, el almacenamiento se registra en la tabla de seguridad de zona como perteneciente a un huésped seguro en lugar de al control de interfaz segura, o hay una disparidad en el par anfitrión-dirección que se usa con el par registrado) en un acceso de control de interfaz segura, se presenta una comprobación.

En otras palabras, el espacio 510 de direcciones virtuales primario de anfitrión incluye las páginas A1.HV y A2.HV virtuales de anfitrión (que pertenecen al huésped A seguro) y B1.HV (que pertenecen al huésped B seguro), que se hacen corresponder a la absoluta A1.HA, A2.HA y B1.HA de anfitrión, respectivamente. Además, el espacio 510 de direcciones virtuales primario de anfitrión incluye la página H3.HV de anfitrión (hipervisor), que se hace corresponder a la absoluta H3.HA de anfitrión. El espacio 520 virtual de inicio de anfitrión incluye dos páginas H1.HV y H2.HV virtuales de anfitrión, que se hacen corresponder en las páginas H1.HA y H2.HA absolutas de anfitrión. Tanto el espacio 510 de direcciones virtuales primario de anfitrión como el espacio 520 de direcciones virtuales de inicio de anfitrión se hacen corresponder en el absoluto 530 de anfitrión único. Las páginas de almacenamiento que pertenecen al huésped A seguro y al huésped B seguro se marcan como seguras y se registran en la tabla 100 de seguridad de zona mostrada en la FIG. 1 con sus dominios seguros y direcciones virtuales de anfitrión asociadas. El almacenamiento de anfitrión, por otro lado, se marca como no seguro. Cuando el hipervisor define los huéspedes seguros, debe donar almacenamiento de anfitrión al control de interfaz segura para usar en los bloques de control seguros necesarios para soportar estos huéspedes seguros. Este almacenamiento se puede definir ya sea en el espacio absoluto de anfitrión o en el virtual de anfitrión y, en un ejemplo, específicamente, en el espacio virtual de inicio de anfitrión. Volviendo a la FIG. 5, unas páginas U1.HA y U2.HA absolutas de anfitrión Absoluta UV Segura es un almacenamiento de control de interfaz segura que se define como almacenamiento absoluto de anfitrión. Como resultado, estas páginas se marcan como seguras y se registran en la tabla 100 de seguridad de zona mostrada en la FIG. 1 como perteneciente al control de interfaz segura y con un dominio seguro asociado. Ya que las páginas se definen como direcciones absolutas de anfitrión, no hay una dirección virtual de anfitrión asociada, por lo que el bit DA se establece en la tabla 100 de seguridad de zona.

Después de la traducción, se puede encontrar un ejemplo del Espacio 530 de Direcciones Absolutas de Hipervisor (Anfitrión) en la FIG. 6. En la FIG. 6, se representa un esquema 600 de sistema con respecto a una memoria de control de interfaz segura según una o más realizaciones de la presente invención. El esquema 600 de sistema ilustra un Espacio 630 de Direcciones Absolutas de Hipervisor (Anfitrión) que incluye un Página A2.HA Absoluta de Anfitrión Huésped A Seguro (para A2.HV); una Página B1.HA Absoluta de Anfitrión Huésped B Seguro (para B1.HV); Página H1.HA Absoluta de Anfitrión No Seguro (Anfitrión); una Página H2.HA Absoluta de Anfitrión No Seguro (Anfitrión); una

Página U3.HA Absoluta de Anfitrión Real UV Seguro (sin correspondencia HV); una Página U1.HA Absoluta de Anfitrión Virtual UV Seguro (para U1.HV); y una Página A1.HA Absoluta de Anfitrión Huésped A Seguro (para A1.HV).

Volviendo ahora a la FIG. 7, se muestra generalmente un flujo 700 de proceso para una operación de importación según una o más realizaciones de la presente invención. Cuando un huésped seguro accede a una página a la que el hipervisor realizó paginación saliente, se produce una secuencia de eventos como la que se muestra en el flujo 700 de proceso para volver a introducir esa página de forma segura. El flujo 700 de proceso comienza en el bloque 705, donde el huésped seguro accede a la página virtual de huésped. Dado que la página, por ejemplo, no es válida, el hardware presenta al hipervisor un fallo de página de anfitrión, indicado por el código 11 de interrupción de programa (PIC11) (véase el bloque 715). El hipervisor, a su vez, identifica una página absoluta de anfitrión no segura disponible para esta página de huésped (véase el bloque 720) y se realiza paginación entrante de la página de huésped cifrada a la página absoluta de anfitrión identificada (véase el bloque 725).

En el bloque 730, la página absoluta de anfitrión se hace corresponder entonces en las tablas DAT de anfitrión apropiadas (con base en la dirección virtual de anfitrión). En el bloque 735, el anfitrión de hipervisor se reenvía después al huésped seguro. En el bloque 740, el huésped seguro vuelve a acceder a la página segura de huésped. El fallo de página ya no existe, pero dado que se trata de un acceso de huésped seguro y la página no está marcada como segura en la tabla 100 de seguridad de zona de la FIG. 100, el hardware presenta una excepción de almacenamiento no seguro (PIC3E) al hipervisor, en el bloque 745. Esta PIC3E impide el acceso del huésped a esta página segura hasta que se haya emitido la importación necesaria. A continuación, el flujo 700 de proceso pasa a "A", que está conectado a la FIG. 8.

Volviendo ahora a la FIG. 8, se muestra generalmente un flujo 800 de proceso para realizar una operación de importación según una o más realizaciones de la presente invención. Un hipervisor de buen comportamiento (por ejemplo, que funcione de la manera esperada sin errores), en respuesta a la PIC3E, emitirá una Importación de UVC (véase el bloque 805). Se observa que, en este punto, una página que se va a importar se marca como no segura y solo pueden acceder a ella el hipervisor, otras entidades no seguras y el control de interfaz segura. Los huéspedes seguros no pueden acceder a la misma.

Como parte de la Importación de UVC, el firmware confiable que actúa como control de interfaz segura comprueba si esta página ya está bloqueada por el control de interfaz segura (véase el bloque 810 de decisión). Si lo está, el flujo 800 de proceso pasa al bloque 820. En el bloque 820, se devuelve un código de retorno "ocupado" al hipervisor que, en respuesta, retrasará (véase el bloque 825) y volverá a emitir la Importación de UVC (el flujo 800 de proceso regresa al bloque 805). Si la página aún no está bloqueada, el flujo 800 de proceso pasa al bloque 822 de decisión.

En el bloque 822 de decisión, el control de interfaz segura comprueba si la página es una página que se comparte con el hipervisor no seguro. Si se comparte (el flujo 800 de proceso pasa al bloque 824 de decisión), el control de interfaz segura registra la dirección absoluta de anfitrión en la tabla de seguridad de zona con el dominio de huésped seguro asociado, la dirección virtual de anfitrión y como compartida. Esta página permanece marcada como no segura. Esto completa la Importación de UVC y la página ahora está disponible para que acceda el huésped. El procesamiento continúa con el hipervisor que reasigna al huésped (bloque 830) y el huésped seguro que accede a la página con éxito (bloque 835).

Si la página virtual de anfitrión que se va a importar no se comparte con el hipervisor (el flujo 800 de proceso pasa al bloque 840), el control de interfaz segura marcará la página como segura, para que el hipervisor ya no pueda acceder a la página. En el bloque 845, el control de interfaz segura bloquea la página, para que ninguna otra UVC pueda modificar el estado de la página. Una vez establecido el bloqueo (en el bloque 850), el control de interfaz segura verificará que el contenido de la página de huésped no cambió mientras se cifraba. Si cambió, se devuelve entonces un código de retorno de error al hipervisor; de lo contrario, el control de interfaz segura descifrará la página segura.

En el bloque 855, el control de interfaz segura desbloquea la página, lo que permite el acceso de otras UVC, registra la página en la tabla de seguridad de zona como segura y asociada con el dominio de huésped y la dirección virtual de anfitrión apropiados para completar el par HV->HA de anfitrión-dirección. Esto permite el acceso del huésped y completa la UVC.

Volviendo ahora a la FIG. 9, un flujo 900 de proceso con respecto a una operación de memoria donada se muestra generalmente según una o más realizaciones de la presente invención. El flujo 900 de proceso comienza en el bloque 905, donde un hipervisor emite una consulta UVC al control de interfaz segura. En el bloque 910, el control de interfaz segura devuelve datos (por ejemplo, Consulta UVC). Estos datos pueden incluir una cantidad de almacenamiento base absoluto de anfitrión específico de la zona requerida; una cantidad de almacenamiento base absoluto de anfitrión específico del dominio de huésped seguro requerida; una cantidad de almacenamiento variable virtual de anfitrión específico del dominio de huésped seguro requerida por MB; y/o una cantidad de almacenamiento base absoluto de anfitrión específico de CPU de huésped seguro requerida.

En el bloque 915, el hipervisor reserva el almacenamiento base específico de zona absoluto de anfitrión (por ejemplo, con base en un tamaño devuelto por la consulta UVC). En el bloque 920, el hipervisor emite una inicialización al control de interfaz segura. En este sentido, el hipervisor puede emitir una inicialización de UVC que proporciona

almacenamiento donado para los bloques de control UV que se necesitan para coordinar entre las configuraciones de huésped seguro para toda la zona. La inicialización de UVC especifica un origen de almacenamiento base específico de zona.

En el bloque 925, el control de interfaz segura implementa la inicialización (por ejemplo, inicialización de UVC) registrando el almacenamiento donado a UV y marcándolo como seguro. Para la inicialización de UVC, el control de interfaz segura puede marcar el almacenamiento donado como seguro; asignar parte de ese almacenamiento donado a la tabla de seguridad de zona; y registrar el almacenamiento donado en la tabla de seguridad de zona para uso UV con un dominio seguro único, pero sin dominio de huésped seguro asociado y sin un par anfitrión-dirección virtual asociado.

En el bloque 930, el hipervisor reserva almacenamiento (por ejemplo, almacenamiento base y variable específico de dominio de huésped seguro). Por ejemplo, el hipervisor reserva almacenamiento base y variable (por ejemplo, con base en un tamaño de almacenamiento de dominio de huésped seguro) específico del dominio de huésped seguro (por ejemplo, un tamaño devuelto por la consulta UVC). En el bloque 935, el hipervisor emite una configuración de creación al control de interfaz segura. En este sentido, el hipervisor puede emitir una creación de configuración de UVC de huésped seguro que especifica el origen de almacenamiento base y variable específico de dominio de huésped seguro. Además, la creación de configuración de UVC de huésped seguro proporciona almacenamiento donado para los bloques de control UV que se necesitan para soportar esta configuración de huésped segura.

En el bloque 940, el control de interfaz segura implementa la creación de configuración (por ejemplo, creación de configuración de UVC de huésped seguro). Para la creación de configuración de UVC de huésped seguro, el control de interfaz segura puede marcar el almacenamiento donado como seguro; registrar el almacenamiento donado en la tabla de seguridad de zona para uso UV; y registrar el almacenamiento donado con el dominio de huésped seguro asociado. El almacenamiento base donado (absoluto de anfitrión) se registra como que no tiene ningún par de direcciones virtuales de anfitrión asociadas. El almacenamiento variable donado (virtual de anfitrión) se registra con el par de direcciones virtuales de anfitrión asociadas.

En el bloque 945, el hipervisor reserva el almacenamiento base específico de CPU de huésped seguro (por ejemplo, un tamaño devuelto por la consulta UV). En el bloque 950, el hipervisor especifica un origen de almacenamiento. Por ejemplo, el hipervisor emite a la UV la creación de CPU de huésped seguro que especifica un origen de almacenamiento base específico de CPU de huésped seguro. En el bloque 955, el control de interfaz segura implementa la creación de CPU (por ejemplo, UVC de creación de CPU de huésped seguro). Para la UVC de creación de CPU de huésped seguro, el control de interfaz segura puede marcar el almacenamiento donado como seguro y registrar el almacenamiento donado en la tabla de seguridad de zona para uso UV, pero sin dominio de huésped seguro asociado y sin un par de anfitrión-dirección virtual asociado.

Volviendo ahora a la FIG. 10, se muestra generalmente un flujo 1000 de proceso con respecto a una transición de páginas de hipervisor no seguras a páginas seguras de un control de interfaz segura según una o más realizaciones de la presente invención. En el flujo 1000 de proceso, se muestran tres páginas de hipervisor (por ejemplo, una Página A de hipervisor no segura, una Página B de hipervisor no segura y una página C de hipervisor no segura).

Las Páginas A, B y C de hipervisor (no seguras) pueden ser accedidas por una entidad no segura (que incluye el hipervisor). Además, las páginas A, B y C de hipervisor (no seguras) se marcan como no seguras (NS), y se registran en una tabla de seguridad de zona (por ejemplo, la tabla 100 de seguridad de zona mostrada en la FIG. 1) como no seguras y no compartidas. En la flecha 1005, se emite una inicialización de UVC, que realiza la transición de la página A de huésped a la página 1010 de almacenamiento real de control de interfaz segura asociada a una zona completa (UV2). El almacenamiento 1010 real de control de interfaz segura puede marcarse como seguro y registrarse en una tabla de seguridad de zona (por ejemplo, la tabla 100 de seguridad de zona mostrada en la FIG. 1) como UV sin dominio de huésped seguro y sin correspondencia de hipervisor a anfitrión absoluto(HV->HA). En su lugar, se registra con un dominio seguro UV2 único y el bit DA se establece en 1. Se observa que el control de interfaz segura puede acceder al almacenamiento 1010 real de control de interfaz segura como real.

Desde la página B de hipervisor (no segura), en la flecha 1025, se emite la creación de configuración SG o creación de UVC SG CPU, que realiza la transición de esta página a un almacenamiento 1030 real de control de interfaz segura asociado a un dominio de huésped seguro (UVS). El almacenamiento real de control de interfaz segura 1030 puede marcarse como seguro y registrarse en una tabla de seguridad de zona (por ejemplo, la tabla 100 de seguridad de zona mostrada en la FIG. 1) como UV con un dominio de huésped seguro asociado y sin correspondencia de hipervisor a anfitrión absoluto (HV->HA) (es decir, bit DA=1). Se observa que el control de interfaz segura puede acceder al almacenamiento real de control de interfaz segura 1010 como real en nombre de un dominio de huésped seguro.

Desde la Página C de hipervisor (no segura), en la flecha 1045, se emite la creación de configuración SG UVC, que realiza la transición de esta página a un almacenamiento 1050 virtual de control de interfaz segura asociado a un dominio de huésped seguro (UVV). El almacenamiento 1050 virtual de control de interfaz segura puede marcarse como seguro, junto con el registro en una tabla de seguridad de zona (por ejemplo, la tabla 100 de seguridad de zona mostrada en la FIG. 1) como UV con un dominio de huésped seguro y correspondencia de hipervisor a anfitrión

absoluto (HV->HA). Se observa que se puede acceder al almacenamiento 1050 virtual de control de interfaz segura como virtual UV en nombre de un dominio de huésped seguro.

Volviendo ahora a la FIG. 11, se representa un flujo 1100 de proceso con respecto a un acceso de almacenamiento seguro realizado por el programa o el control de interfaz segura según una o más realizaciones. Esto representa la situación en la que el control de interfaz segura va a acceder al almacenamiento de huésped o al almacenamiento de control de interfaz segura y debe etiquetar ese acceso correctamente para permitir que el hardware verifique la seguridad de ese acceso. 1100 describe este etiquetado de los accesos de almacenamiento mediante el control de interfaz segura. El flujo 1100 de proceso comienza en el bloque 1110, donde el control de interfaz segura determina si está accediendo a un almacenamiento de control de interfaz segura.

Si este no es un acceso al almacenamiento de control de interfaz segura, entonces el flujo 1100 de proceso pasa al bloque 1112 de decisión (como se muestra mediante la flecha NO). En el bloque 1112 de decisión, el control de interfaz segura determina si está accediendo a un almacenamiento de huésped seguro. Si este no es un acceso al almacenamiento de huésped seguro, entonces el flujo 1100 de proceso pasa a “B” (que está conectado al flujo 1200 de proceso de la FIG. 12), que usará el ajuste predeterminado para accesos no seguros. Si este es un acceso al almacenamiento de huésped seguro, entonces el flujo 1100 de proceso pasa al bloque 1113 de decisión, donde el control de interfaz segura determina si se está usando un dominio de huésped seguro predeterminado. Si es así, entonces el flujo 1100 de proceso procede a pasar a “B” (que está conectado al flujo 1200 de proceso de la FIG. 12), que usará el ajuste predeterminado para accesos de huésped seguro. Si no, entonces el flujo 1100 de proceso pasa al bloque 1114. En el bloque 1114, se carga un dominio de huésped seguro apropiado en el registro de dominio seguro SG (y pasa a "B", que está conectado al flujo 1200 de proceso de la FIG. 12).

Si este es un acceso al almacenamiento de control de interfaz segura, entonces el flujo 1100 de proceso pasa al bloque 1120 (como muestra la flecha SÍ). En el bloque 1120, el acceso se etiqueta como UV seguro (por ejemplo, usa el registro de dominio seguro UV).

El flujo 1100 de proceso pasa entonces al bloque 1130 de decisión, donde el control de interfaz segura determina si este es un acceso al espacio UVV (por ejemplo, Tabla Variable de Configuración SG). Si es un acceso al espacio UVV, entonces el flujo 1100 de proceso pasa al bloque 1134 (como se muestra mediante la flecha SÍ). En el bloque 1134, el acceso se etiqueta como virtual. En el bloque 1136, un dominio de huésped seguro aplicable se carga en el registro de dominio seguro UV. En el bloque 1138, la traducción DAT y el acceso a almacenamiento están listos para comenzar. Volviendo al bloque 1130 de decisión, si este no es un acceso al espacio UVV, entonces el flujo 1100 de proceso pasa al bloque 1140 (como se muestra mediante la flecha NO). En el bloque 1140, el acceso se etiqueta como real.

En el bloque 1150 de decisión, el control de interfaz segura determina si este es un acceso al espacio UVS (por ejemplo, Configuración SG o tabla de CPU). Si este es un acceso al espacio UVS, entonces el flujo 1100 de proceso pasa al bloque 1136 (como se muestra mediante la flecha SÍ). Si este no es un acceso al espacio UVS, entonces el flujo 1100 de proceso pasa al bloque 1170 (como se muestra mediante la flecha NO). Este acceso sería entonces un acceso al espacio UV2 (por ejemplo, Tabla de Seguridad de Zona). En el bloque 1170, se carga un dominio seguro UV2 único en el registro de dominio seguro UV.

