(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 762 988**

(51) Int. CI.:

G06F 21/56 (2013.01)
G06F 9/455 (2008.01)
G06F 9/54 (2006.01)
H04L 29/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **19.12.2016 PCT/EP2016/081697**

(87) Fecha y número de publicación internacional: **22.06.2017 WO17103254**

(96) Fecha de presentación y número de la solicitud europea: **19.12.2016 E 16825369 (8)**

(97) Fecha y número de publicación de la concesión europea: **02.10.2019 EP 3391274**

(54) Título: **Autoanálisis de memoria dual para asegurar múltiples puntos finales de red**

(30) Prioridad:

19.12.2015 US 201562269952 P

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
26.05.2020

(73) Titular/es:

BITDEFENDER IPR MANAGEMENT LTD. (100.0%)
Kreontos 12
1076 Nicosia , CY

(72) Inventor/es:

LUTAS, DAN-HOREA;
LUKACS, SANDOR;
TICLE, DANIEL-IOAN;
CIOCAS, RADU-IOAN y
ANICHITEI, IONEL-CRISTINEL

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Autoanálisis de memoria dual para asegurar múltiples puntos finales de red

Solicitudes relacionadas

Esta solicitud reivindica los beneficios de la fecha de cumplimentación de la solicitud de patente provisional de los EE.UU. Nº 62/269,952, archivada el 19 de diciembre de 2015, titulada “Autoanálisis de Memoria Dual para Asegurar Múltiples Puntos Finales de Red”.

Antecedentes

La invención se relaciona con los sistemas y métodos de seguridad informática, y en concreto con los sistemas y métodos para proteger los entornos de virtualización de hardware de amenazas de seguridad informática tales como el software malicioso y la intrusión.

El software malicioso, también conocido como malware, afecta a un gran número de sistemas informáticos alrededor del mundo. En sus distintas formas, tales como los virus, los gusanos, los encubridores, y el spyware, el malware presenta un serio riesgo a millones de usuarios informáticos, que se hacen vulnerables a la pérdida de datos e información sensible, el robo de identidad, y la pérdida de productividad, entre otros.

El software de seguridad informática se puede usar para proteger los sistemas informáticos del software y la intrusión maliciosa. Sin embargo, en los sistemas informáticos distribuidos tales como las redes corporativas y los sistemas informáticos en la nube, el software de seguridad convencional normalmente no responde bien a los ataques. Incluso cuando el software de seguridad es capaz de detectar un ataque, el análisis y la remediación puede aún requerir que un operador humano sea enviado al sistema cliente afectado, por ejemplo para aplicar un parche, recuperar los datos perdidos, etc. Además, una vez que se detecta y analiza una nueva amenaza, las versiones actualizadas del software de seguridad se pueden distribuir inmediatamente a todos los sistemas informáticos protegidos.

Un sistema de seguridad informática alternativo puede ejecutarse en un servidor central, que recibe datos relevantes de los clientes de seguridad sobre una red de comunicación. El servidor puede determinar según los datos recibidos si el respectivo cliente está infectado con malware, y puede comunicar un veredicto al respectivo cliente. Aunque dichas configuraciones están mejor equipadas para negociar con las amenazas emergentes, requieren una sustancial potencia de cálculo en el lado del servidor.

Las operaciones de seguridad informática se complicaron más debido a la llegada de la virtualización de hardware. Según se negocian en línea más y más bienes y servicios, y según el trabajo resulta progresivamente deslocalizado, la infraestructura como servicio (IAAS) ha resultado una alternativa viable a la adquisición de hardware informático. Una proporción sustancial de las actividades informáticas se llevan a cabo actualmente usando máquinas virtuales. En aplicaciones típicas, tales como las torres de servidores y la computación en la nube, cientos de máquinas virtuales pueden ejecutarse de manera concurrente en una plataforma de hardware única. Todas dichas máquinas virtuales pueden requerir protección contra malware y/o intrusión. Un ejemplo de solución de seguridad informática para una plataforma de virtualización de hardware se describe en la publicación previa a concesión de los EE.UU. nº 2011/ 0004935 A1, por M. Moffie et al, titulada “SISTEMA DE DETECCIÓN DE INTRUSIÓN BASADA EN VMM”, en donde los eventos detectados durante la operación de una máquina virtual se consolidan en características que se comparan con las características de un sistema operativo conocido normal. Las diferencias sustanciales entre las características recopiladas y las características normales pueden indicar una máquina virtual comprometida.

Adaptar a la naturaleza siempre cambiante del software malicioso y a los retos de una mano de obra móvil requiere el despliegue de sistemas y protocolos de seguridad informática innovadores, y especialmente de sistemas y métodos que permitan una gestión eficiente de las operaciones de seguridad informática a lo largo de múltiples clientes distribuidos.