La FIG. 12 representa un flujo 1200 de proceso según una o más realizaciones de la presente invención. Cuando se asigna un huésped, el firmware de Entrada SIE puede indicar al hardware que se está ejecutando un huésped (por ejemplo, modo huésped activo) y puede indicar si el huésped es seguro. Si el huésped es seguro, el dominio de huésped seguro asociado se puede cargar en el hardware (por ejemplo, en el registro de dominio seguro SG). Cuando un programa accede al almacenamiento, el hardware puede etiquetar el acceso con base en el estado actual del programa en el momento del acceso. La FIG. 12 ilustra un ejemplo de este proceso en el flujo 1200 de proceso. En el bloque 1205, el hardware puede determinar si la máquina se está ejecutando actualmente en modo huésped y, de no ser así, puede etiquetar el acceso como un acceso de anfitrión en el bloque 1210 y como un acceso no seguro en el bloque 1215. Si la máquina se está ejecutando en modo huésped en el bloque 1205, el acceso puede etiquetarse como un acceso de huésped en el bloque 1220 y determinar además si el huésped actual es un huésped seguro en el bloque 1225. Si el huésped no es seguro, el acceso puede etiquetarse como no seguro en el bloque 1215. Si el huésped es seguro, el hardware puede etiquetar al huésped como seguro en el bloque 1230, lo que puede asociar al huésped seguro con el registro de dominio seguro SG que se cargó cuando se asignó el huésped seguro. Tanto para los huéspedes seguros como para los no seguros, se puede comprobar el estado DAT en el bloque 1235. El acceso puede etiquetarse como real en el bloque 1240, si la DAT está desactivada. El acceso se puede etiquetar como virtual en el bloque 1245, si la DAT está activada. Una vez que el acceso se etiqueta como real en el bloque 1240 con la DAT desactivada o como virtual en el bloque 1245 con la DAT activada, el hardware está listo para comenzar la traducción y acceder al almacenamiento en el bloque 1250, como se describe adicionalmente en la FIG. 13.

La FIG. 13 representa un ejemplo de traducción realizada por el hardware para soportar tanto accesos seguros como no seguros en el flujo 1300 de proceso según una o más realizaciones de la presente invención. En el bloque 1305, el hardware puede determinar si el acceso se etiqueta como una traducción de huésped y, de ser así, y el acceso es virtual en el bloque 1310, entonces la DAT de huésped se puede realizar en el bloque 1315. Durante la traducción DAT de huésped, se pueden realizar búsquedas intermedias anidadas para las tablas DAT de huésped. Las

búsquedas de tabla se pueden etiquetar como reales de huésped y como seguras si la traducción original se etiquetó como segura. Las búsquedas de tabla también pueden seguir el proceso de traducción del flujo de proceso 1300. Después de realizar la DAT de huésped para un acceso etiquetado como virtual de huésped en el bloque 1315 y para cualquier acceso etiquetado como real de huésped en el bloque 1310 (virtual=No), se pueden aplicar en el bloque 1320 prefijación de huésped y desplazamiento de memoria de huésped. Al completar el proceso de traducción de huésped, la dirección resultante se puede etiquetar como virtual de anfitrión y como segura si la traducción original de huésped se etiquetó como segura en el bloque 1325. El proceso 1300 puede continuar como para cualquier acceso etiquetado como virtual de anfitrión. Si el acceso original es un acceso de anfitrión en el bloque 1305 (huésped=No) y virtual en el bloque 1330, entonces la DAT de anfitrión puede realizarse en el bloque 1335. Las búsquedas de la tabla de anfitrión se pueden marcar como no seguras en el bloque 1335. Después de realizar la DAT de anfitrión en el bloque 1335, o si el acceso de anfitrión original se etiquetó como real (virtual=No) en el bloque 1330, entonces se puede aplicar el prefijado de anfitrión en el bloque 1340. La dirección resultante puede ser una dirección absoluta de anfitrión en el bloque 1345.

La FIG. 14 representa un ejemplo de traducción DAT con protección de almacenamiento seguro que puede realizarse mediante el hardware en el flujo 1400 de proceso según una o más realizaciones de la presente invención. Continuando desde el bloque 1345 de la FIG. 13, si se identifica un acceso UV seguro en el bloque 1405, entonces el hardware puede verificar si el almacenamiento está registrado como almacenamiento UV seguro en el bloque 1410 y, si no, se presenta un error en el bloque 1415. Se puede realizar un acceso UV seguro mediante la interfaz de control segura cuando se accede al almacenamiento UV. Si el almacenamiento está registrado como almacenamiento UV seguro en el bloque 1410, las comprobaciones de protección pueden continuar ya que se puede realizar para cualquier acceso seguro, excepto que el registro de dominio seguro UV (configurado por la interfaz de control segura antes de realizar un acceso UV seguro) pueda usarse como el dominio seguro especificado para la comprobación de dominio en el bloque 1420, donde continúa el procesamiento. Además, cualquier violación que se detecte (punto de entrada D) para un acceso UV en el bloque 1425 puede presentarse como un error en el bloque 1430 en lugar de una excepción al hipervisor en el bloque 1435, como se hace para una violación de huésped seguro en el bloque 1425 (UV seguro=No).

Para el acceso que no está etiquetado como acceso UV seguro en el bloque 1405, el hardware determina si el acceso es un acceso de huésped seguro en el bloque 1440 y, si no, y si la página está marcada como segura en el bloque 1445, se puede presentar una excepción al hipervisor en el bloque 1435. De lo contrario, si el acceso no es un acceso de huésped seguro en el bloque 1440 y la página no está marcada como segura en el bloque 1445, entonces la traducción se realiza con éxito en el bloque 1450.

Si el acceso es un acceso de huésped seguro en el bloque 1440 o un acceso UV seguro al almacenamiento registrado como almacenamiento UV seguro en el bloque 1410, el hardware puede hacer una comprobación para asegurar que el almacenamiento está registrado en la entidad segura asociada al acceso en el bloque 1420. Si este es un acceso UV seguro, el dominio seguro especificado se puede obtener del registro de dominio seguro UV (cargado mediante la interfaz de control segura con base en el almacenamiento UV seguro al que se accede) y, para un acceso de huésped seguro, el dominio seguro especificado se obtiene del registro de dominio seguro SG (cargado cuando se asigna a la entidad segura). Si el almacenamiento al que se está accediendo no está registrado en el dominio seguro especificado en el bloque 1420, entonces para los accesos UV seguros en el bloque 1425 se produce un error en el bloque 1430 y para los accesos de huésped seguro en el bloque 1425 (UV seguro=No) se presenta una excepción al hipervisor en el bloque 1435.

Para accesos seguros al almacenamiento en el bloque 1440 y el bloque 1410 que están registrados en el dominio seguro especificado en el bloque 1420, si la comprobación de direcciones virtuales está deshabilitada, es decir, el bit DA=1 en el bloque 1455 y el acceso es real en el bloque 1460, entonces la traducción se completa en el bloque 1450. Sin embargo, si el bit DA=1 en el bloque 1455 pero el acceso es virtual en el bloque 1460 (real=No), entonces para los accesos UV seguros en el bloque 1425 se produce un error en el bloque 1430 y para los accesos de huésped seguro en el bloque 1425 (UV seguro=No) se presenta una excepción al hipervisor en el bloque 1435. Si el bit DA=0 en el bloque 1455 y el acceso es un acceso virtual en el bloque 1475, entonces el hardware puede determinar si la correspondencia de anfitrión virtual a anfitrión absoluto del acceso coincide con la registrada para esta dirección absoluta de anfitrión en el bloque 1470. Si es así, entonces la traducción se completa con éxito en el bloque 1450. Si la correspondencia no coincide en el bloque 1470, entonces para los accesos UV seguros en el bloque 1425 se produce un error en el bloque 1430 y para los accesos de huésped seguro en el bloque 1425 (UV seguro=No) se presenta una excepción al hipervisor en el bloque 1435. Si el bit DA=0 y el acceso es un acceso real en el bloque 1475 (virtual=No), entonces para los accesos UV seguros en el bloque 1425 se produce un error en el bloque 1430 y para los accesos de huésped seguro en el bloque 1425 (UV seguro=No) se presenta una excepción al hipervisor en el bloque 1435; alternativamente, la traducción puede completarse con éxito en el bloque 1450. Cualquier acceso por el subsistema I/O en el bloque 1480 puede comprobar si la página está marcada como segura en el bloque 1445 y si la página es segura, se puede presentar una excepción al hipervisor en el bloque 1435; si la página no está marcada como segura, la traducción se realiza con éxito en el bloque 1450.

Se pueden gestionar colectivamente varias comprobaciones de registro y la correspondencia del almacenamiento a través de la interfaz 1485 de tabla de seguridad de zona. Por ejemplo, los bloques 1410, 1420, 1455, 1470 y 1475 pueden interactuar con una tabla de seguridad de zona que está asociada a una misma zona para gestionar varios accesos.

Como se discute en la presente memoria, una o más realizaciones de la presente memoria aprovechan una interfaz de ultravisor segura ligera y eficiente entre el software y la máquina para proporcionar seguridad adicional. En este caso, esta interfaz se usa para permitir que el ultravisor emule la mayoría de las instrucciones de activación de interrupciones (por ejemplo, cargar la palabra de estado del programa o el control de carga) y, al mismo tiempo, permitir que el hipervisor mantenga las interrupciones pendientes en nombre del huésped. El hipervisor requiere esta estructura de interrupciones pendientes para gestionar la priorización de las interrupciones cuando el huésped seguro no se envía al hardware. Los efectos y beneficios técnicos de una o más realizaciones de la presente memoria incluyen la reducción de la complejidad y el riesgo al hacer que este código complejo resida en un solo lugar sin permitir el acceso por el hipervisor al estado o memoria de huésped seguro.

En vista de lo anterior, las operaciones para la intercepción de instrucciones de alto nivel de control de interfaz segura para la habilitación de interrupciones se analizan con respecto a las FIG. 15-16. La FIG. 15 representa un flujo de proceso de etiquetado de hardware de almacenamiento seguro de control de interfaz segura según una o más realizaciones de la presente invención; El flujo de proceso 1500 superpone una entidad segura 1502 (por ejemplo, un huésped o contenedor seguro), un control de interfaz segura 1504 y una entidad no confiable 1506 (por ejemplo, un hipervisor o un sistema operativo) para ilustrar qué operación está realizando un componente del entorno seguro.

El flujo 1500 de proceso está en el bloque 1510, donde la entidad 1502 segura emite una instrucción que puede permitir interrupciones (por ejemplo, LPSW y LCTL), que está siendo monitorizada por la entidad no confiable 1506. En el bloque 1520, el control 1504 de interfaz segura obtiene una nueva PSW (por ejemplo, para LPSW) o un valor de registro de control (por ejemplo, para LCTL) del almacenamiento huésped seguro. En el bloque 1530, el control 1504 de interfaz segura carga la nueva PSW o registro de control en el estado de entidad segura, que puede ser en respuesta a la búsqueda.

En el bloque 1540, el control 1504 de interfaz segura notifica a la entidad 1506 no confiable las actualizaciones de habilitación de interrupciones para huéspedes. En el bloque 1550, la entidad 1506 no confiable prioriza las interrupciones pendientes y habilitadas para determinar la interrupción del huésped habilitada de mayor prioridad. En el bloque 1560, la entidad 1506 no confiable almacena la información de interrupción para la interrupción de mayor prioridad (por ejemplo, la interrupción de huésped habilitada y de mayor prioridad) en un almacenamiento no seguro. En un ejemplo, este almacenamiento no seguro puede ser la descripción del estado asociada a esta interrupción del huésped. En otro ejemplo, este almacenamiento no seguro es el bloque de parámetros que se utilizará como entrada en una instrucción UVC de interrupción de inyección. El flujo de proceso 1500 pasa al Círculo Z, que está conectado a la FIG. 16 y al flujo 1600 de proceso.

Volviendo ahora a la FIG. 16, se representa un flujo 1600 de proceso para intercepción de instrucciones de alto nivel de control de interfaz segura para habilitar la interrupción según una o más realizaciones de la presente invención; El flujo 1600 de proceso se superpone a una entidad 1602 segura, un control 1604 de interfaz segura y una entidad 1606 no confiable para ilustrar qué operación está realizando un componente del entorno seguro. Obsérvese que la entidad 1602 segura, el control 1604 de interfaz segura y la entidad 1606 no confiable de la FIG. 16 son similares a la entidad 1502 segura, el control 1504 de interfaz segura y la entidad 1506 no confiable de la FIG. 16.

El flujo 1600 de proceso se encuentra en el bloque 1610, donde la entidad 1606 no confiable solicita al control 1604 de interfaz segura que presente la interrupción de huésped habilitada y de mayor prioridad. Esta solicitud puede ser en respuesta a la notificación. En un ejemplo, esta solicitud puede ser un envío del SIE con una indicación de que se debe inyectar una interrupción e información sobre esa interrupción. En otro ejemplo, esta solicitud puede ser una instrucción UVC de Interrupción de Inyección con la información de interrupción y una indicación del huésped asociado en el bloque de parámetros asociado a la UVC. En el bloque 1620 de decisión, el control 1604 de interfaz segura determina si la inyección de interrupción es válida. Por ejemplo, el control 1604 de interfaz segura determina si el huésped asociado está habilitado para la interrupción que se está inyectando. Si la inyección de interrupción no es válida, el flujo 1600 del proceso pasa al bloque 1640 (como se muestra mediante la flecha NO). En el bloque 1640, el control 1604 de interfaz de segura emite una excepción a la entidad 1606 no confiable.

Si la inyección de interrupción no es válida, el flujo 1600 del proceso pasa al bloque 1660 (como se muestra mediante la flecha SI). En el bloque 1660, el control 1604 de interfaz segura mueve la información de interrupción a una página de prefijos de huésped e inyecta la interrupción en la entidad 1602 segura (por ejemplo, actualizando el estado de huésped). En el bloque 1670, la entidad 1602 segura ejecuta un gestor de interrupciones en respuesta a la recepción de la interrupción inyectada.

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

La informática en la nube es un modelo de distribución de servicios para permitir un acceso de red conveniente y bajo demanda a un conjunto compartido de recursos informáticos configurables (por ejemplo, redes, ancho de banda de red, servidores, procesamiento, memoria, almacenamiento, aplicaciones, VM y servicios) que se pueden aprovisionar y liberar 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 las siguientes:

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

Acceso de red amplia: las capacidades están disponibles a través de una red y se accede a través de mecanismos estándar que promueven el uso por plataformas de cliente heterogéneas ligeras o pesadas (por ejemplo, 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 percepción de independencia de ubicación ya 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 (por ejemplo, país, estado o centro de datos).

Elasticidad rápida: las capacidades se pueden aprovisionar rápida y elásticamente, en algunos casos de manera automática, para ampliar y liberar rápidamente para reducir rápidamente. Para el consumidor, las capacidades disponibles para el aprovisionamiento a menudo parecen ser ilimitadas y pueden adquirirse en cualquier cantidad y 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 (por ejemplo, almacenamiento, procesamiento, ancho de banda y cuentas de usuario activas). El uso de recursos se puede monitorizar, controlar y notificar, proporcionando transparencia tanto para el proveedor como para el consumidor del servicio utilizado.

Los modelos de servicio son los siguientes:

Software como un servicio (SaaS): la capacidad proporcionada al consumidor es para usar 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 ligera, tal como un navegador web (por ejemplo, correo electrónico basado en web). El consumidor no gestiona ni controla la infraestructura en la nube subyacente, incluidas 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 de usuario.

Plataforma como un servicio (PaaS): la capacidad proporcionada al consumidor es para desplegar en la infraestructura en la nube aplicaciones 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 que incluye las redes, los servidores, los sistemas operativos o el almacenamiento, pero tiene control sobre las aplicaciones desplegadas y, posiblemente, las configuraciones de 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 los componentes de conexión en red seleccionados (por ejemplo, los cortafuegos del anfitrión).

Los Modelos de Implementación son los siguientes:

Nube privada: La infraestructura en la nube opera solamente para una organización. Se puede gestionar por la organización o 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 (por ejemplo, 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 está disponible para el público en general o para un gran grupo industrial y es propiedad de una organización que vende servicios en la nube.

Nube híbrida: la infraestructura en la nube es una composición de dos o más nubes (privadas, comunitarias o públicas) que siguen siendo entidades únicas, pero están unidas por una tecnología estandarizada o patentada que permite la portabilidad de los datos y aplicaciones (por ejemplo, ráfaga en la nube para el equilibrio de cargas entre nubes).

Un entorno de informática en la nube está orientado al 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. 17, se representa el entorno 50 de informática en la nube ilustrativo. Como se muestra, el entorno 50 de informática en la nube incluye uno o más nodos 10 de informática en la nube con los que pueden comunicarse los dispositivos informáticos locales usados por los consumidores de la nube, tales como, por ejemplo, asistente digital personal (PDA) o teléfono 54A móvil, ordenador 54B de sobremesa, ordenador 54C portátil y/o el sistema 54N informático de automóvil. Los nodos 10 pueden comunicarse entre sí. 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 50 de informática en la nube ofrezca infraestructura, plataformas y/o software como servicios para los que un consumidor de la nube no necesita mantener recursos en un dispositivo informático local. Se entiende que los tipos de dispositivos 54A-N informáticos mostrados en la FIG. 17 están destinados a ser únicamente ilustrativos y que los nodos 10 informáticos y el entorno 50 informático en la nube pueden comunicarse con cualquier tipo de dispositivo informatizado a través de cualquier tipo de red y/o conexión direccionable de red (por ejemplo, usando un navegador web).

Haciendo referencia ahora a la FIG. 18, se muestra un conjunto de capas de abstracción funcionales proporcionadas por el entorno 50 de informática en la nube (FIG. 17). Debe entenderse de antemano que los componentes, capas y funciones mostrados en la FIG. 18 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 60 de hardware y software incluye componentes de hardware y software. Ejemplos de componentes de hardware incluyen: ordenadores centrales 61; servidores 62 basados en la arquitectura RISC (Ordenador de Conjunto Reducido de Instrucciones); servidores 63; servidores 64 blade; dispositivos 65 de almacenamiento; y redes y componentes 66 de interconexión de redes. En algunas realizaciones, los componentes de software incluyen software 67 de servidor de aplicaciones de red y el software 68 de base de datos.

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

En un ejemplo, la capa 80 de gestión puede proporcionar las funciones que se describen a continuación. El aprovisionamiento 81 de recursos 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 comprobación de identidad para los consumidores y las tareas en la nube, así como protección para los datos y otros recursos. El portal 83 de usuario proporciona acceso al entorno informático en la nube para consumidores y administradores de sistemas. La gestión 84 de nivel de servicio proporciona la asignación y gestión de recursos informáticos en la nube de tal 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 90 de cargas de trabajo proporciona ejemplos de funcionalidad para la que se puede utilizar el entorno de informática en la nube. Ejemplos de cargas de trabajo y funciones que se pueden proporcionar desde esta capa incluyen: correspondencia y navegación 91; desarrollo de software y gestión del ciclo de vida 92; distribución de educación en el aula virtual 93; procesamiento de analíticas de datos 94; procesamiento de transacciones 95; y procesamiento de instrucciones 96. Se entiende que estos son solo algunos ejemplos y que, en otras realizaciones, las capas pueden incluir diferentes servicios.