Compendio

Según un aspecto, un sistema informático cliente comprende un procesador de hardware configurado para ejecutar un hipervisor, un motor de autoanálisis en vivo, y un motor de autoanálisis bajo demanda. El hipervisor se configura para exponer una máquina virtual (VM) invitada y una VM de seguridad distinta de la VM invitada, en donde el motor de autoanálisis bajo demanda se ejecuta dentro de la VM de seguridad, y en donde el motor de autoanálisis en vivo se ejecuta fuera de las VM invitada y de seguridad. El motor de autoanálisis en vivo se configura, en respuesta a la detección de una ocurrencia de un evento dentro de la VM invitada, para transmitir un indicador de evento a un sistema informático servidor remoto a través de una red de comunicación. El motor de autoanálisis bajo demanda se configura, en respuesta al motor de autoanálisis en vivo que transmite el indicador del evento al sistema informático servidor remoto, para recibir una solicitud de análisis desde el sistema informático servidor remoto, indicando la solicitud de análisis una herramienta de seguridad que reside en un repositorio de herramientas remotas configurado para distribuir herramientas de seguridad a una pluralidad de clientes que incluyen al sistema informático cliente, comprendiendo la herramienta de seguridad software configurado para analizar la ocurrencia del evento, siendo la

herramienta de seguridad seleccionada por el sistema informático servidor remoto según un tipo de evento del evento. El motor de autoanálisis bajo demanda se configura además, en respuesta a la recepción de la solicitud de análisis, para identificar la herramienta de seguridad desde el repositorio de herramientas, en donde la recuperación de la herramienta de seguridad comprende conectar al repositorio de herramientas central a través de la red de comunicación. El motor de autoanálisis bajo demanda se configura además, en respuesta a la recuperación selectiva de la herramienta de seguridad, para ejecutar la herramienta de seguridad y para transmitir un resultado de la ejecución de la herramienta de seguridad al sistema informático servidor remoto.

Según otro aspecto, se configura un ordenador servidor para realizar las transacciones de seguridad informática con una pluralidad de sistemas cliente. El sistema informático servidor comprende un procesador de hardware configurado, en respuesta a la recepción de un indicador de evento desde un sistema cliente de la pluralidad de sistemas cliente, siendo el indicador de evento indicativo de una ocurrencia de un evento dentro de una VM invitada que se ejecuta en el sistema cliente, para seleccionar una herramienta de seguridad que reside en un repositorio de herramientas remoto configurado para distribuir herramientas de seguridad a la pluralidad de sistemas cliente, comprendiendo la herramienta de seguridad software configurada para analizar la ocurrencia del evento, en donde la selección de la herramienta de seguridad se realiza según un tipo de evento del evento. El procesador de hardware se configura además, en respuesta a la selección de la herramienta de seguridad, para transmitir una solicitud de análisis al sistema cliente a través de una red de comunicación, comprendiendo la solicitud de análisis un identificador de la herramienta de seguridad; y en respuesta, recibir desde el sistema cliente un resultado de la ejecución de la herramienta de seguridad en el sistema cliente. El sistema cliente se configura para ejecutar un hipervisor, un motor de autoanálisis en vivo, y un motor de autoanálisis bajo demanda. El hipervisor se configura para exponer la VM invitada y una VM de seguridad distinta de la VM invitada, en donde el motor de autoanálisis bajo demanda se ejecuta dentro de la VM de seguridad, y en donde el motor de autoanálisis en vivo se ejecuta fuera de las VM invitada y de seguridad. El motor de autoanálisis en vivo se configura, en respuesta a la detección de la ocurrencia del evento, para transmitir el indicador de evento al sistema informático servidor. El motor de autoanálisis bajo demanda se configura, en respuesta a la recepción de la solicitud de análisis, para identificar la herramienta de seguridad según la solicitud de análisis. El motor de autoanálisis bajo demanda se configura además, en respuesta a la identificación de la herramienta de seguridad, para recuperar de manera selectiva la herramienta de seguridad desde el repositorio de herramientas, en donde la recuperación de la herramienta de seguridad comprende que el sistema cliente se conecte al repositorio de herramientas remoto a través la red de comunicación. El motor de autoanálisis bajo demanda se configura además, en respuesta a la recuperación de la herramienta de seguridad, para ejecutar la herramienta de seguridad para producir el resultado.

Según otro aspecto, un medio legible por ordenador no transitorio comprende un conjunto de instrucciones que, al ejecutarse en un procesador de hardware de un sistema informático cliente, provocan que el sistema informático cliente cree un hipervisor, un motor de autoanálisis en vivo, y un motor de autoanálisis bajo demanda. El hipervisor se configura para exponer una máquina virtual (VM) invitada y una VM de seguridad distinta de la VM invitada, en donde el motor de autoanálisis bajo demanda se ejecuta dentro de la VM de seguridad, y en donde el motor de autoanálisis en vivo se ejecuta fuera de las VM invitada y de seguridad. El motor de autoanálisis en vivo se configura, en respuesta a la detección de una ocurrencia de un evento dentro de la VM invitada, para transmitir un indicador del evento a un sistema informático servidor remoto a través de una red de comunicación. El motor de autoanálisis bajo demanda se configura, en respuesta al motor de autoanálisis en vivo que transmite el indicador del evento al sistema informático servidor remoto, para recibir una solicitud de análisis desde el sistema informático servidor remoto, indicando la solicitud de análisis una herramienta de seguridad que reside en un repositorio de herramientas remoto configurado para distribuir las herramientas de seguridad a una pluralidad de clientes incluyendo el sistema informático cliente, comprendiendo la herramienta de seguridad software configurado para analizar la ocurrencia del evento, siendo la herramienta de seguridad seleccionada por el sistema informático servidor remoto según un tipo de evento del evento. El motor de autoanálisis bajo demanda se configura además, en respuesta a la recepción de la solicitud de análisis, identificar la herramienta de seguridad según la solicitud de análisis, y en respuesta, recuperar de manera selectiva la herramienta de seguridad del repositorio de herramientas, en donde la recuperación de la herramienta de seguridad comprende conectarse al repositorio de herramientas central a través de la red de comunicación. El motor de autoanálisis bajo demanda se configura además, en respuesta a la recuperación de manera selectiva de la herramienta de seguridad, para ejecutar la herramienta de seguridad y para transmitir un resultado de la ejecución de la herramienta de seguridad al sistema informático servidor remoto.

Breve descripción de los dibujos

Los aspectos y ventajas anteriores de la presente invención resultarán mejor entendidos tras la lectura de la siguiente descripción detallada y tras la referencia a los dibujos donde:

La Figura 1 ilustra una configuración ejemplar en donde se protegen los múltiples sistemas cliente contra amenazas de seguridad informática según algunas realizaciones de la presente invención.

La Figura 2-A ilustra una configuración de hardware ejemplar de un sistema cliente según algunas realizaciones de la presente invención.

La Figura 2-B ilustra una configuración de hardware ejemplar de un sistema informático servidor de seguridad según algunas realizaciones de la presente invención.

La Figura 3-A muestra un conjunto ejemplar de máquinas virtuales expuestas por un hipervisor que se ejecutan en un sistema cliente protegido, y un par ejemplar de motores de autoanálisis según algunas realizaciones de la presente invención.

La Figura 3-B muestra una configuración alternativa de los componentes de seguridad según algunas realizaciones de la presente invención.

La Figura 4 muestra una secuencia ejemplar de pasos llevada a cabo por una aplicación de instalación para configurar la seguridad informática en un sistema cliente según algunas realizaciones de la presente invención.

La Figura 5 muestra la configuración de una conexión segura de red privada virtual (VPN) entre un sistema cliente y el servidor de seguridad según algunas realizaciones de la presente invención.

La Figura 6 muestra un intercambio de datos ejemplar entre un sistema cliente y el servidor de seguridad, el intercambio se produce durante la detección de malware según algunas realizaciones de la presente invención.

La Figura 7 muestra una secuencia ejemplar de pasos realizada por el motor de autoanálisis en vivo según algunas realizaciones de la presente invención.

La Figura 8 muestra una secuencia ejemplar de pasos realizada por el motor de autoanálisis bajo demanda según algunas realizaciones de la presente invención.

La Figura 9 ilustra una secuencia ejemplar de pasos realizada por el servidor de seguridad según algunas realizaciones de la presente invención.