Volviendo ahora a la FIG. 19, se representa un sistema 1900 de acuerdo con una o más realizaciones de la presente invención. El sistema 1900 incluye un nodo 10 de ejemplo (por ejemplo, un nodo anfitrión) que está en comunicación directa o indirecta con uno o más dispositivos 20A-20E cliente, tal como a través de una red 165. El nodo 10 puede ser un centro de datos o un servidor anfitrión de un proveedor de informática en la nube. El nodo 10 ejecuta un hipervisor 12, lo que facilita el despliegue de una o más VM 15 (15A-15N). El nodo 10 incluye además una capa 11 de hardware/firmware que proporciona soporte directo para las funciones requeridas por las VM 15A-N y el hipervisor 12, así como facilita que el hipervisor 12 proporcione uno o más servicios a las VM 15. En las implementaciones contemporáneas, la comunicación se proporciona entre la capa 11 de hardware/firmware y el hipervisor 12, entre la capa 11 de hardware/firmware y las VM 15, entre el hipervisor 12 y las VM 15, y entre el hipervisor 12 y las VM 15 a través de la capa 11 de hardware/firmware. Según una o más realizaciones de la presente invención, se proporciona un control de interfaz segura en la capa 11 de hardware/firmware, y se elimina la comunicación directa entre el hipervisor 12 y las VM 15.

Por ejemplo, el nodo 10 puede facilitar que un dispositivo 20A cliente implemente una o más de las VM 15A-15N. Las VM 15A-15N pueden desplegarse en respuesta a solicitudes respectivas de los distintos dispositivos 20A-20E cliente.

Por ejemplo, la VM 15A puede desplegarse mediante el dispositivo 20A cliente, la VM 15B puede desplegarse mediante el dispositivo 20B cliente y la VM 15C puede desplegarse mediante el dispositivo 20C cliente. El nodo 10 también puede facilitar que un cliente aprovisione un servidor físico (sin ejecutarlo como una VM). Los ejemplos descritos en la presente memoria incorporan el aprovisionamiento de recursos en el nodo 10 como parte de una VM, sin embargo, las soluciones técnicas descritas también se pueden aplicar para aprovisionar los recursos como parte de un servidor físico.

En un ejemplo, los dispositivos 20A-20E cliente pueden pertenecer a la misma entidad, tal como una persona, una empresa, una agencia gubernamental, un departamento dentro de una empresa o cualquier otra entidad, y el nodo 10 se puede operar como una nube privada de la entidad. En este caso, el nodo 10 aloja únicamente las VM 15A-15N que se despliegan por los dispositivos 20A-20E cliente que pertenecen a la entidad. En otro ejemplo, los dispositivos 20A-20E cliente pueden pertenecer a entidades distintas. Por ejemplo, una primera entidad puede poseer el dispositivo 20A cliente, mientras que una segunda entidad puede poseer el dispositivo 20B cliente. En este caso, el nodo 10 se puede operar como una nube pública que aloja las VM de diferentes entidades. Por ejemplo, las VM 15A-15N pueden implementarse de manera oculta, en la que la VM 15A no facilita el acceso a la VM 15B. Por ejemplo, el nodo 10 puede ocultar las VM 15A-15N usando una característica de Partición Lógica (LPAR) de Gestor de Sistemas/Recurso de Procesador (PR/SM) de IBM z Systems®. Estas características, tal como LPAR PR/SM, proporcionan aislamiento entre particiones, facilitando así que el nodo 10 despliegue dos o más VM 15A-15N para diferentes entidades en el mismo nodo 10 físico en diferentes particiones lógicas.

Un dispositivo 20A cliente de los dispositivos 20A-20e cliente es un aparato de comunicación tal como un ordenador, un teléfono inteligente, un ordenador tableta, un ordenador de sobremesa, un ordenador portátil, un ordenador de servidor o cualquier otro aparato de comunicación que solicite la implementación de una VM por el hipervisor 12 del nodo 10. El dispositivo 20A cliente puede enviar una solicitud para su recepción por el hipervisor a través de la red 165. Una VM 15A, de las VM 15A-15N, es una imagen de VM que el hipervisor 12 despliega en respuesta a una solicitud del dispositivo 20A cliente de los dispositivos 20A-20e cliente. El hipervisor 12 es un monitor de VM (VMM), que puede ser software, firmware o hardware que crea y ejecuta las VM. El hipervisor 12 facilita que la VM 15A use los componentes de hardware del nodo 10 para ejecutar programas y/o almacenar datos. Con las características y modificaciones apropiadas, el hipervisor 12 puede ser z Systems® de IBM, VM Server de Oracle, XenServer de Citrix, ESX de VMware, el hipervisor Hyper-V de Microsoft de o cualquier otro hipervisor. El hipervisor 12 puede ser un hipervisor nativo que se ejecuta directamente en el nodo 10, o un hipervisor alojado que se ejecuta en otro hipervisor.

Volviendo ahora a la FIG. 20, se muestra un nodo 10 para implementar las enseñanzas de la presente memoria según una o más realizaciones de la invención. El nodo 10 puede ser una plataforma informática electrónica que comprenda y/o emplee cualquier número y combinación de dispositivos informáticos y redes que utilicen varias tecnologías de comunicación, tal y como se describe en la presente memoria. El nodo 10 puede ser fácilmente escalable, extensible y modular, con la capacidad de cambiar a diferentes servicios o reconfigurar algunas características independientemente de otras.

En esta realización, el nodo 10 tiene un procesador 2001, que puede incluir una o más unidades 2001a, 2001b, 2001c, etc., centrales de procesamiento (CPU). El procesador 2001, también denominado circuito de procesamiento, microprocesador, unidad informática, se acopla mediante un bus 2002 de sistema a una memoria 2003 de sistema y a varios otros componentes. La memoria 2003 de sistema incluye la memoria 2004 de solo lectura (ROM) y la memoria 2005 de acceso aleatorio (RAM). La ROM 2004 se acopla al bus 2002 de sistema y puede incluir un sistema básico de entrada/salida (BIOS), que controla determinadas funciones básicas del nodo 10. La RAM es una memoria de lectura-escritura acoplada al bus 2002 de sistema para su uso por el procesador 2001.

El nodo 10 de la FIG. 20 incluye un disco 2007 duro, que es un ejemplo de un medio de almacenamiento tangible legible ejecutable por el procesador 2001. El disco 2007 duro almacena el software 2008 y los datos 2009. El software 2008 se almacena como instrucciones para ejecución en el nodo 10 por el procesador 2001 (para realizar un proceso, tal como los procesos descritos con referencia a las FIG. 1-19). Los datos 2009 incluyen un conjunto de valores de variables cualitativas o cuantitativas organizados en varias estructuras de datos para soportar y ser usados por las operaciones del software 2008.

El nodo 10 de la FIG. 20 incluye uno o más adaptadores (por ejemplo, controladores de disco duro, adaptadores de red, adaptadores gráficos, etc.) que interconectan y soportan las comunicaciones entre el procesador 2001, la memoria 2003 del sistema, el disco 2007 duro y otros componentes del nodo 10 (por ejemplo, dispositivos periféricos y externos). En una o más realizaciones de la presente invención, el uno o más adaptadores pueden conectarse a uno o más buses I/O que se conectan al bus 2002 de sistema a través de un puente de bus intermedio, y el uno o más buses I/O pueden utilizar protocolos comunes, tales como la Interconexión de Componentes Periféricos (PCI).

Como se muestra, el nodo 10 incluye un adaptador 2020 de interfaz que interconecta un teclado 2021, un ratón 2022, un altavoz 2023 y un micrófono 2024 al bus 2002 de sistema. El nodo 10 incluye un adaptador 2030 de elemento de visualización que interconecta el bus 2002 de sistema a un elemento de visualización 2031. El adaptador 2030 de elemento de visualización (y/o el procesador 2001) pueden incluir un controlador de gráficos para proporcionar rendimiento de gráficos, tal como un elemento de visualización y gestión de una GUI 2032. Un adaptador 2041 de comunicaciones interconecta el bus 2002 de sistema con una red 2050 que permite al nodo 10 comunicarse con otros

sistemas, dispositivos, datos y software, tales como un servidor 2051 y una base de datos 2052. En una o más realizaciones de la presente invención, las operaciones del software 2008 y los datos 2009 pueden implementarse en la red 2050 por el servidor 2051 y la base de datos 2052. Por ejemplo, la red 2050, el servidor 2051 y la base de datos 2052 pueden combinarse para proporcionar iteraciones internas del software 2008 y los datos 2009 como una plataforma como un servicio, un software como un servicio y/o infraestructura como un servicio (por ejemplo, como una aplicación web en un sistema distribuido).

Las realizaciones descritas en la presente memoria están necesariamente enraizadas en la tecnología informática y, particularmente, en los servidores informáticos que alojan VM. Además, una o más realizaciones de la presente invención facilitan una mejora en la operación de la propia tecnología informática, en particular los servidores informáticos que alojan VM, facilitando que los servidores informáticos que alojan VM alojen VM seguras, en las que incluso el hipervisor tiene prohibido acceder a la memoria, los registros y otros datos similares asociados a la VM segura. Además, una o más realizaciones de la presente invención proporcionan pasos significativos hacia las mejoras de los servidores informáticos de alojamiento de VM usando un control de interfaz segura (también denominado en la presente memoria "ultravisor" o "UV") que incluye hardware, firmware (por ejemplo, milicódigo) o una combinación de los mismos para facilitar una separación de la VM segura y el hipervisor y, por lo tanto, mantener una seguridad de las VM alojadas por el servidor informático. El control de interfaz segura proporciona operaciones intermedias ligeras para facilitar la seguridad, sin añadir una sobrecarga sustancial para asegurar el estado de la VM durante la inicialización/salida de las VM, como se describe en la presente memoria.

Las realizaciones de la invención descritas en la presente memoria pueden incluir un sistema, método y/o producto de programa informático (en la presente memoria, un sistema) que implementa el intercepción de instrucciones de alto nivel de control de interfaz segura para la habilitación de interrupciones. Se observa que, para cada una de las explicaciones, los identificadores de los elementos se reutilizan para otros elementos similares de diferentes figuras.

En la presente memoria se describen varias realizaciones de la invención con referencia a los dibujos relacionados. Se pueden idear realizaciones alternativas de la invención sin apartarse del alcance de esta invención. En la siguiente descripción y en los dibujos se exponen varias conexiones y relaciones posicionales (por ejemplo, por encima, por debajo, adyacentes, etc.) entre los elementos. Estas conexiones y/o relaciones posicionales, a menos que se especifique lo contrario, pueden ser directas o indirectas, y la presente invención no pretende ser limitante a este respecto. Por consiguiente, un acoplamiento de entidades puede referirse a un acoplamiento directo o indirecto, y una relación posicional entre entidades puede ser una relación posicional directa o indirecta. Además, las diversas tareas y pasos del proceso descritas en la presente memoria pueden incorporarse en un procedimiento o proceso más completo que tenga pasos o funcionalidad adicionales no descritas en detalle en la presente memoria.

Deben usarse las siguientes definiciones y abreviaturas para la interpretación de las reivindicaciones y la memoria descriptiva. Tal como se usan en la presente memoria, los términos "comprende", "que comprende", "incluye", "que incluye", "tiene", "que tiene", "contiene" o "que contiene" u otras variaciones de los mismos, se pretende que cubran una inclusión no exclusiva. Por ejemplo, una composición, una mezcla, proceso, método, artículo o aparato que comprenden una lista de elementos no se limitan necesariamente solo a esos elementos, sino que pueden incluir otros elementos no enumerados expresamente o inherentes a dicha composición, mezcla, proceso, método, artículo o aparato.

Además, el término "ejemplar" se usa en la presente memoria para significar “que sirve como un ejemplo, caso o ilustración”. Cualquier realización o diseño descrito en la presente memoria como "de ejemplo" no debe interpretarse necesariamente como preferido o ventajoso sobre otras realizaciones o diseños. Puede entenderse que los términos "al menos uno" y "uno o más" incluyen cualquier número entero mayor o igual a uno, es decir, uno, dos, tres, cuatro, etc. Se puede entender que los términos "una pluralidad" incluyen cualquier número entero mayor que o igual a dos, es decir, dos, tres, cuatro, cinco, etc. El término "conexión" puede incluir tanto una "conexión" indirecta como una "conexión" directa.

Los términos "alrededor", "sustancialmente", "aproximadamente" y sus variaciones pretenden incluir el grado de error asociado con la medición de la cantidad particular en función del equipo disponible en el momento de presentar la solicitud. Por ejemplo, "alrededor" puede incluir un intervalo de ± 8% o 5%, o 2% de un valor dado.

La presente invención puede ser un sistema, un método y/o un producto de programa informático en cualquier nivel posible de detalle técnico de integración. El producto de programa informático puede incluir un medio (o medios) de almacenamiento legible por ordenador que tenga 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 de acceso aleatorio estática (SRAM), un disco compacto portátil de memoria de solo lectura (CD-ROM), un disco versátil digital (DVD), una tarjeta de memoria, un disco, 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, tal como se usa en la presente memoria, 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 otro medio de transmisión (por ejemplo, pulsos de luz que pasan a través de un cable de fibra óptica) o señales eléctricas transmitidas a través de un cable.

Las instrucciones de programa legibles por ordenador descritas en la presente memoria pueden descargarse a dispositivos informáticos/de procesamiento respectivos 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, cortafuegos, conmutadores, ordenadores de puerta de enlace y/o servidores periféricos. Una tarjeta de adaptador de red o interfaz de red en cada dispositivo informático/de procesamiento recibe instrucciones de programa legibles por ordenador desde la red y reenvía 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 circuito integrado o código fuente o código objeto escritos en cualquier combinación de uno o más lenguajes de programación, incluyendo un lenguaje de programación orientado a objetos como Smalltalk, C++ o similares, y lenguajes de programación procedimentales, como el lenguaje de programación "C" o lenguajes de programación similares. Las instrucciones de programa legibles por ordenador pueden ejecutarse completamente en el ordenador del usuario, parcialmente en el ordenador del usuario, como un paquete de software independiente, parcialmente en el ordenador del usuario y parcialmente en un ordenador remoto o completamente en el ordenador o servidor. En este último escenario, el ordenador remoto puede conectarse al ordenador del usuario a través de cualquier tipo de red, que incluye 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 usando un Proveedor de Servicios de Internet). En algunas realizaciones, circuitos electrónicos que incluyen, por ejemplo, circuitos lógicos programables, matrices de puertas programables en campo (FPGA) o matrices lógicas programables (PLA) pueden ejecutar instrucciones de programa legibles por ordenador utilizando información de estado de las instrucciones de programa legibles por ordenador para personalizar los circuitos electrónicos, con el fin de realizar aspectos de la presente invención.

Los aspectos de la presente invención se describen en la presente memoria con referencia a ilustraciones de diagramas de flujo y/o diagramas de bloques de métodos, aparatos (sistemas) y productos de programas informáticos según las realizaciones de la invención. Se entenderá que cada bloque de las ilustraciones de diagrama de flujo y/o diagramas de bloques, y las combinaciones de bloques en las ilustraciones de diagrama de flujo y/o 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, un ordenador de propósito especial u otro aparato de procesamiento de datos programable para producir una máquina, de manera que las instrucciones, que se ejecutan a través del procesador del ordenador u otro aparato programable de procesamiento de datos, creen medios para implementar las funciones/acciones especificadas en el bloque o bloques del diagrama de flujo y/o 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 tal manera que el medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo comprende un artículo de fabricación que incluye instrucciones que implementen aspectos de la función/acciones especificadas en el bloque o bloques en el diagrama de flujo y/o diagrama de bloques.

Las instrucciones de programa legibles por ordenador también pueden cargarse en un ordenador, otro aparato de procesamiento de datos programable u otro dispositivo para hacer que se realicen una serie de pasos operativos en el ordenador, otro aparato programable u otro dispositivo para producir un proceso implementado por ordenador, de tal manera que las instrucciones que se ejecutan en el ordenador, en otro aparato programable o en otro dispositivo implementen las funciones/acciones especificadas en el diagrama de flujo y/o diagrama de bloques, bloque o bloques.

El diagrama de flujo y los diagramas de bloques de las Figuras ilustran la arquitectura, funcionalidad y operación de posibles implementaciones de sistemas, métodos y productos de programas informáticos según varias realizaciones de la presente invención. A este respecto, cada bloque en el diagrama de flujo o los diagramas de bloques puede representar un módulo, un segmento o una parte 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 usada en la presente memoria tiene el propósito de describir las realizaciones particulares únicamente y no se pretende que sea limitante. Como se usa en la presente memoria, las formas singulares "un", "una" y "el" se pretenden 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, componentes de elementos y/o grupos de los mismos.

Las descripciones de las varias realizaciones de la presente memoria se han presentado con propósitos de ilustración, pero no se pretende que sean exhaustivas ni se limiten a las realizaciones descritas. Muchas modificaciones y alteraciones serán evidentes para los expertos en la materia sin apartarse del alcance de las realizaciones descritas. La terminología usada en la presente memoria se eligió para explicar mejor los principios de las realizaciones, la aplicación práctica o la mejora técnica sobre las tecnologías que se encuentran en el mercado, o para permitir que otros expertos en la materia entiendan las realizaciones descritas en la presente memoria.

REIVINDICACIONES

1. Un método, que comprende:

buscar, mediante un control (1504) de interfaz segura de un ordenador (10) que proporciona una interpretación parcial de instrucciones para una instrucción que permite una interrupción, una palabra de estado de programa o un valor de registro de control desde un almacenamiento (100) de huésped seguro;

notificar, mediante el control (1504) de interfaz segura, a una entidad (1506) no confiable las actualizaciones de la máscara de interrupción del huésped, ejecutándose la entidad (1506) no confiable sobre y en comunicación con el hardware del ordenador (10) a través del control (1504) de interfaz segura para soportar las operaciones de una entidad (1502) segura que se ejecuta en la entidad (1506) no confiable;

recibir, mediante el control (1504) de interfaz segura, de la entidad (1506) no confiable, una solicitud para presentar una interrupción para huéspedes habilitada y de máxima prioridad en respuesta a la notificación de las actualizaciones de máscara de interrupción de huésped; y

mover, mediante el control (1504) de interfaz segura la información de interrupción a una página de prefijos de huésped e inyectando la interrupción en la entidad (1502) segura cuando se determina que una inyección de la interrupción es válida.

en donde el control (1504) de interfaz segura incluye una interfaz segura implementada en hardware y/o firmware interno, seguro y confiable entre la entidad (1506) no confiable y la entidad (1502) segura, la interfaz segura incluyendo un mecanismo de hardware que impide que la entidad no confiable acceda al contenido del almacenamiento (100) de huésped seguro.

2. El método de la reivindicación 1, que comprende, además:

emitir, por parte de la entidad (1502) segura, la palabra de estado del programa de carga o el control de carga que está siendo supervisado por la entidad (1506) no confiable.

3. El método de la reivindicación 1 o la reivindicación 2, que comprende, además:

cargar, mediante el control (1504) de interfaz segura, la palabra de estado de programa o el registro de control en respuesta a la búsqueda.

4. El método según una cualquiera de las reivindicaciones 1 a 3, que comprende, además:

priorizar, mediante la entidad 1506 no confiable, las interrupciones pendientes y habilitadas para determinar la interrupción de huésped habilitada de mayor prioridad.

5. El método según una cualquiera de las reivindicaciones 1 a 4, que comprende, además:

almacenar, mediante la entidad (1506) no confiable, la información de interrupción para la interrupción de huésped habilitada de mayor prioridad en un almacenamiento no seguro.

6. El método de la reivindicación 5, en donde la entidad (1506) no confiable proporciona la información de interrupción en una descripción de estado.

7. El método de la reivindicación 5, en donde la entidad (1506) no confiable emite una instrucción para proporcionar la información de interrupción al control (1504) de interfaz segura y la información de interrupción se pasa como un parámetro para la instrucción.

8. El método según una cualquiera de las reivindicaciones 1 a 7, que comprende, además:

emitir, mediante el control (1504) de interfaz segura, una excepción a la entidad (1506) no confiable cuando se determina que la inyección de la interrupción no es válida.

9. El método según una cualquiera de las reivindicaciones 1 a 8, que comprende, además:

ejecutar, por parte de la entidad (1502) segura, un gestor de interrupciones en respuesta a la recepción de la interrupción inyectada.

10. El método de la reivindicación 9, en donde la entidad (15A, 1502) segura comprende un huésped seguro y la entidad (1506) no confiable comprende un hipervisor (12).

11. Un producto de programa informático que comprende un medio de almacenamiento legible por ordenador que tiene instrucciones de programa incorporadas en el mismo, instrucciones de programa ejecutables por un ordenador (10) para hacer a las operaciones:

buscar, mediante un control (1504) de interfaz segura de un ordenador (10) que proporciona una interpretación parcial de instrucciones para una instrucción que permite una interrupción, una palabra de estado del programa o un valor de registro de control desde un almacenamiento (100) huésped seguro;

notificar, mediante el control (1504) de interfaz segura, a una entidad (1506) no confiable las actualizaciones de la máscara de interrupción de huésped, ejecutándose la entidad (1506) no confiable sobre y en comunicación con el hardware del ordenador (10) a través del control (1504) de interfaz segura para soportar las operaciones de una entidad (1502) segura que se ejecuta en la entidad (1506) no confiable;

recibir, mediante el control (1504) de interfaz segura, desde la entidad (1506) no confiable, una solicitud para presentar una interrupción de huésped habilitada y de máxima prioridad en respuesta a la notificación de las actualizaciones de máscara de interrupción de huésped; y

mover, mediante el control (1504) de interfaz segura la información de interrupción a una página de prefijos de huésped e inyectando la interrupción en la entidad (1502) segura cuando se determina que una inyección de la interrupción es válida.

en donde el control (1504) de interfaz segura incluye una interfaz segura implementada en hardware y/o firmware interno, seguro y confiable entre la entidad (1506) no confiable y la entidad (1502) segura, la interfaz segura incluyendo un mecanismo de hardware que impide que la entidad no confiable acceda al contenido del almacenamiento (100) huésped seguro.

12. El producto de programa informático de la reivindicación 11, en donde las instrucciones de programa son ejecutables además para hacer:

emitir, por parte de la entidad (1502) segura, la palabra de estado del programa de carga o el control de carga que está siendo supervisado por la entidad (1506) no confiable.

13. El producto de programa informático de la reivindicación 11 o 12, en donde las instrucciones de programa son ejecutables además para hacer:

cargar, mediante el control (1504) de interfaz segura, la palabra de estado del programa o el registro de control en respuesta a la búsqueda.

14. El producto de programa informático de una de las reivindicaciones 11 a 13, en donde las instrucciones de programa son ejecutables además para provocar:

priorizar, mediante la entidad (1506) no confiable, las interrupciones pendientes y habilitadas para determinar la interrupción de huésped habilitada de mayor prioridad.

15. El producto de programa informático de una de las reivindicaciones 11 a 14, en donde las instrucciones de programa son ejecutables además para hacer:

almacenar, por parte de la entidad (1506) no confiable, la información de interrupción para la interrupción de huésped habilitada de mayor prioridad en un almacenamiento no seguro.

16. El producto de programa informático de la reivindicación 15, en donde la entidad (1506) no confiable proporciona la información de interrupción en una descripción de estado.

17. El producto de programa informático de la reivindicación 15, en donde la entidad (1506) no confiable emite una instrucción para proporcionar la información de interrupción al control (1504) de interfaz segura y la información de interrupción se pasa como un parámetro para la instrucción.

18. El producto de programa informático de una de las reivindicaciones 11 a 17, en donde las instrucciones de programa son ejecutables además para hacer:

emitir, mediante el control (1504) de interfaz segura, una excepción a la entidad (1506) no confiable cuando se determina que la inyección de la interrupción no es válida.

19. El producto de programa informático de una de las reivindicaciones 11 a 18, en donde las instrucciones de programa son ejecutables además para hacer:

ejecutar, por parte de la entidad (1502) segura, un gestor de interrupciones en respuesta a la recepción de la interrupción inyectada.

20. El producto de programa informático de la reivindicación 19, en donde la entidad (15A, 1502) segura comprende un huésped seguro y la entidad (1506) no confiable comprende un hipervisor (12).

21. Un sistema que comprende:

un control (1504) de interfaz segura de un ordenador (10) que proporciona una interpretación parcial de instrucciones para una instrucción que permite una interrupción;

buscar, mediante el control (1504) de interfaz segura, una palabra de estado de programa o un valor de registro de control de un almacenamiento (100)de huésped seguro.

notificar, mediante el control (1504) de interfaz segura, a una entidad (1506) no confiable las actualizaciones de la máscara de interrupción de huésped, ejecutándose la entidad (1506) no confiable sobre y en comunicación con el hardware del ordenador (10) a través del control (1504) de interfaz segura para soportar las operaciones de una entidad (1502) segura que se ejecuta en la entidad (1506) no confiable;

recibir, mediante el control (1504) de interfaz segura desde la entidad (1506) no confiable, una solicitud para presentar una interrupción de huésped habilitada de mayor prioridad en respuesta a la notificación de las actualizaciones de máscara de interrupción de huésped.

mover, mediante el control (1504) de interfaz segura la información de interrupción a una página de prefijos de huésped e inyectando la interrupción en la entidad (1502) segura cuando se determina que una inyección de la interrupción es válida.

en donde el control (1504) de interfaz segura incluye una interfaz segura implementada en hardware y/o firmware interno, seguro y confiable entre la entidad (1506) no confiable y la entidad (1502) segura, la interfaz segura incluyendo un mecanismo de hardware que impide que la entidad no confiable acceda al contenido del almacenamiento (100) huésped seguro.

22. El sistema de la reivindicación 21, en donde el sistema es ejecutable para proporcionar las operaciones de:

emitir, por parte de la entidad (1502) segura, la palabra de estado del programa de carga o el control de carga que está siendo monitorizado por la entidad (1506) no confiable.

23. El sistema de la reivindicación 21 o 22, en donde el sistema es ejecutable para proporcionar las operaciones de:

cargar, mediante el control (1504) de interfaz segura, la palabra de estado del programa o el registro de control en respuesta a la búsqueda.

24. El sistema de una cualquiera de las reivindicaciones 21 a 23, en donde el sistema es ejecutable para proporcionar las operaciones de:

priorizar, mediante la entidad (1506) no confiable, las interrupciones pendientes y habilitadas para determinar la interrupción de huésped habilitada de mayor prioridad.

25. El sistema de una cualquiera de las reivindicaciones 21 a 24, en donde el sistema es ejecutable para proporcionar las operaciones de:

almacenar, por parte de la entidad (1506) no confiable, la información de interrupción para interrupción de huésped habilitada de mayor prioridad en un almacenamiento no seguro.

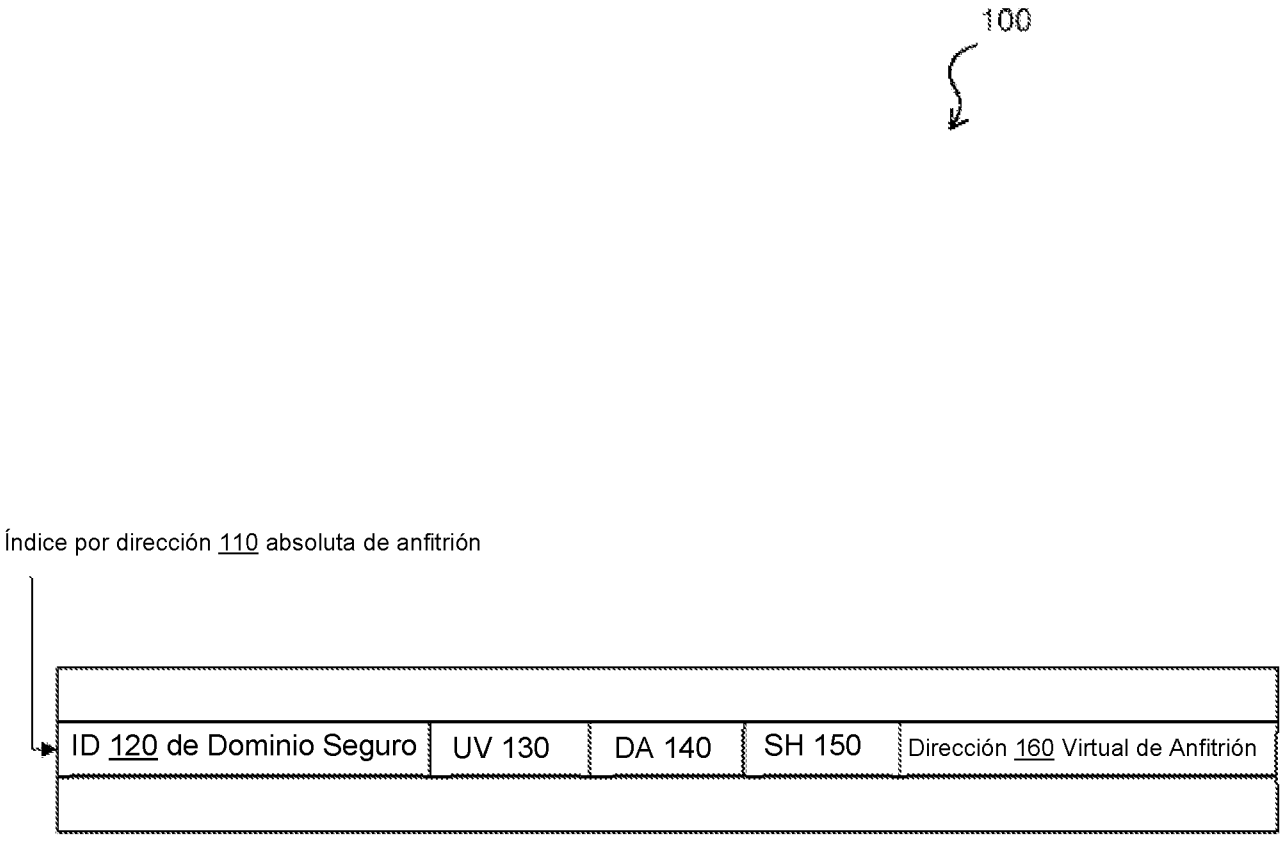
100
Índice por dirección 110 absoluta de anfitrión
ID 120 de Dominio Seguro
UV 130
DA 140
SH 150
Dirección 160 Virtual de Anfitrión

FIG. 1

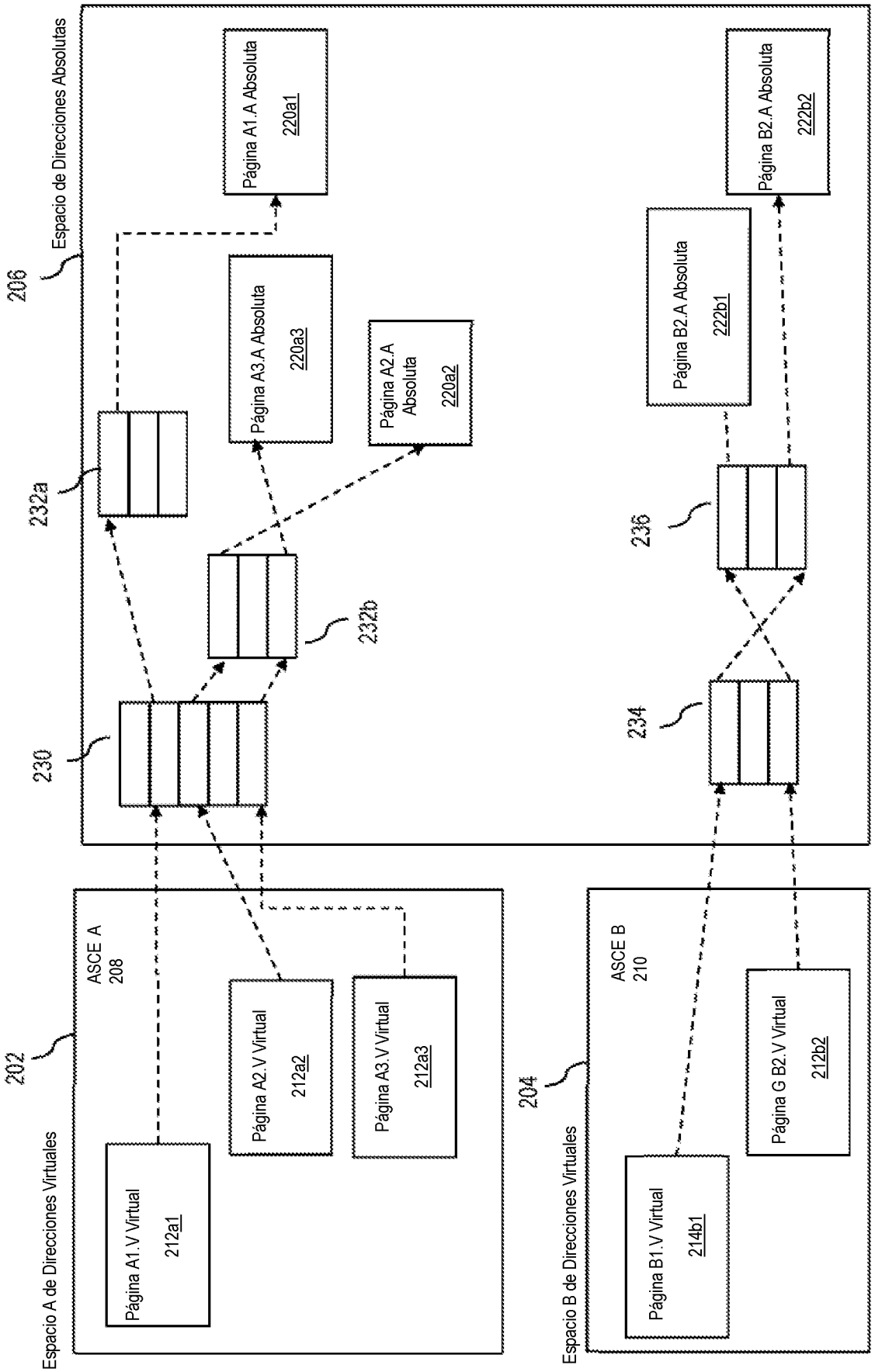
Espacio de Direcciones Absolutas
206
Página A1.A Absoluta
220a1
Página A3.A Absoluta
220a3
Página A2.A Absoluta
220a2
232a
232b
230
Página B2.A Absoluta
222b1
Página B2.A Absoluta
222b2
236
234
Espacio A de Direcciones Virtuales
202
ASCE A
208
Página A1.V Virtual
212a1
Página A2.V Virtual
212a2
Página A3.V Virtual
212a3
Espacio B de Direcciones Virtuales
204
ASCE B
210
Página B1.V Virtual
214b1
Página G B2.V Virtual
212b2

FIG. 2

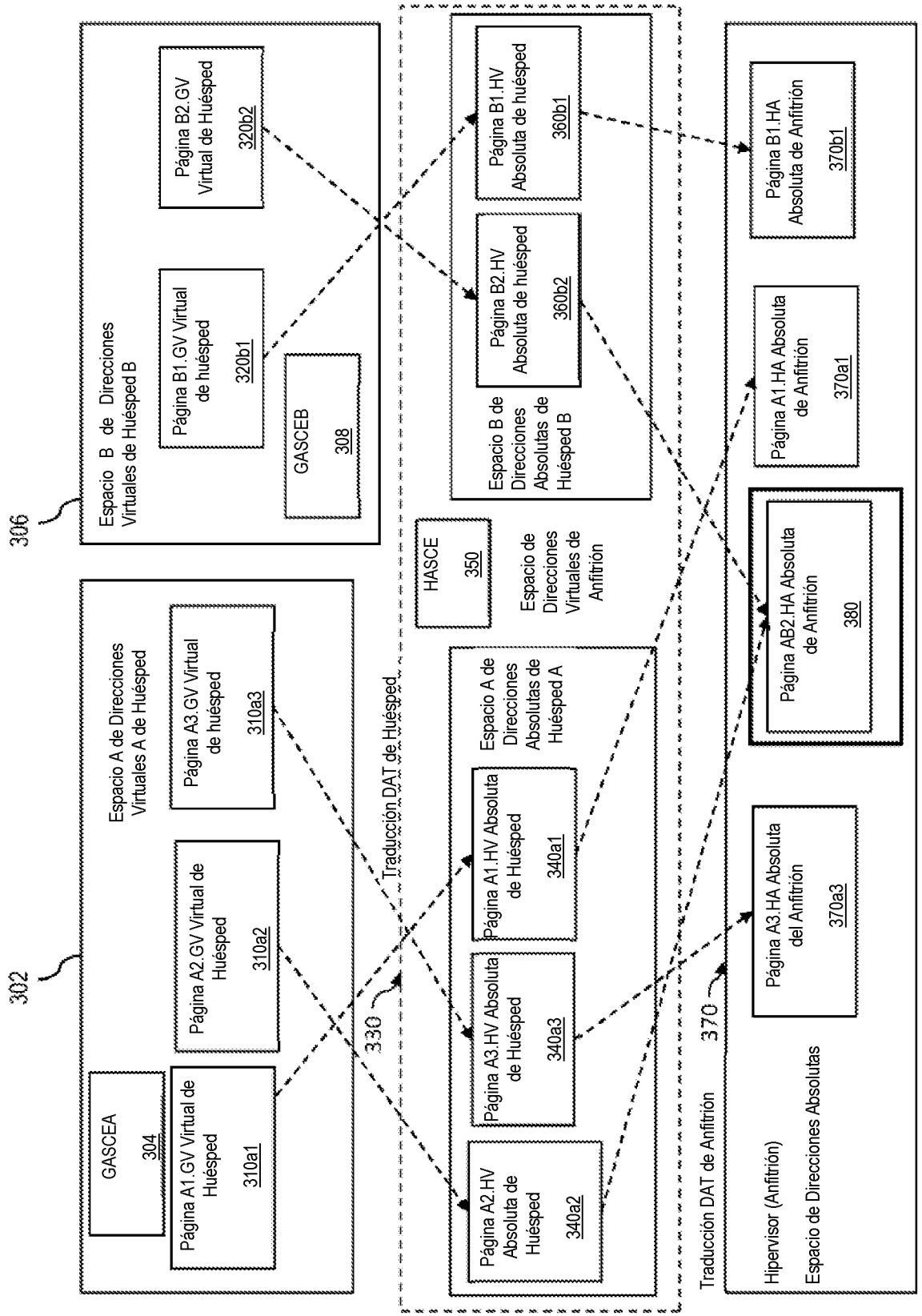
302
306
330
370
Espacio A de Direcciones Virtuales A de Huésped
Espacio B de Direcciones Virtuales de Huésped B
Traducción DAT de Huésped
Traducción DAT de Anfitrión
Hipervisor (Anfitrión)
Espacio de Direcciones Absolutas
GASCEA 304
Página A1.GV Virtual de Huésped 310a1
Página A2.GV Virtual de Huésped 310a2
Página A3.GV Virtual de huésped 310a3
GASCEB 308
Página B1.GV Virtual de huésped 320b1
Página B2.GV Virtual de Huésped 320b2
HASCE 350
Espacio de Direcciones Virtuales de Anfitrión
Espacio A de Direcciones Absolutas de Huésped A
Página A1.HV Absoluta de Huésped 340a1
Página A2.HV Absoluta de Huésped 340a2
Página A3.HV Absoluta de Huésped 340a3
Espacio B de Direcciones Absolutas de Huésped B
Página B1.HV Absoluta de huésped 360b1
Página B2.HV Absoluta de huésped 360b2
Página AB2.HA Absoluta de Anfitrión 380
Página A1.HA Absoluta de Anfitrión 370a1
Página A3.HA Absoluta del Anfitrión 370a3
Página B1.HA Absoluta de Anfitrión 370b1