Descripción detallada de las realizaciones preferidas

En la siguiente descripción, se entiende que todas las conexiones recitadas entre las estructuras pueden ser conexiones operativas directas o conexiones operativas indirectas a través de estructuras intermedias. Un conjunto de elementos incluye uno o más elementos. Cualquier recitación de un elemento se entiende que hace referencia a al menos un elemento. Una pluralidad de elementos incluye al menos dos elementos. A menos que se requiera lo contrario, los pasos descritos de cualquier método no necesitan ser necesariamente realizados en un orden ilustrado concreto. Un primer elemento (por ejemplo, los datos) obtenido a partir de un segundo elemento abarca un primer elemento igual al segundo elemento, así como un primer elemento generado procesando el segundo elemento y opcionalmente otros datos. Tomar una determinación o una decisión según un parámetro implica tomar la determinación o la decisión según el parámetro y de manera opcional según los otros datos. A menos que se especifique lo contrario un indicador de alguna cantidad/datos puede ser la propia cantidad/datos en sí, o un indicador diferente de la cantidad/datos en sí. La seguridad informática abarca la protección de usuarios y equipos contra un acceso no deseado o no autorizado a los datos y/o el hardware, una modificación o una modificación no deseada o no autorizada a los datos y/o hardware, y la destrucción de datos y/o hardware. Un programa informático es una secuencia de instrucciones de procesador que llevan a cabo una tarea. Los programas informáticos descritos en algunas realizaciones de la presente invención pueden ser entidades o sub entidades de software autónomas (por ejemplo, subrutinas, librerías) de otros programas informáticos. A menos que se especifique lo contrario, el software invitado se ejecuta dentro de una máquina virtual. Se dice que un programa se ejecuta dentro de una máquina virtual cuando se ejecuta en un procesador virtual de la máquina virtual respectiva. A menos que se especifique lo contrario, una página representa la menor unidad de memoria virtual que se puede hacer corresponder de manera individual a una memoria física de un sistema anfitrión. A menos que se especifique lo contrario, una imagen de un sistema cliente comprende una copia de un contenido de una sección de memoria usada por el respectivo sistema cliente. Los medios legibles por ordenador abarcan medios no transitorios tales como los medios de almacenamiento magnético, óptico, y semiconductor (por ejemplo, discos duros, discos ópticos, memoria flash, DRAM), así como enlaces de comunicación tales como los cables conductivos y los enlaces de fibra óptica. Según algunas realizaciones, la presente invención proporcionar, entre otros, sistemas informáticos que comprenden hardware (por ejemplo, uno o más microprocesadores) programado para realizar los métodos descritos en la presente memoria, así como medio legibles por ordenador que codifican instrucciones para realizar los métodos descritos en la presente memoria.

La siguiente descripción ilustra las realizaciones de la invención a modo de ejemplo y no necesariamente a modo de limitación.

La Figura 1 muestra una configuración ejemplar para proteger una pluralidad de sistemas 12a-d cliente contra amenazas de seguridad informática según algunas realizaciones de la presente invención. Los sistemas 12a-d cliente ejemplares incluyen sistemas informáticos personales, plataformas informáticas móviles (ordenadores portátiles, tabletas, teléfonos móviles), dispositivos de entretenimiento (TV, video consolas), dispositivos portátiles (relojes inteligentes, bandas de fitness), electrodomésticos, y cualquier otro dispositivo electrónico que comprenda un procesador y una memoria y sea capaz de operar una plataforma de virtualización de hardware. Otra categoría

ejemplar de sistemas cliente incluye los servidores de centros de datos y las plataformas de virtualización de hardware que ejecutan aplicaciones basadas en la nube tales como los servidores web y/o la infraestructura de escritorio virtual.

Los sistemas 12a-d de cliente se interconectan a través de una red 11 de comunicación, tal como una red de una casa, una red corporativa, Internet etc. La red 11 incluye al menos un conmutador y/o un enrutador. Partes de la red 11 pueden incluir una red de área local (LAN) y/o una red de telecomunicación (por ejemplo, una red de telefonía móvil 4G, una LAN inalámbrica).

En algunas realizaciones, un servidor 14 de seguridad se acopla de manera comunicativa a los sistemas 12a-d de cliente a través de la red 11 y colabora con los sistemas 12a-d de cliente para rechazar las amenazas de seguridad informática tal como se describe en detalle a continuación. El servidor 14 describe de manera general un conjunto de sistemas informáticos interconectados, que pueden estar o no en la proximidad física los unos con los otros. En algunas realizaciones, el servidor 14 se configura para recibir las notificaciones de evento de los sistemas 12a-d cliente, y en respuesta, seleccionar según un tipo de evento un tipo de análisis forense, un protocolo de mitigación de amenaza, y/o una herramienta de limpieza a ser usada por el respectivo sistema cliente. Los análisis forenses ejemplares incluyen, por ejemplo, la obtención de datos específicos sobre la causa y/o el contexto del respectivo evento. Los protocolos de mitigación de amenaza se pueden seleccionar según un tipo de software malicioso indicado por el evento respectivo, y pueden incluir la descarga y/o la ejecución de un código de limpieza y/o de control de daños en el respectivo cliente.