FIG. 3

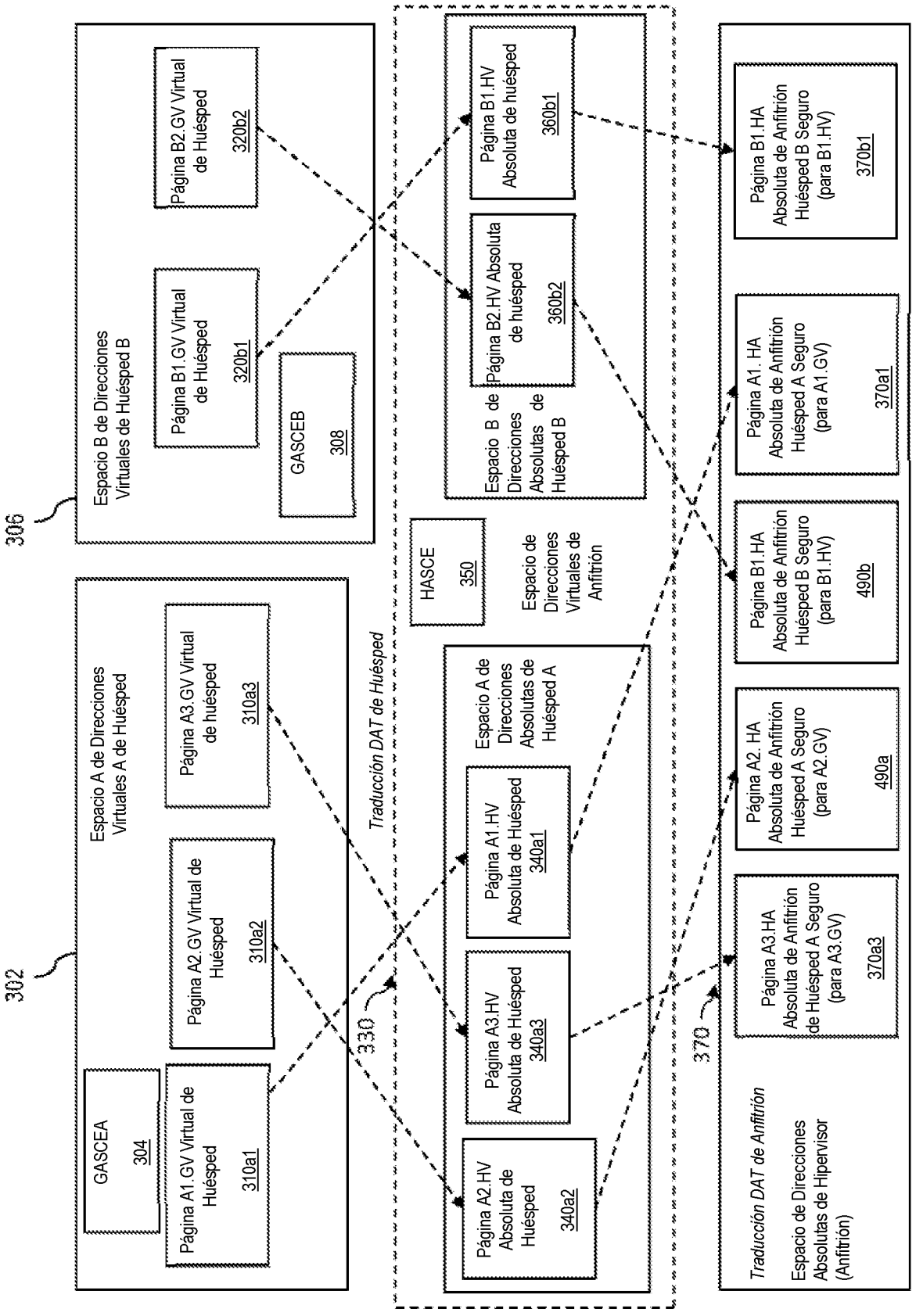
302
Espacio A de Direcciones Virtuales A de Huésped
GASCEA 304
Página A1.GV Virtual de Huésped 310a1
Página A2.GV Virtual de Huésped 310a2
Página A3.GV Virtual de huésped 310a3
306
Espacio B de Direcciones Virtuales de Huésped B
GASCEB 308
Página B1.GV Virtual de Huésped 320b1
Página B2.GV Virtual de Huésped 320b2
330
Traducción DAT de Huésped
HASCE 350
Espacio de Direcciones Virtuales de Anfitrión
Espacio A de Direcciones Absolutas de Huésped A
Página A1.HV Absoluta de Huésped 340a1
Página A3.HV Absoluta de Huésped 340a3
Página A2.HV Absoluta de Huésped 340a2
Espacio B de Direcciones Absolutas de Huésped B
Página B2.HV Absoluta de huésped 360b2
Página B1.HV Absoluta de huésped 360b1
370
Traducción DAT de Anfitrión
Espacio de Direcciones Absolutas de Hipervisor (Anfitrión)
Página A3.HA Absoluta de Anfitrión de Huésped A Seguro (para A3.GV) 370a3
Página A2. HA Absoluta de Anfitrión Huésped A Seguro (para A2.GV) 490a
Página B1.HA Absoluta de Anfitrión Huésped B Seguro (para B1.HV) 490b
Página A1. HA Absoluta de Anfitrión Huésped A Seguro (para A1.GV) 370a1
Página B1.HA Absoluta de Anfitrión Huésped B Seguro (para B1.HV) 370b1

FIG. 4

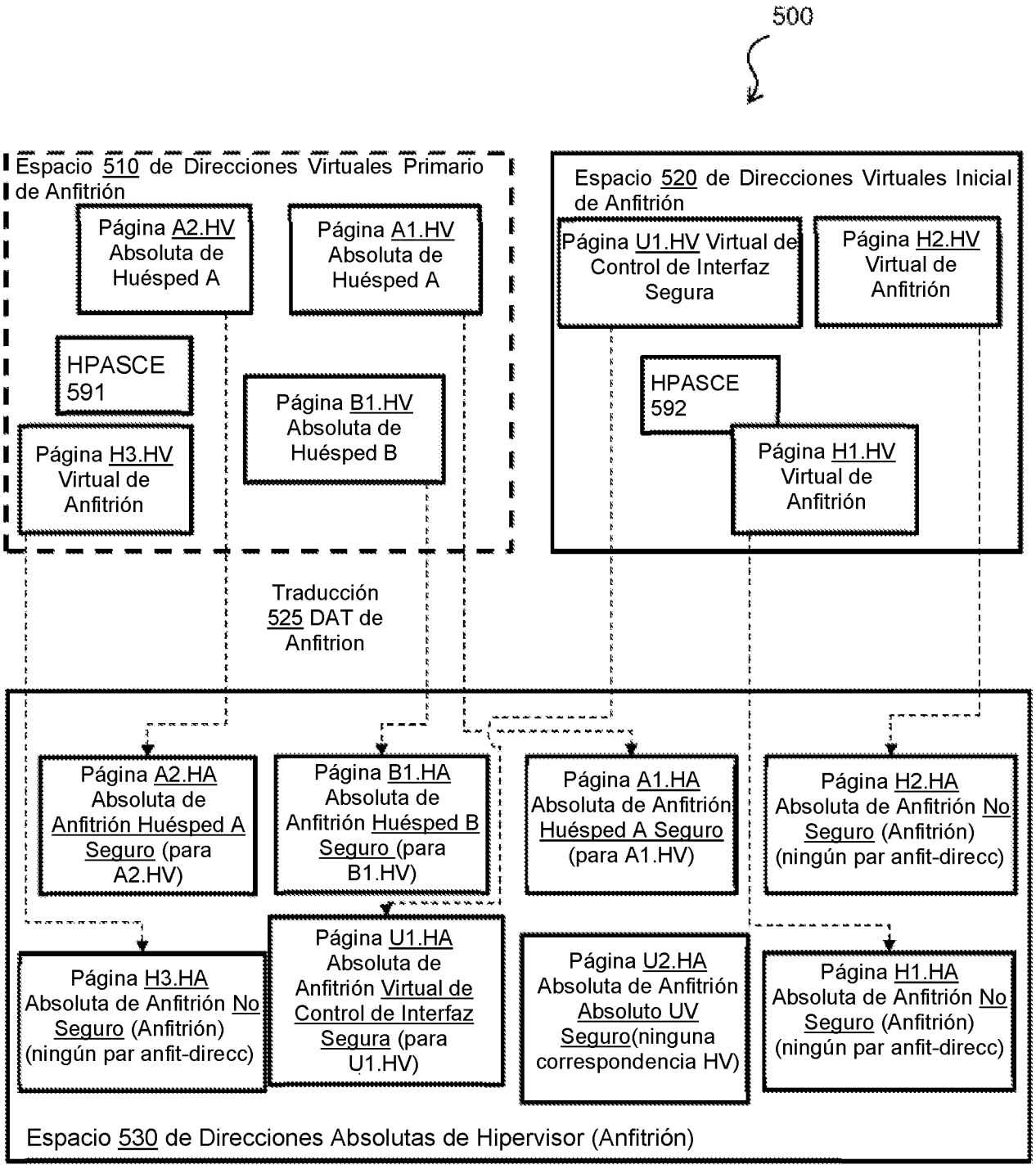
500
Espacio 510 de Direcciones Virtuales Primario de Anfitrión
Página A2.HV Absoluta de Huésped A
Página A1.HV Absoluta de Huésped A
HPASCE 591
Página B1.HV Absoluta de Huésped B
Página H3.HV Virtual de Anfitrión
Espacio 520 de Direcciones Virtuales Inicial de Anfitrión
Página U1.HV Virtual de Control de Interfaz Segura
Página H2.HV Virtual de Anfitrión
HPASCE 592
Página H1.HV Virtual de Anfitrión
Traducción 525 DAT de Anfitrion
Página A2.HA Absoluta de Anfitrión Huésped A Seguro (para A2.HV)
Página B1.HA Absoluta de Anfitrión Huésped B Seguro (para B1.HV)
Página A1.HA Absoluta de Anfitrión Huésped A Seguro (para A1.HV)
Página H2.HA Absoluta de Anfitrión No Seguro (Anfitrión) (ningún par anfit-direcc)
Página H3.HA Absoluta de Anfitrión No Seguro (Anfitrión) (ningún par anfit-direcc)
Página U1.HA Absoluta de Anfitrión Virtual de Control de Interfaz Segura (para U1.HV)
Página U2.HA Absoluta de Anfitrión Absoluto UV Seguro(ninguna correspondencia HV)
Página H1.HA Absoluta de Anfitrión No Seguro (Anfitrión) (ningún par anfit-direcc)
Espacio 530 de Direcciones Absolutas de Hipervisor (Anfitrión)

FIG. 5

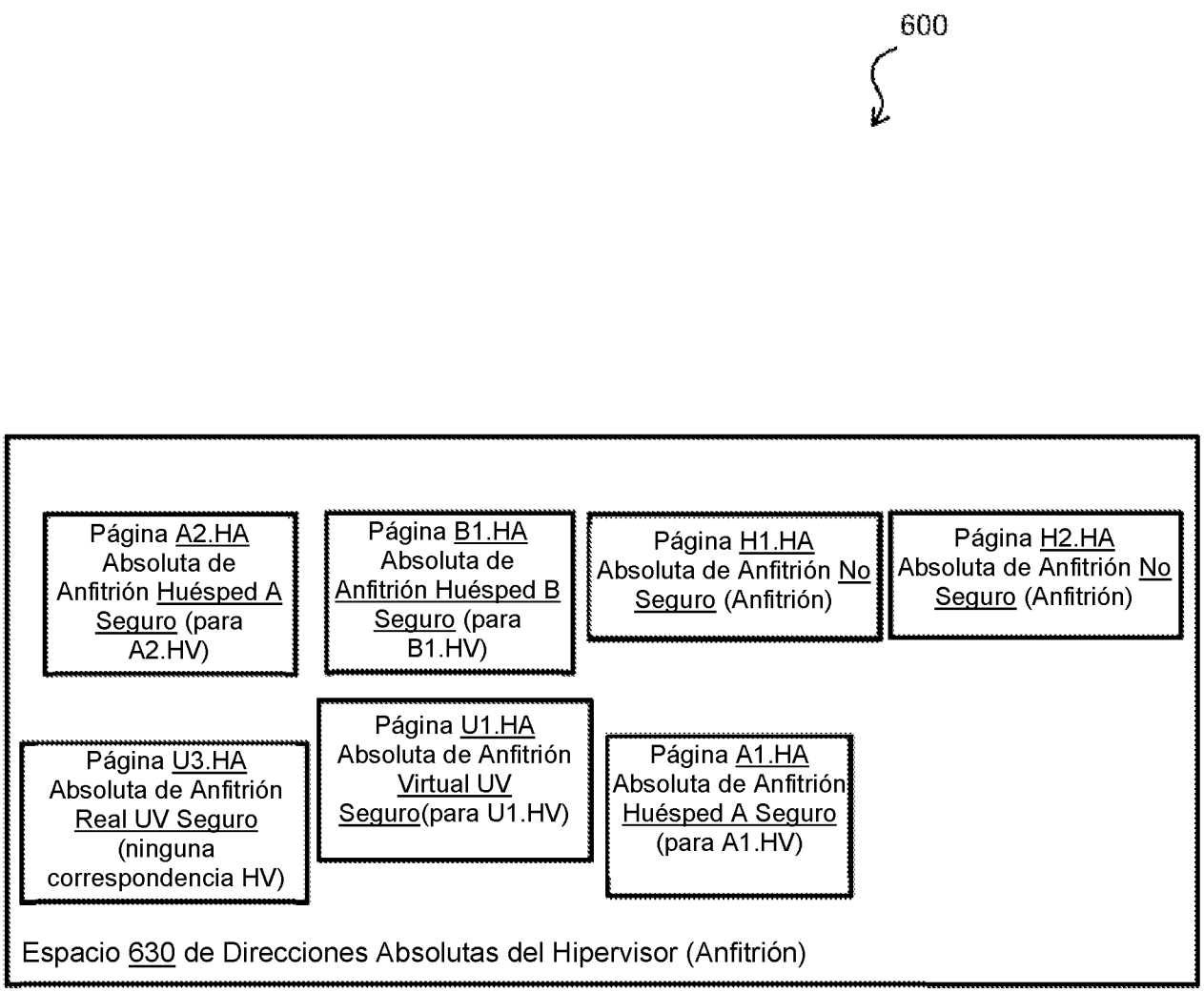
600
Página A2.HA Absoluta de Anfitrión Huésped A Seguro (para A2.HV)
Página B1.HA Absoluta de Anfitrión Huésped B Seguro (para B1.HV)
Página H1.HA Absoluta de Anfitrión No Seguro (Anfitrión)
Página H2.HA Absoluta de Anfitrión No Seguro (Anfitrión)
Página U3.HA Absoluta de Anfitrión Real UV Seguro (ninguna correspondencia HV)
Página U1.HA Absoluta de Anfitrión Virtual UV Seguro(para U1.HV)
Página A1.HA Absoluta de Anfitrión Huésped A Seguro (para A1.HV)
Espacio 630 de Direcciones Absolutas del Hipervisor (Anfitrión)

FIG. 6

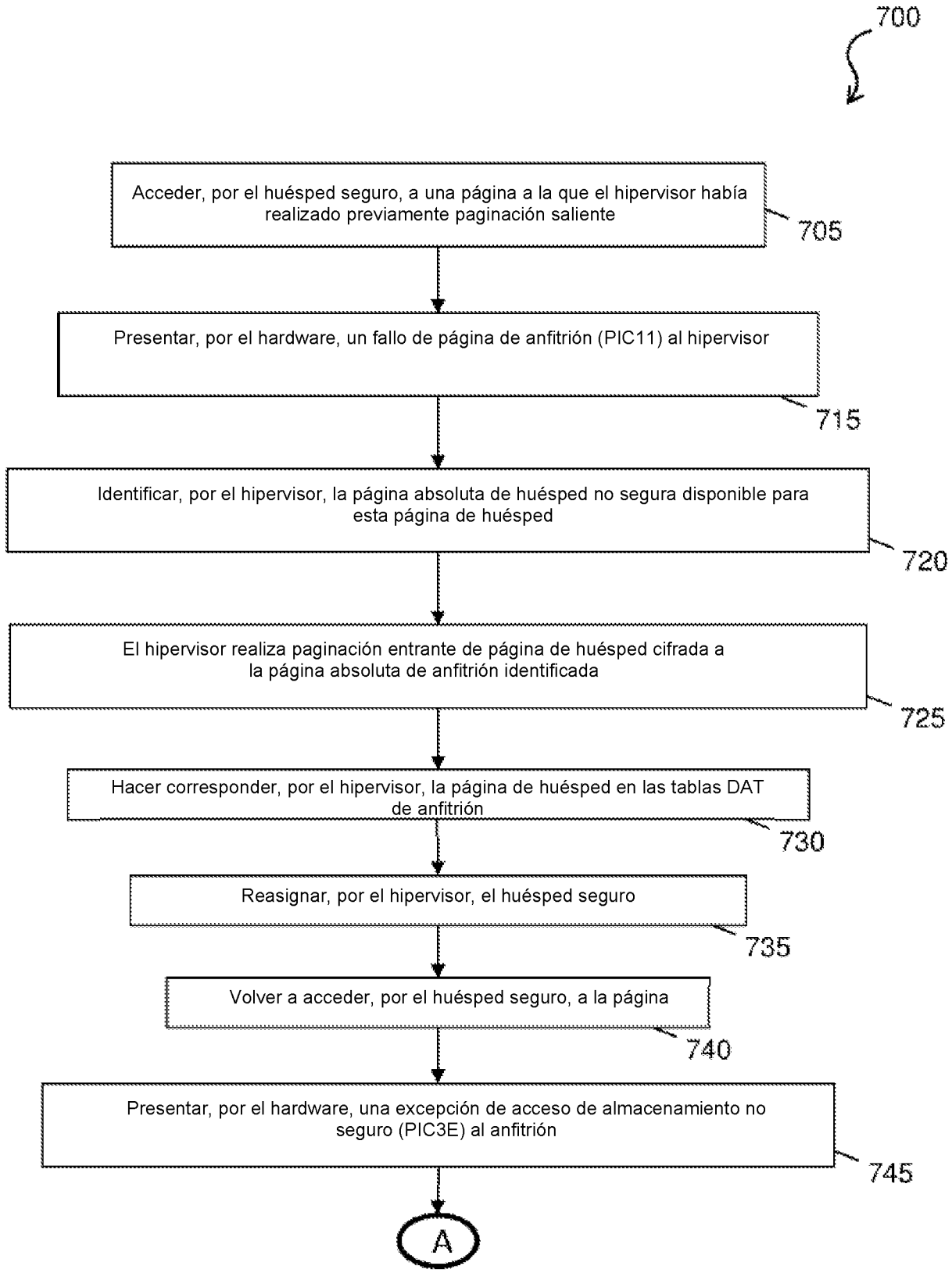
700
Acceder, por el huésped seguro, a una página a la que el hipervisor había realizado previamente paginación saliente
705
Presentar, por el hardware, un fallo de página de anfitrión (PIC11) al hipervisor
715
Identificar, por el hipervisor, la página absoluta de huésped no segura disponible para esta página de huésped
720
El hipervisor realiza paginación entrante de página de huésped cifrada a la página absoluta de anfitrión identificada
725
Hacer corresponder, por el hipervisor, la página de huésped en las tablas DAT de anfitrión
730
Reasignar, por el hipervisor, el huésped seguro
735
Volver a acceder, por el huésped seguro, a la página
740
Presentar, por el hardware, una excepción de acceso de almacenamiento no seguro (PIC3E) al anfitrión
745
A