En algunas realizaciones, el servidor 14 de seguridad se configura además para interactuar con una base de datos 17 cliente. En una base de datos 17 cliente ejemplar, cada entrada se asocia con un sistema 12a-d cliente protegido y/o con una máquina virtual que se ejecuta en el respectivo sistema cliente protegido, y puede incluir un registro de eventos desencadenantes y/o reportes forenses (véase más adelante) reportados por el respectivo sistema cliente/máquina virtual. Una entrada ejemplar de la base de datos 17 puede comprender además los datos de perfil de sistema (por ejemplo, incluyendo la versión de OS, las aplicaciones instaladas, los diversos ajustes, el propietario, la información de contacto, etc.) para el respectivo sistema cliente/máquina virtual. Otra entrada ejemplar de la base de datos 17 cliente puede comprender un conjunto de valores de parámetros que representan una política de seguridad específica de cliente asociada con el respectivo sistema cliente. Dichos ajustes pueden ser especificados por un operador humano, o se pueden establecer de manera automática según un conjunto de reglas. En algunas realizaciones de la presente invención, las políticas específicas de cliente y/o los ajustes de seguridad varían de manera dinámica en respuesta a los eventos que se producen en el respectivo cliente, o en otros clientes protegidos.

En algunas realizaciones, los sistemas 12a-d de cliente se conectan además a un repositorio 15 de herramientas central a través de la red 11. El repositorio 15 de herramientas puede comprender un medio legible por ordenador o herramientas y recursos de seguridad de almacenamiento en máquina física en forma de código (programas informáticos) y datos. Los sistemas 12a-d cliente pueden conectarse al repositorio 15 para recuperar de manera selectiva herramientas y datos según las instrucciones recibidas desde el servidor 14 de seguridad, como se muestra en detalle más adelante. El repositorio 15 de herramientas está disponible para múltiples clientes, por lo que en una realización disponible de la presente invención, el repositorio 15 no reside en ningún sistema cliente concreto. Conectar al repositorio 15 por lo tanto comprende la transmisión y/o la recepción de las comunicaciones hacia/desde el repositorio 15 a través de un adaptador de red del sistema cliente respectivo. Dichas comunicaciones pueden atravesar un conmutador o enrutador de red en el camino.

Las herramientas de seguridad almacenadas en el repositorio 15 pueden incluir herramientas forenses, anti malware, y/o de mitigación de amenazas. Los datos del repositorio pueden comprender además los valores de parámetros para configurar o ajustar las herramientas respectivas según un tipo de evento bajo investigación, o según configuraciones de hardware/software locales. Las herramientas anti malware permiten la detección de software malicioso que se ejecuta en los sistemas 12a-d cliente, y pueden incluir una codificación de un conjunto de reglas heurísticas y/o bases de datos de firmas de identificación de malware. Las herramientas de mitigación de amenazas pueden incluir las herramientas de limpieza programadas para eliminar o al menos incapacitar un agente de software malicioso que se ejecuta en un sistema cliente. Otras herramientas de mitigación de amenazas ejemplares se programan para evitar que un sistema cliente infectado transmita software malicioso a otro sistema cliente, por ejemplo controlando la forma en la que el sistema cliente infectado usa su adaptador de red.

Las herramientas forenses permiten el análisis de eventos relacionados con la seguridad que se producen en los sistemas 12a-d cliente. Algunos ejemplos de herramientas forenses incluyen las herramientas de generación de imágenes, programadas para obtener una imagen de memoria de un sistema cliente o de una máquina virtual que se ejecuta en el sistema cliente respectivo. La imagen puede incluir datos de memoria asociados con un sistema operativo (OS) o con otra aplicación que se ejecuta actualmente en el sistema cliente respectivo. Una imagen de un núcleo OS puede incluir, entre otras cosas, una copia del código del núcleo y secciones de datos, diversos controladores de núcleo en memoria (secciones de código y/o datos), hilos de núcleo en memoria y sus pilas correspondientes, las estructuras de datos de núcleo – tales como la lista de módulos cargados, la lista de procesos, etc. Una imagen ejemplar de una aplicación comprende una copia de una captura de memoria de la aplicación,

incluyendo su código y las secciones de datos, las pilas en memoria usadas por los hilos de la aplicación, las páginas de memoria de la aplicación respectiva, etc.

En algunas realizaciones, tomar una imagen de memoria comprende suspender la ejecución de la VM 32 invitada para permitir la copia del contenido de las respectivas secciones de memoria. Una realización alternativa realiza análisis forense de la memoria "en vivo" sin realizar capturas. En dichas realizaciones, el hipervisor 30 puede hacer la correspondencia de un conjunto de páginas de memoria física usadas por la VM 32 invitada a páginas de memoria virtual mediante la VM 33 de seguridad. La VM 33 de seguridad puede inspeccionar entonces el contenido de las respectivas páginas de memoria, por ejemplo en respuesta a un evento concreto, sin tener que suspender la ejecución de la VM 32 invitada o copiar y transferir el contenido respectivo. Un ejemplo de una herramienta forense de memoria "en vivo" es la estructura Volatility® de la Fundación Volatility.