FIG. 7

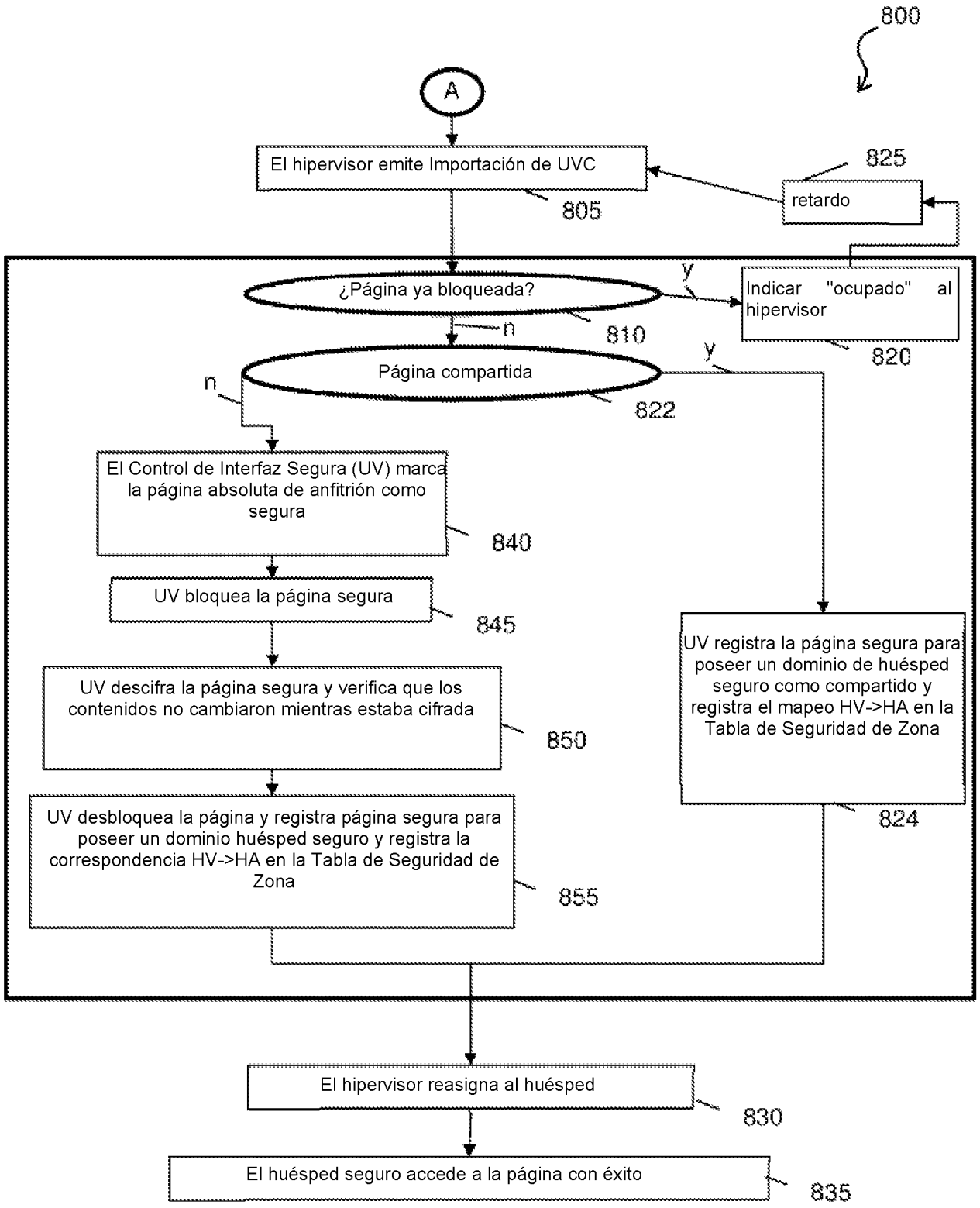
800
A
El hipervisor emite Importación de UVC
805
825
retardo
¿Página ya bloqueada?
y
Indicar "ocupado" al hipervisor
820
n
810
Página compartida
y
n
822
El Control de Interfaz Segura (UV) marca la página absoluta de anfitrión como segura
840
UV bloquea la página segura
845
UV descifra la página segura y verifica que los contenidos no cambiaron mientras estaba cifrada
850
UV registra la página segura para poseer un dominio de huésped seguro como compartido y registra el mapeo HV->HA en la Tabla de Seguridad de Zona
824
UV desbloquea la página y registra página segura para poseer un dominio huésped seguro y registra la correspondencia HV->HA en la Tabla de Seguridad de Zona
855
El hipervisor reasigna al huésped
830
El huésped seguro accede a la página con éxito
835

FIG. 8

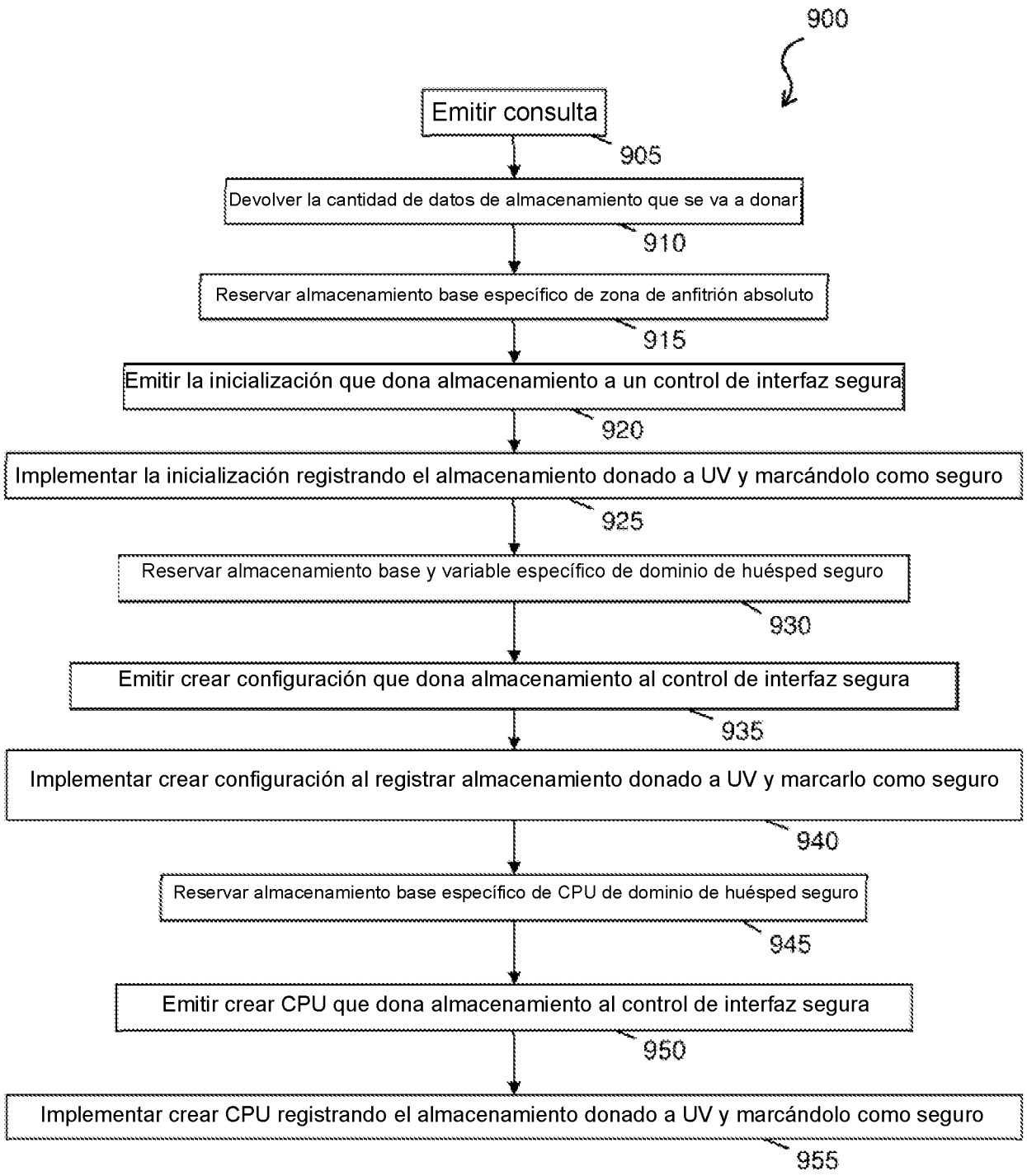
900
Emitir consulta
905
Devolver la cantidad de datos de almacenamiento que se va a donar
910
Reservar almacenamiento base específico de zona de anfitrión absoluto
915
Emitir la inicialización que dona almacenamiento a un control de interfaz segura
920
Implementar la inicialización registrando el almacenamiento donado a UV y marcándolo como seguro
925
Reservar almacenamiento base y variable específico de dominio de huésped seguro
930
Emitir crear configuración que dona almacenamiento al control de interfaz segura
935
Implementar crear configuración al registrar almacenamiento donado a UV y marcarlo como seguro
940
Reservar almacenamiento base específico de CPU de dominio de huésped seguro
945
Emitir crear CPU que dona almacenamiento al control de interfaz segura
950
Implementar crear CPU registrando el almacenamiento donado a UV y marcándolo como seguro
955

FIG. 9

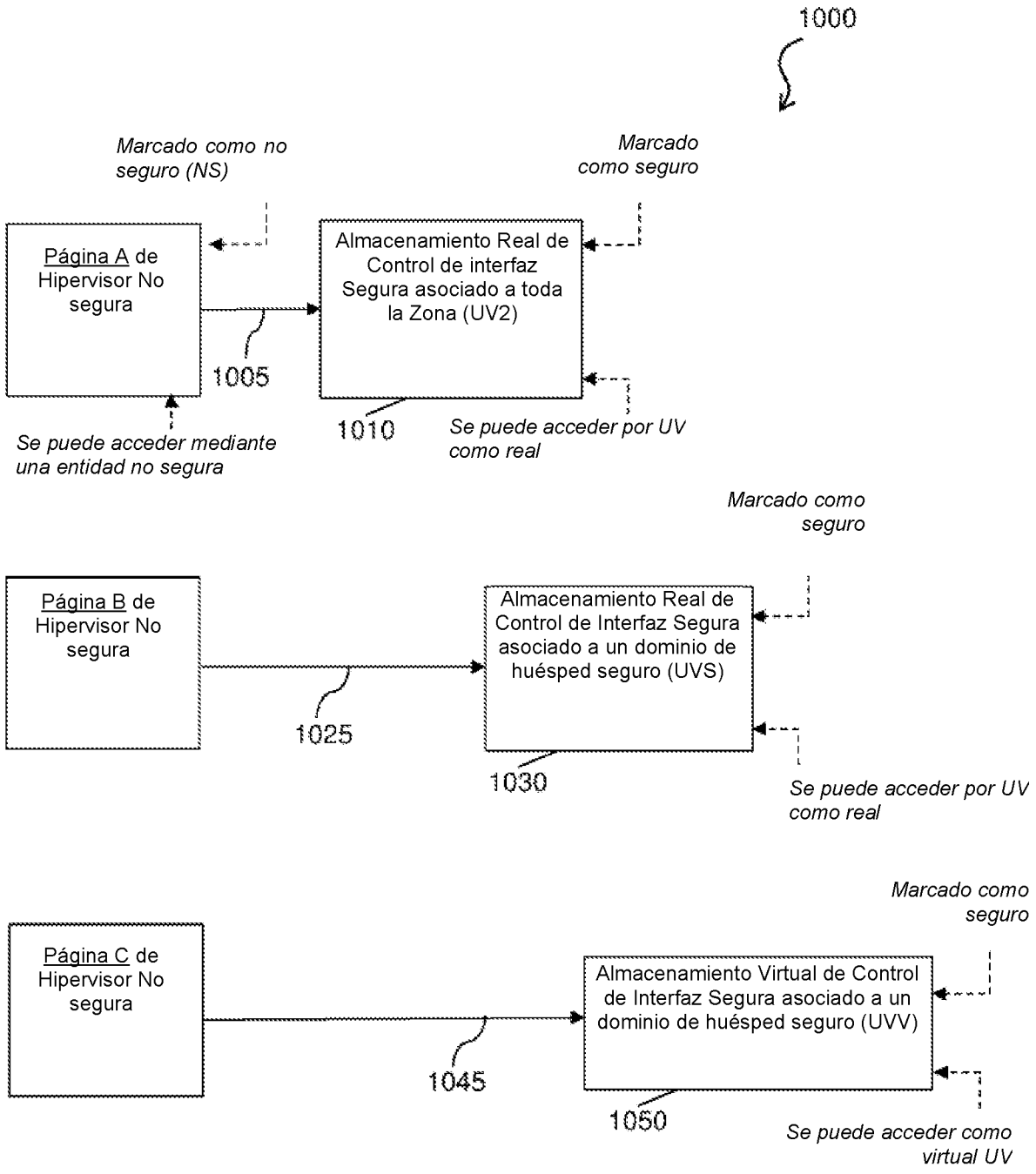
1000
Marcado como no seguro (NS)
Marcado como seguro
Página A de Hipervisor No segura
Almacenamiento Real de Control de interfaz Segura asociado a toda la Zona (UV2)
1005
1010
Se puede acceder mediante una entidad no segura
Se puede acceder por UV como real
Marcado como seguro
Página B de Hipervisor No segura
Almacenamiento Real de Control de Interfaz Segura asociado a un dominio de huésped seguro (UVS)
1025
1030
Se puede acceder por UV como real
Marcado como seguro
Página C de Hipervisor No segura
Almacenamiento Virtual de Control de Interfaz Segura asociado a un dominio de huésped seguro (UVV)
1045
1050
Se puede acceder como virtual UV

FIG. 10

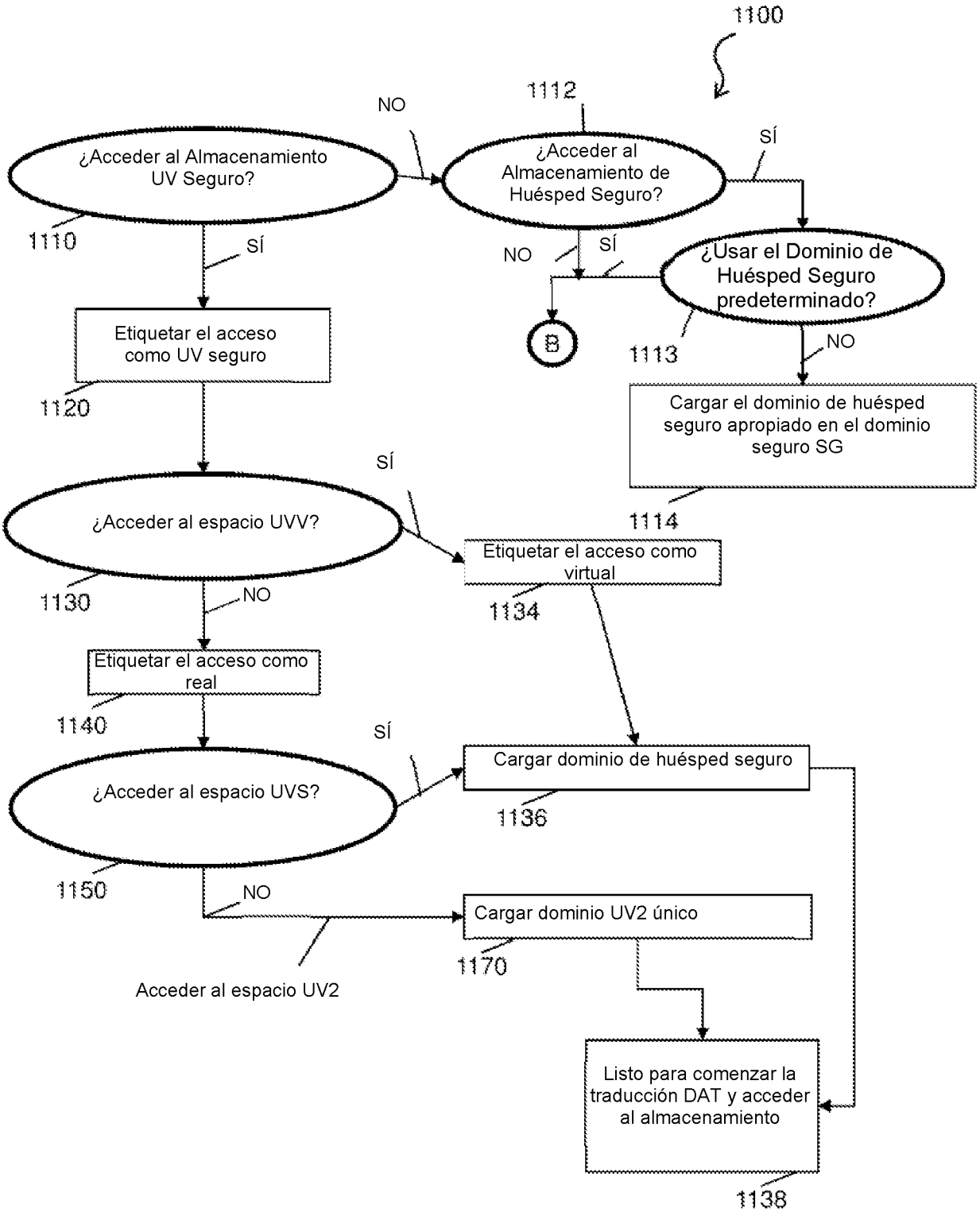
1100
1112
NO
¿Acceder al Almacenamiento UV Seguro?
¿Acceder al Almacenamiento de Huésped Seguro?
SÍ
1110
SÍ
NO
SÍ
¿Usar el Dominio de Huésped Seguro predeterminado?
Etiquetar el acceso como UV seguro
B
1113
NO
1120
Cargar el dominio de huésped seguro apropiado en el dominio seguro SG
SÍ
1114
¿Acceder al espacio UVV?
Etiquetar el acceso como virtual
1130
NO
1134
Etiquetar el acceso como real
1140
SÍ
Cargar dominio de huésped seguro
¿Acceder al espacio UVS?
1136
1150
NO
Cargar dominio UV2 único
1170
Acceder al espacio UV2
Listo para comenzar la traducción DAT y acceder al almacenamiento
1138

FIG. 11

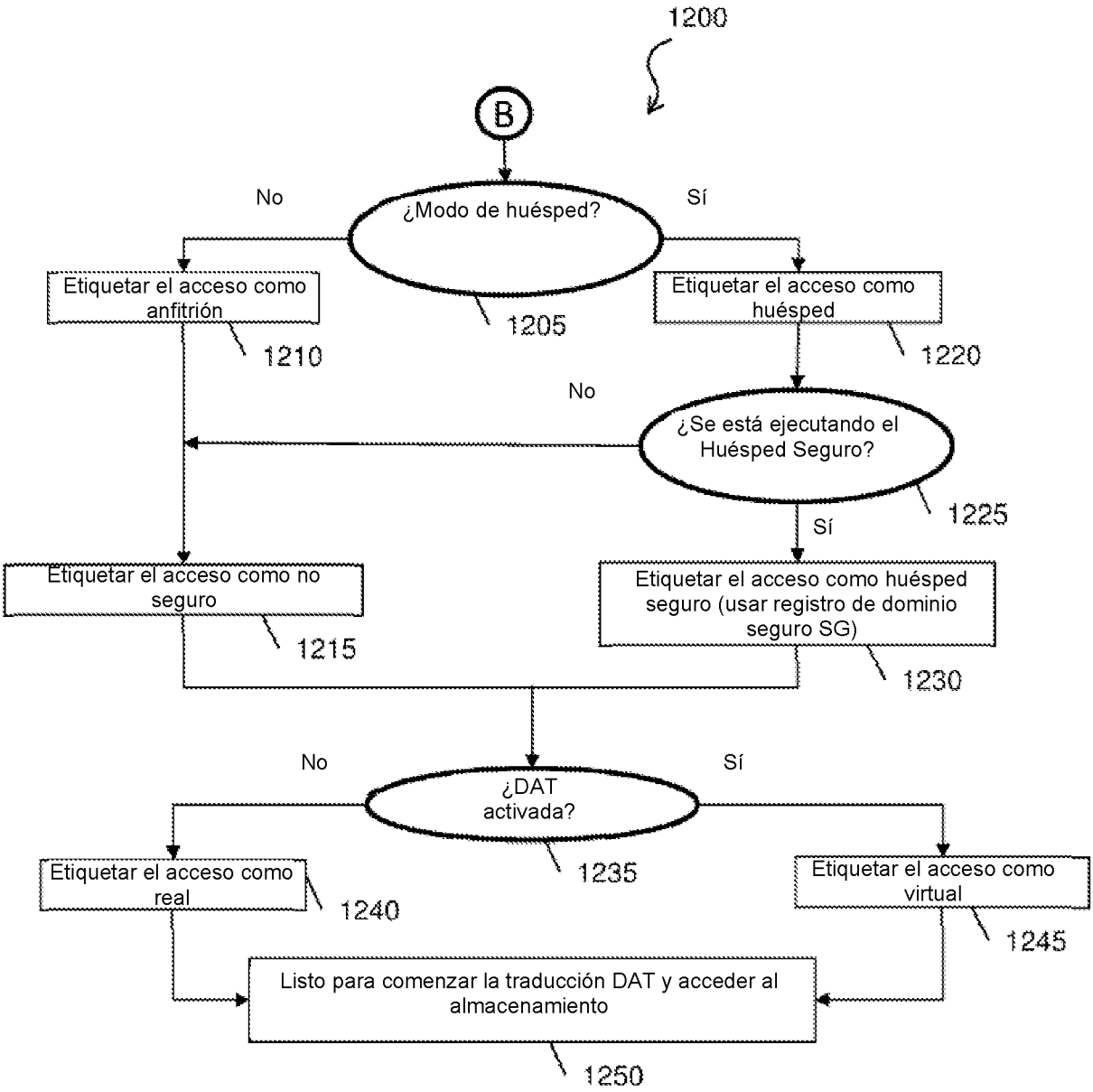
1200
B
¿Modo de huésped?
1205
No
Sí
Etiquetar el acceso como anfitrión
1210
Etiquetar el acceso como huésped
1220
No
¿Se está ejecutando el Huésped Seguro?
1225
Sí
Etiquetar el acceso como no seguro
1215
Etiquetar el acceso como huésped seguro (usar registro de dominio seguro SG)
1230
No
¿DAT activada?
1235
Sí
Etiquetar el acceso como real
1240
Etiquetar el acceso como virtual
1245
Listo para comenzar la traducción DAT y acceder al almacenamiento
1250

FIG. 12

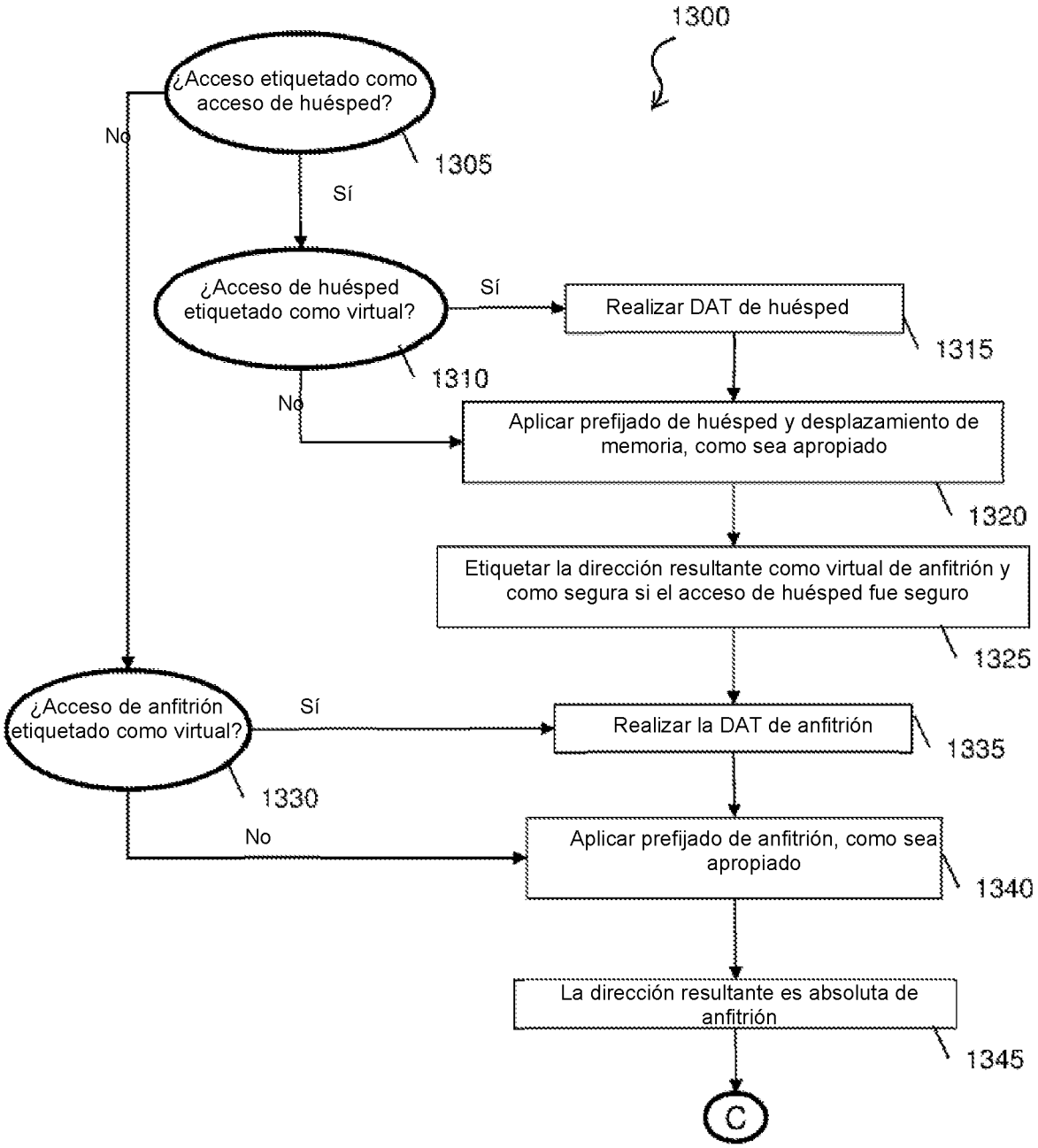
1300
¿Acceso etiquetado como acceso de huésped?
1305
No
Sí
¿Acceso de huésped etiquetado como virtual?
Sí
Realizar DAT de huésped
1315
1310
No
Aplicar prefijado de huésped y desplazamiento de memoria, como sea apropiado
1320
Etiquetar la dirección resultante como virtual de anfitrión y como segura si el acceso de huésped fue seguro
1325
¿Acceso de anfitrión etiquetado como virtual?
Sí
Realizar la DAT de anfitrión
1335
1330
No
Aplicar prefijado de anfitrión, como sea apropiado
1340
La dirección resultante es absoluta de anfitrión
1345
C

FIG. 13

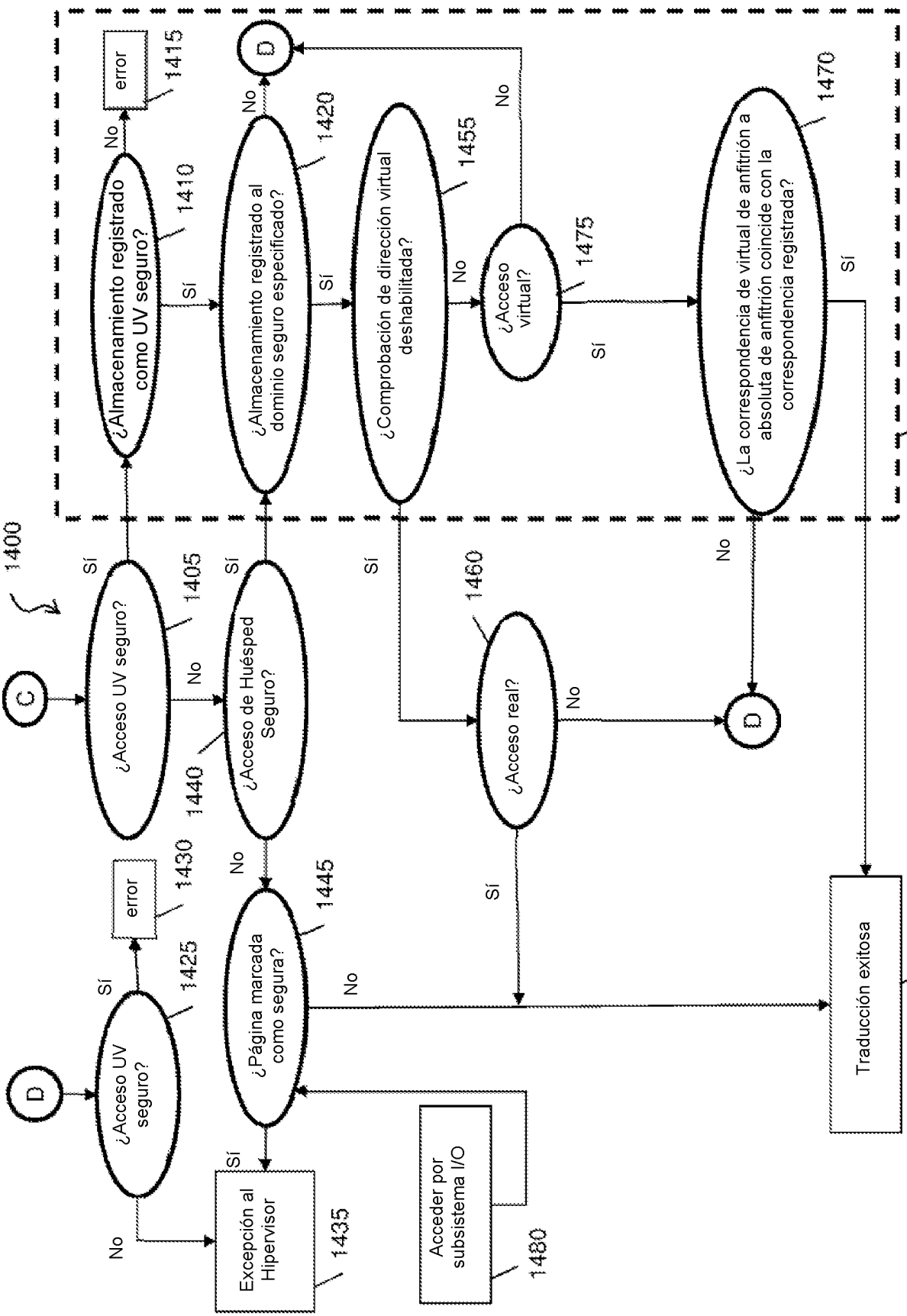
1400
C
D
¿Acceso UV seguro?
1405
Sí
No
error
1415
¿Almacenamiento registrado como UV seguro?
1410
¿Almacenamiento registrado al dominio seguro especificado?
1420
¿Comprobación de dirección virtual deshabilitada?
1455
¿Acceso virtual?
1475
¿La correspondencia de virtual de anfitrión a absoluta de anfitrión coincide con la correspondencia registrada?
1470
1485
¿Acceso de Huésped Seguro?
1440
¿Acceso real?
1460
¿Acceso UV seguro?
1425
error
1430
¿Página marcada como segura?
1445
Excepción al Hipervisor
1435
Acceder por subsistema I/O
1480
Traducción exitosa
1450

FIG. 14

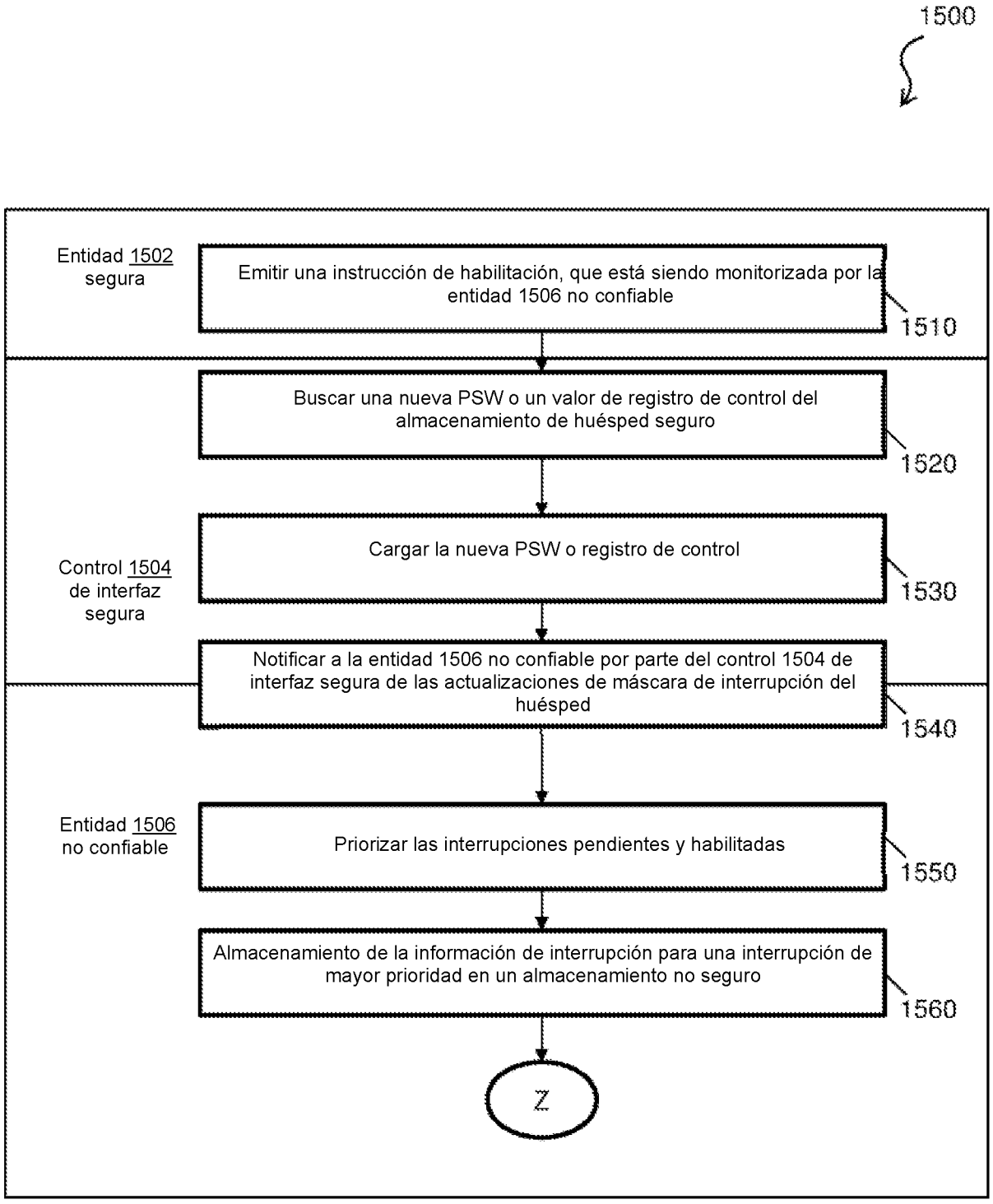
1500
Entidad 1502 segura
Emitir una instrucción de habilitación, que está siendo monitorizada por la entidad 1506 no confiable
1510
Buscar una nueva PSW o un valor de registro de control del almacenamiento de huésped seguro
1520
Control 1504 de interfaz segura
Cargar la nueva PSW o registro de control
1530
Notificar a la entidad 1506 no confiable por parte del control 1504 de interfaz segura de las actualizaciones de máscara de interrupción del huésped
1540
Entidad 1506 no confiable
Priorizar las interrupciones pendientes y habilitadas
1550
Almacenamiento de la información de interrupción para una interrupción de mayor prioridad en un almacenamiento no seguro
1560
Z