Otro ejemplo de una herramienta forense es una herramienta de inventario de aplicaciones configurada para enumerar las entidades de software actualmente instaladas y/o que se ejecutan en un sistema cliente. Aún otro ejemplo de herramienta forense es un capturador de configuraciones programado para obtener un conjunto de ajustes de configuración (por ejemplo, los valores actuales de los diversos parámetros OS, ajustes de hardware, ajustes de seguridad, ajustes de cortafuegos, etc.). Otras herramientas forenses ejemplares se programan para capturar registros de eventos de sistema y/o aplicación, o datos de ruptura del sistema (por ejemplo los mini contenedores de ruptura de Windows®).

La Figura 2-A muestra una configuración de hardware ejemplar de un sistema 12 cliente, tal como los sistemas 12a-d en la Figura 1. Por simplicidad, el sistema cliente ilustrado es un sistema informático: la configuración de hardware de otros sistemas cliente tales como teléfonos móviles, relojes, etc., pueden diferir algo de la configuración ilustrada. El sistema 12 cliente comprende un conjunto de dispositivos físicos, incluyendo un procesador 16 de hardware y una unidad 18 de memoria. En algunas realizaciones el procesador 12 comprende un dispositivo físico (por ejemplo un microprocesador, un circuito integrado multi núcleo creado en un substrato semiconductor, etc.) configurado para ejecutar operaciones computacionales y/o lógicas con un conjunto de señales y/o datos. En algunas realizaciones, dichas operaciones se entregan al procesador 12 en la forma de una secuencia de instrucciones de procesador (por ejemplo de código máquina u otro tipo de codificación). La unidad 18 de memoria puede comprender medios de almacenamiento legibles por ordenador (por ejemplo DRAM, SRAM) que almacenan instrucciones y/o datos accedidos o generados por el procesador 16.

Dependiendo del tipo y rendimiento del dispositivo, el sistema 12 cliente puede comprender además un conjunto de dispositivos 20 de entrada, tales como un teclado, un ratón, una pantalla táctil, etc., que permiten a un usuario introducir datos y/o instrucciones al sistema 12 cliente. Un conjunto de dispositivos 22 de salida, tal como un monitor o una pantalla de cristal líquido, pueden transmitir información al usuario, por ejemplo, a través de una interfaz de usuario gráfica. Los dispositivos 24 de almacenamiento incluyen medios legibles por ordenador que permiten el almacenamiento no volátil, la lectura, y la escritura de instrucciones de procesador y/o datos. Los dispositivos 24 de almacenamiento ejemplares incluyen discos magnéticos y ópticos y dispositivos de memoria flash, así como medios extraíbles tales como discos CD y/o DVD y controladores. El conjunto de adaptadores 26 de red permite al sistema 12 cliente conectarse a la red 11 de comunicación y/o a otros dispositivos/sistemas informáticos. El concentrador 28 de controladores representa generalmente la pluralidad de buses de sistema, periféricos, y/o de chipset, y/o toda la demás circuitería que permite la comunicación entre el procesador 16 y los dispositivos 18, 20, 22, 24 y 26. Por ejemplo el concentrador 28 de controladores puede incluir una unidad de gestión de memoria (MMU), un controlador de entrada/salida (I/O), y un controlador de interrupciones, entre otros. En otro ejemplo, el concentrador 28 de controladores puede comprender un procesador 16 de conexión puente norte hacia la memoria 18 y/o un procesador 16 de conexión puente sur hacia los dispositivos 20, 22, 24 y 26. En algunas realizaciones, el concentrador 28 de controladores puede estar integrado, en parte o completamente, con el procesador 16, por ejemplo, la MMU puede compartir un sustrato semiconductor común con el procesador 16.

La Figura 2-B muestra una configuración de hardware ejemplar del servidor 14 de seguridad. El servidor 14 comprende un procesador 116 hardware, una memoria 118 de servidor, un conjunto de dispositivos 124 de almacenamiento de servidor, y un conjunto de adaptadores 126 de red, todos conectados mediante un concentrador 128 de controladores de servidor. La operación de los dispositivos 116, 118, 124, y 126 puede reflejar la de los dispositivos 16, 18, 24, y 26 descritos anteriormente. Por ejemplo, el procesador 116 servidor puede comprender un circuito integrado configurado para ejecutar las operaciones computacionales y/o lógicas con un conjunto de señales y/o datos. La memoria 118 de servidor puede comprender medios legibles por ordenador no transitorios (por ejemplo, una RAM) que almacenan datos/señales accedidos o generados por el procesador 116 mientras realiza los cálculos. Los adaptadores 126 de red permiten al servidor 14 de seguridad conectarse a la red 11 de comunicación.

En algunas realizaciones, el sistema 12 cliente se configura para exponer un conjunto de máquinas virtuales, por ejemplo tal como se ilustra en las Figura 3-A-B. Una máquina virtual (VM) emula un sistema informático/máquina física real, usando cualquiera de entre una variedad de técnicas conocidas en la técnica de la virtualización de hardware. En algunas configuraciones ejemplares, un hipervisor 30 se ejecuta en el sistema 12 cliente, el hipervisor 30 se configura para crear o permitir una pluralidad de dispositivos virtualizados, tales como un procesador virtual y una unidad de gestión de memoria virtual, y para presentar dichos dispositivos virtualizados en software, para simular los dispositivos físicos, reales del sistema 12 cliente. Dichas operaciones son comúnmente conocidas en la

técnica como exponer una máquina virtual. El hipervisor 30 puede permitir además múltiples máquinas virtuales para compartir los recursos de hardware del sistema 12 de anfitrión, de manera que cada VM pueda operar de manera independiente y no ser conscientes de las otras VM que se ejecutan de manera concurrente en el sistema 12. Ejemplos de hipervisores populares incluyen el VMware ® vSphere® de VMware Inc. Y el hipervisor Xen® de código abierto, entre otros.

En las configuraciones ejemplares ilustradas en las Figura 3-A-B, una VM 32 invitada ejecuta un sistema operativo 34 invitado y una aplicación 36. Aunque las Figura 3-A-B muestran sólo una VM invitada, en aplicaciones tales como una infraestructura de escritorio virtual (VDI) y una torre de servidores, el sistema 12 cliente puede ejecutar múltiples (por ejemplo, cientos) de dichas VM de manera concurrente. Cada VM invitada incluye al menos un procesador virtualizado, y puede incluir además otros dispositivos virtualizados tales como los dispositivos de entrada, de salida, de almacenamiento, y de red virtualizados, así como el controlador virtualizado, entre otros. Cada procesador virtualizado comprende una emulación de al menos alguna de las funcionalidades del procesador 16 de hardware, y se configura para recibir instrucciones de procesador para su ejecución. El software que usa el procesador virtual para su ejecución se dice que se ejecuta dentro de la máquina virtual respectiva. Por ejemplo, en los ejemplos de las Figura 3-A-B, el OS 34 invitado y la aplicación 36 se dice que se ejecutan dentro de la VM 32 invitada. En contraste, el hipervisor 30 se dice que se ejecuta fuera, o por debajo, de la VM 32 invitada.

El OS 34 proporciona una interfaz entre la aplicación 36 y el hardware virtualizado de la VM 32 invitada. El sistema operativo 34 puede comprender cualquier sistema operativo ampliamente disponible tal como Microsoft Windows®, MacOS®, Linux®, iOS®, o Android®, entre otros. La aplicación 36 representa generalmente cualquier programa informático tal como un procesador de textos, un procesador de imágenes, un reproductor de medios, una base de datos, un calendario, un gestor de contactos personales, un navegador, un juego, una comunicación de voz, una aplicación de comunicación de datos, entre otras.

En algunas realizaciones, el hipervisor 30 expone además una VM 33 de seguridad, que se puede ejecutar de manera concurrente con la VM 32 invitada, protegiendo la VM 32 invitada contra amenazas de seguridad informática tales como el malware y la intrusión. Una VM de seguridad única puede proteger a múltiples VM invitadas que se ejecutan en el respectivo sistema cliente. Los entornos virtuales de la VM 32 invitada y la VM 33 de seguridad se pueden aislar el uno del otro para asegurar que el software malicioso que se ejecuta dentro de la VM 32 invitada no infecta o interfiere de cualquier otra forma con el software que se ejecuta dentro de la VM 33 de seguridad. Por ejemplo, los procesadores virtuales de la VM 33 de seguridad se distinguen de los procesadores virtuales de otras máquinas virtuales que se ejecutan en el sistema 12 cliente; las traducciones de memoria para la VM 33 de seguridad y la VM 32 invitada usan distintos conjuntos de tablas de páginas. La VM 33 de seguridad se puede configurar para colaborar con el servidor 14 de seguridad tal como se muestra en detalle más adelante. Algunas realizaciones de la VM 33 de seguridad comprenden un sistema operativo mínimo, aligerado (por ejemplo, una versión personalizada de OS Linux®), un motor 42 de autoanálisis bajo demanda y un filtro 44 de red. En una realización alternativa, el filtro 44 de red se ejecuta fuera de la VM 33 de seguridad, por ejemplo en el nivel de privilegio de procesador del hipervisor 30 (por ejemplo, nivel raíz, anillo -1).