FIG. 15

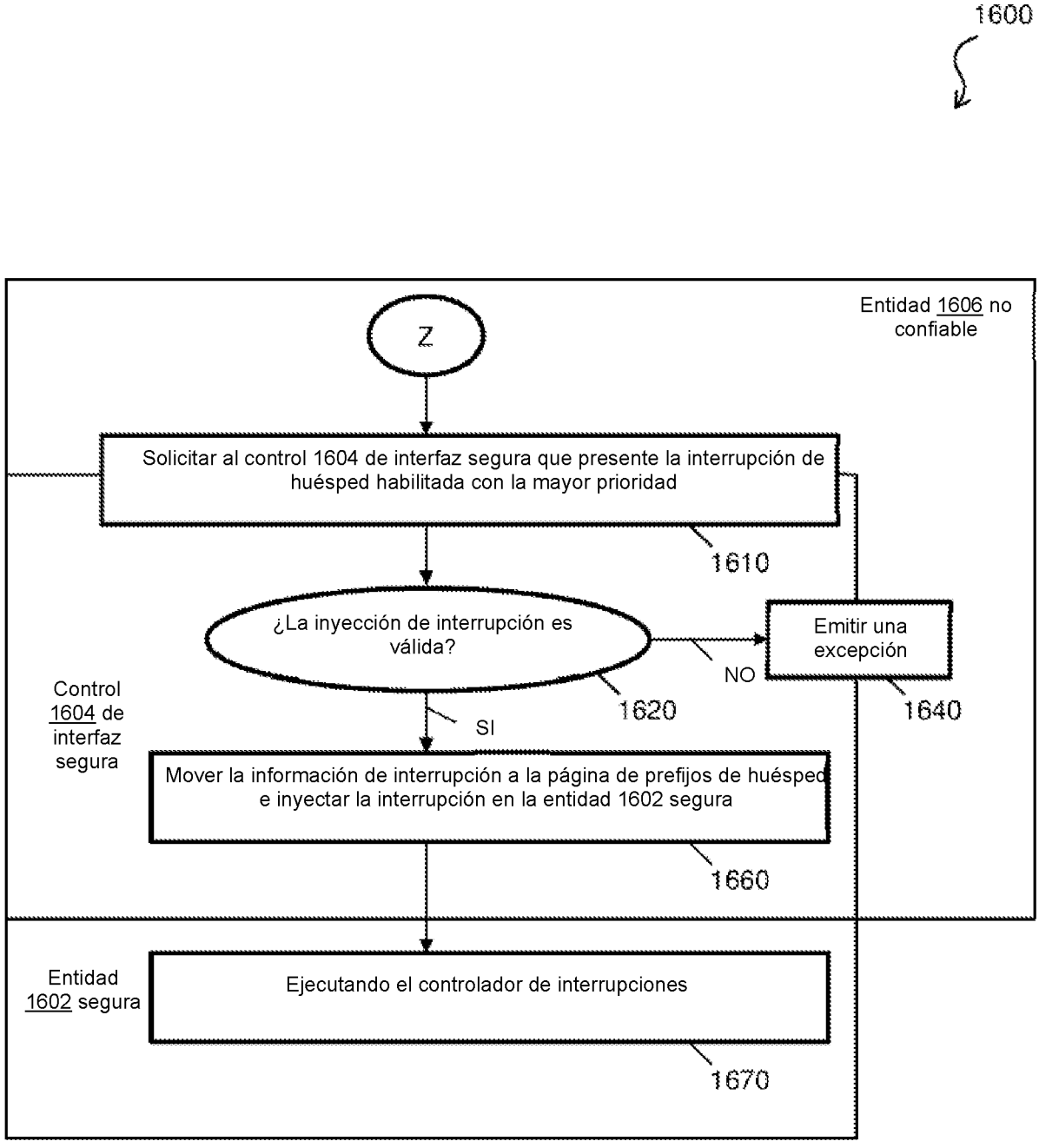
1600
Entidad 1606 no confiable
Z
Solicitar al control 1604 de interfaz segura que presente la interrupción de huésped habilitada con la mayor prioridad
1610
¿La inyección de interrupción es válida?
1620
NO
Emitir una excepción
1640
SI
Control 1604 de interfaz segura
Mover la información de interrupción a la página de prefijos de huésped e inyectar la interrupción en la entidad 1602 segura
1660
Entidad 1602 segura
Ejecutando el controlador de interrupciones
1670

FIG. 16

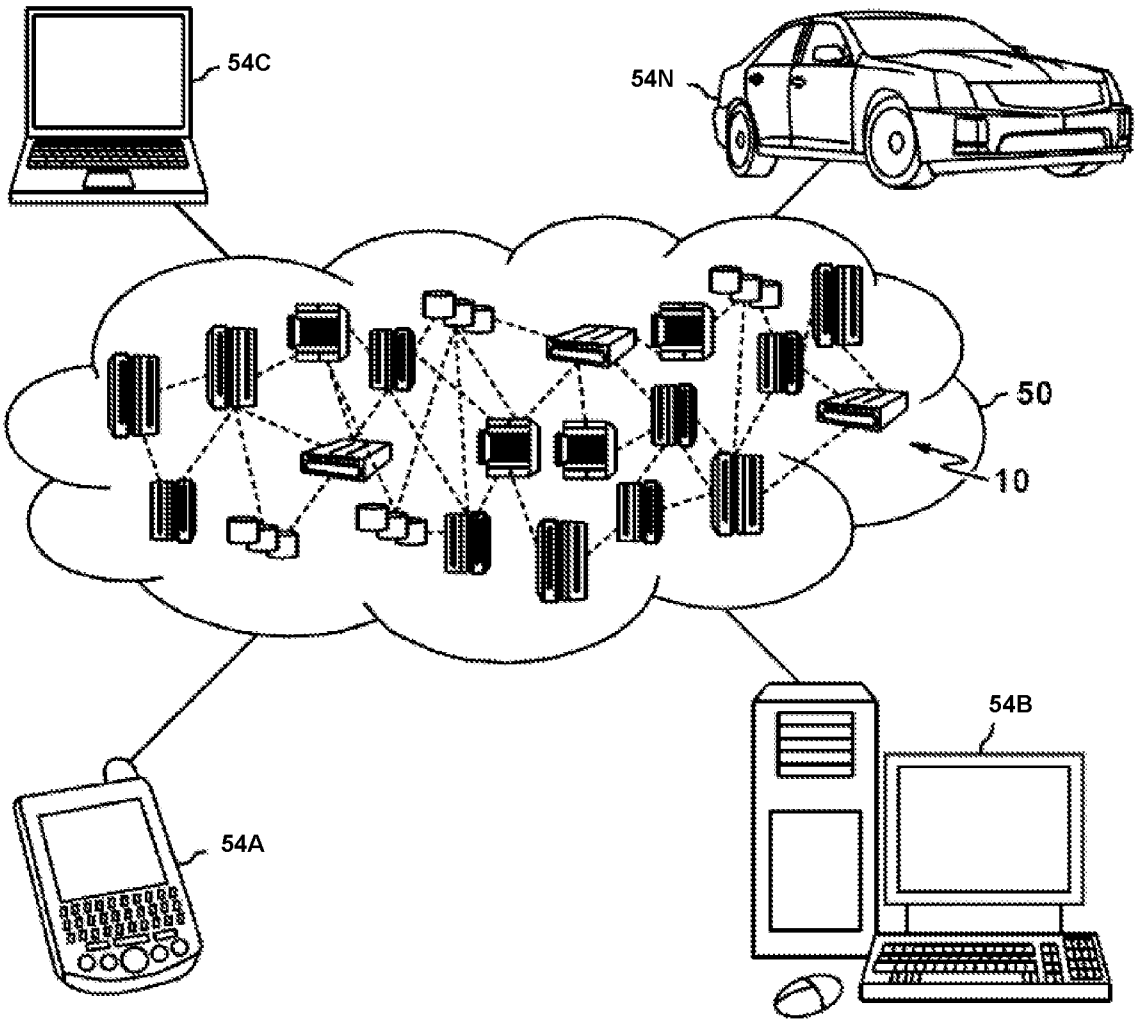

FIG. 17

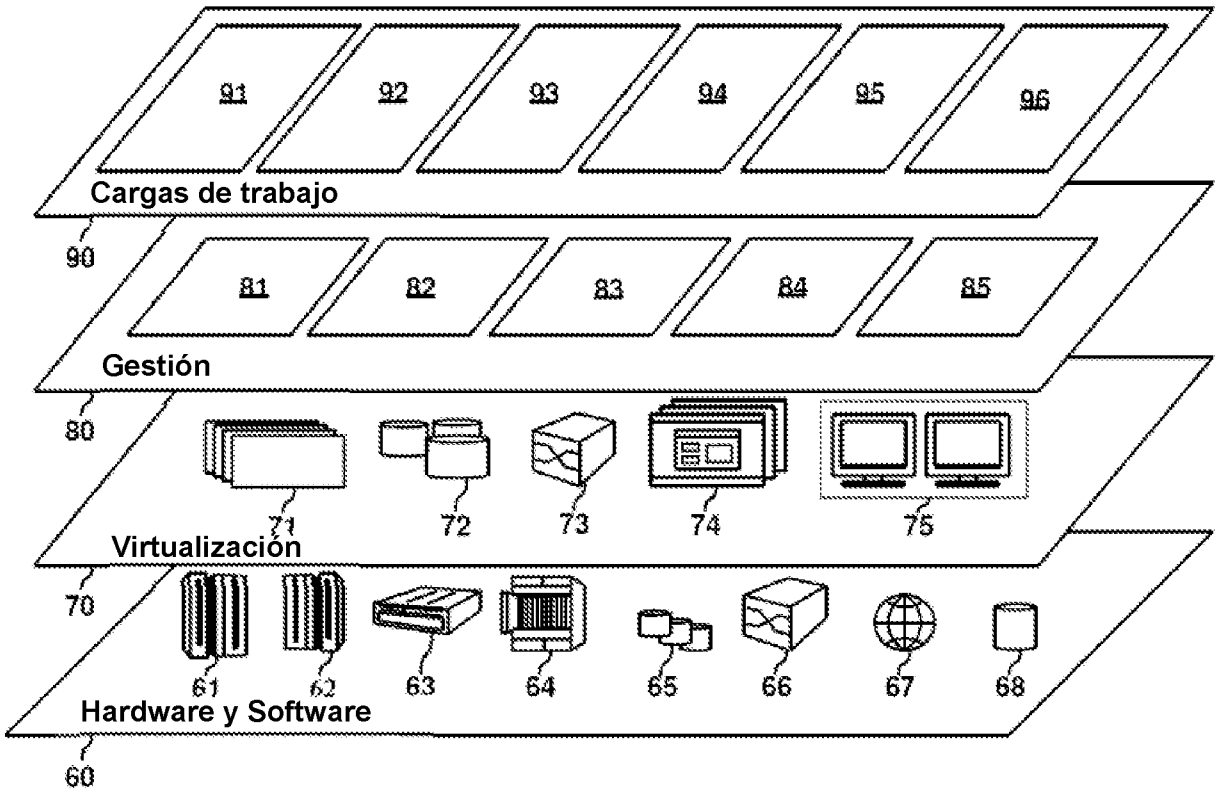
91
92
93
94
95
96
Cargas de trabajo
90
81
82
83
84
85
Gestión
80
71
72
73
74
75
Virtualización
70
61
62
63
64
65
66
67
68
Hardware y Software
60

FIG. 18

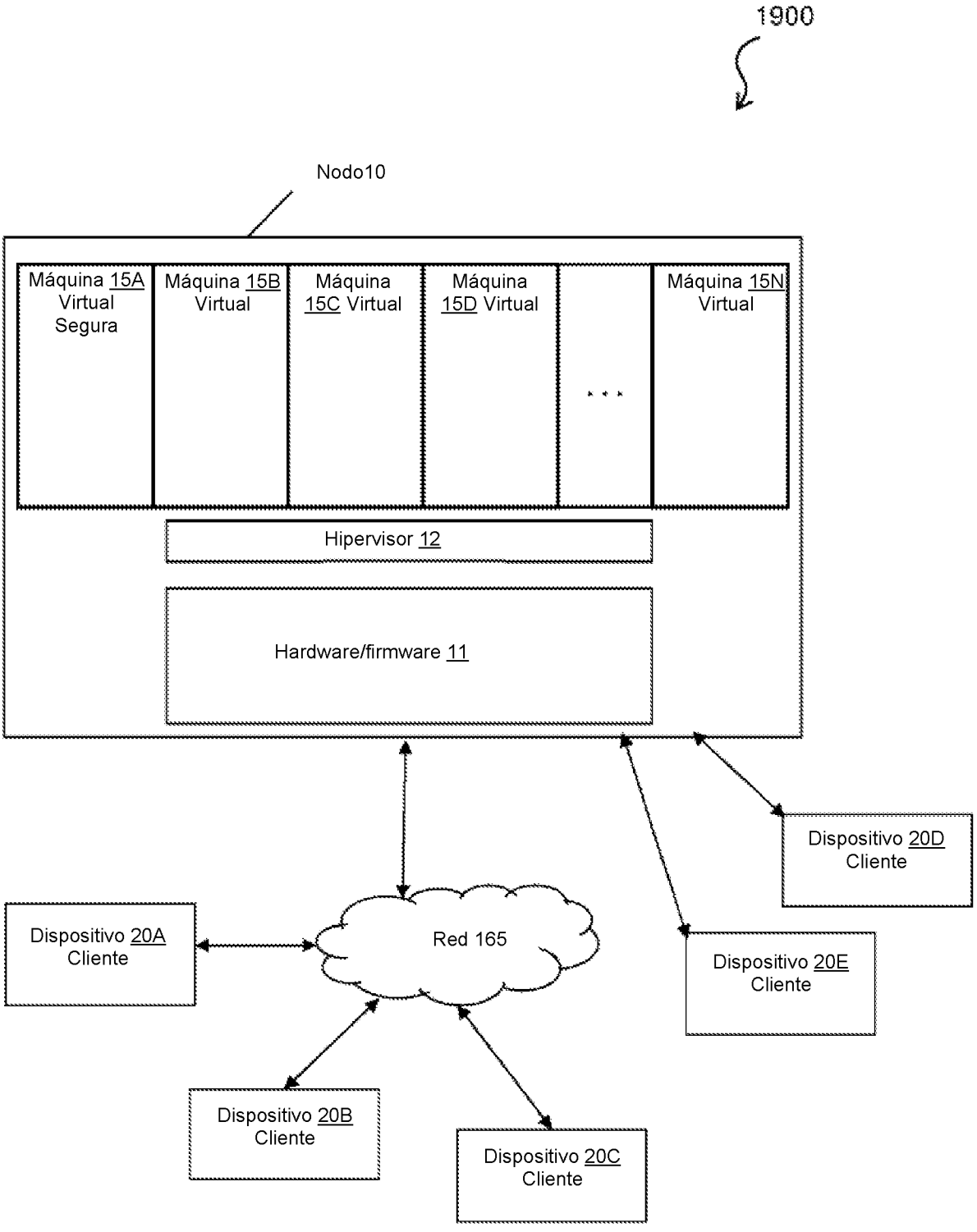
1900
Nodo10
Máquina 15A Virtual Segura
Máquina 15B Virtual
Máquina 15C Virtual
Máquina 15D Virtual
Máquina 15N Virtual
Hipervisor 12
Hardware/firmware 11
Dispositivo 20A Cliente
Red 165
Dispositivo 20D Cliente
Dispositivo 20E Cliente
Dispositivo 20B Cliente
Dispositivo 20C Cliente

FIG. 19

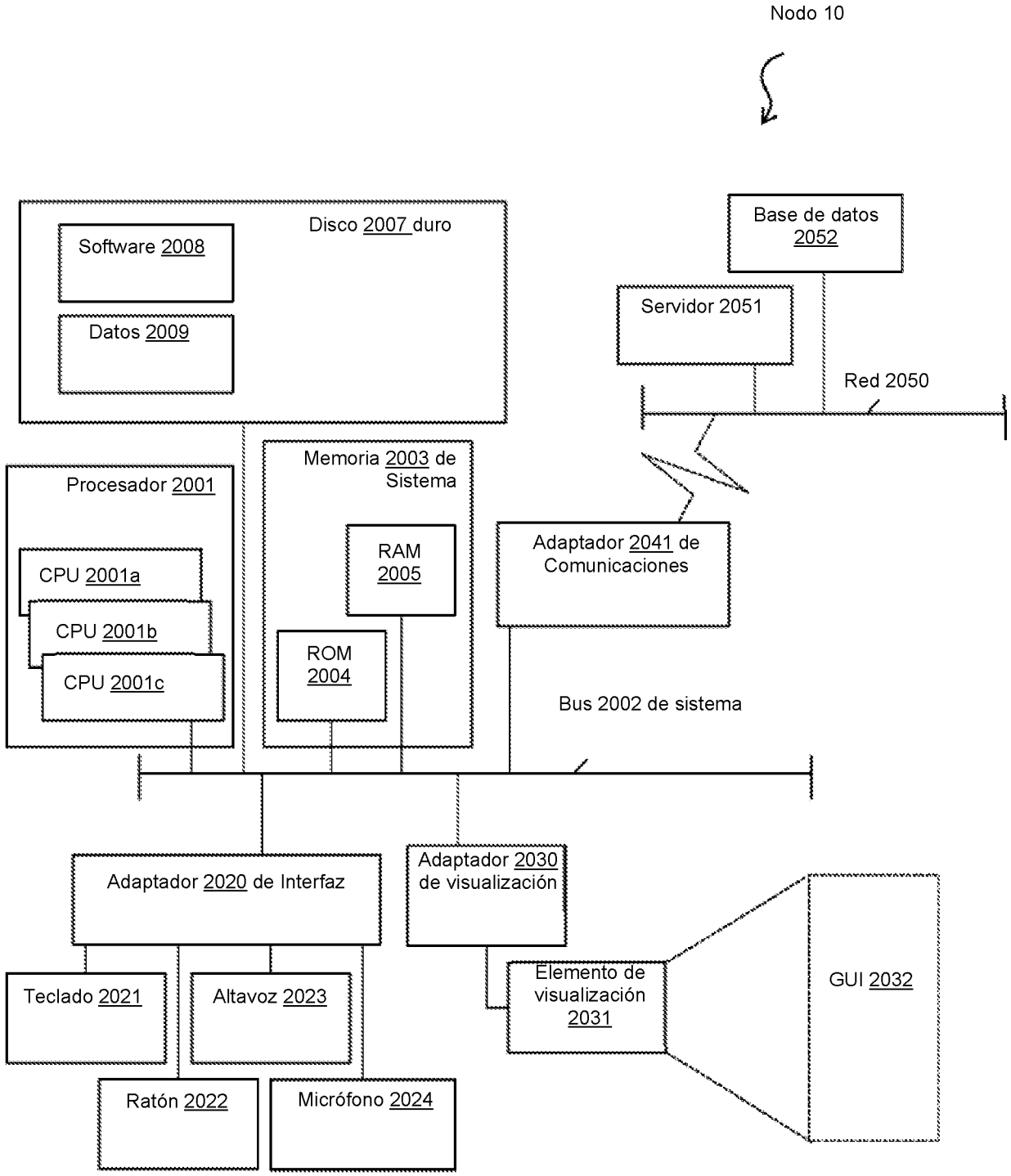
Nodo 10
Disco 2007 duro
Software 2008
Datos 2009
Base de datos 2052
Servidor 2051
Red 2050
Memoria 2003 de Sistema
Procesador 2001
RAM 2005
CPU 2001a
CPU 2001b
CPU 2001c
ROM 2004
Adaptador 2041 de Comunicaciones
Bus 2002 de sistema
Adaptador 2020 de Interfaz
Adaptador 2030 de visualización
Teclado 2021
Altavoz 2023
Elemento de visualización 2031
GUI 2032
Ratón 2022
Micrófono 2024

FIG. 20