En algunas realizaciones, el hipervisor 30 puede exponer sólo un subconjunto de dispositivos virtualizados a la VM 32 invitada, y puede dar a la VM 33 de seguridad un uso directo y exclusivo de algunos dispositivos de hardware del sistema 12 cliente. En uno de dichos ejemplos, la VM 32 invitada puede tener un uso exclusivo de los dispositivos 20 de entrada y los dispositivos 22 de salida, pero falta un adaptador de red virtualizado. Mientras tanto, la VM 33 de seguridad puede tener un uso directo y exclusivo del adaptador o adaptadores 26 de red. En una de dichas realizaciones, toda la comunicación hacia y/o desde la VM 32 invitada se envía/recibe a través de la VM 33 de seguridad. El hipervisor 30 puede enrutar de manera activa los paquetes de red entre la VM 32 invitada y la VM 33 de seguridad usando un mecanismo de compartición de memoria, por ejemplo. La VM 33 de seguridad puede además usar el filtro 44 para permitir o evitar de manera selectiva las comunicaciones entre la VM 32 invitada y una tercera parte. Dichas configuraciones se pueden implementar, por ejemplo, usando una tecnología VT-d® de Intel®.

En algunas realizaciones, el software de seguridad que se ejecuta en el sistema 12 cliente comprende además un motor 40 de autoanálisis en vivo que se ejecuta fuera de la VM 32 invitada. El término “autoanálisis” se usa aquí para denotar las actividades destinadas a recopilar información sobre el software que se ejecuta dentro de una VM objetivo desde una posición fuera de la VM respectiva. Ejemplos de autoanálisis incluyen, entre otros, determinar si el software que se ejecuta dentro de la VM respectiva realiza ciertas acciones (por ejemplo, ejecutando ciertas instrucciones de procesador accediendo a ciertos recursos de hardware, usando ciertos servicios del OS, accediendo a ciertas ubicaciones de memoria, etc.). Otros ejemplos de autoanálisis comprenden determinar las direcciones de memoria usadas por los diversos objetos de software que se ejecutan dentro de la VM respectiva, y/o controlar el acceso a una ubicación de memoria indicada por dichas direcciones.

El motor 40 se puede incorporar en el hipervisor 30, por ejemplo como una librería, o se puede entregar como un programa informático distinto e independiente del hipervisor 30, pero que se ejecuta en el nivel de privilegio de procesador del hipervisor 30 (por ejemplo, modo raíz, anillo -1). En una realización alternativa, el motor de autoanálisis en vivo puede ejecutarse en una máquina virtual separada distinta de la VM 32 invitada. El motor 40 puede ser un proceso que tiene un hilo de ejecución planificado separado, o puede operar como una colección de

objetos de código no planificados que se ejecutan cuando son desencadenados por ciertos eventos, tal como se muestra más adelante.

El motor 40 se configura para monitorizar el comportamiento de una pluralidad de entidades ejecutables (por ejemplo, procesos, hilos, aplicaciones). Estos pueden comprender la detección de la ocurrencia de diversos eventos durante la ejecución del software respectivo, y el reporte de manera selectiva de dichos eventos al servidor 14 de seguridad. Se pueden detectar de esta manera diversos tipos de eventos, por ejemplo, llamadas a ciertas funciones OS, llamadas de sistema, etc. Otros ejemplos de eventos detectados incluyen un intento de modificar una función del OS 34 (una manipulación de código comúnmente conocida en la técnica como parcheo o modificación de código), un intento por parte de una entidad de software de inyectar código dentro de otra entidad de software, un intento de lanzar un componente de software que no está firmado digitalmente, y un intento de evitar una verificación de firma digital, entre otros. Otros tipos de eventos detectados pueden incluir la apertura de un archivo, la creación de un archivo, la escritura de un archivo, la eliminación de un archivo, la copia de un archivo, la creación de un proceso, la terminación de un proceso, la planificación de un hilo para su ejecución, la suspensión de un hilo debido a un evento de sincronización (por ejemplo una exclusión mutua), la creación de una pila, la asignación de memoria de la pila, la extensión del tamaño de una pila de ejecución, el cambio de un permiso de acceso a memoria, la realización de una operación de volcado hacia dentro (por ejemplo de disco a memoria), la realización de una operación de volcado hacia fuera (por ejemplo de memoria a disco), la carga de un módulo ejecutable (por ejemplo, una librería compartida – DLL), la apertura de una clave de registro, el renombrado de una clave de registro, la detección de la incorporación de un nuevo dispositivo de hardware, el establecimiento de una nueva conexión de red, la recepción de un paquete de datos, el aumento de los privilegios de ejecución de un hilo, un cambio del permiso de control de acceso discrecional (DAC) asociado con un archivo.

Son conocidos diversos métodos para detectar dichos eventos en la técnica. Estos incluyen parchear ciertas funciones OS, modificar las tablas de envío, etc. En las plataformas de virtualización de hardware, una categoría especial de métodos para detectar los eventos relevantes de seguridad se basa en la detección de una violación de un permiso de acceso a memoria. Los sistemas informáticos más modernos se configuran para operar con memoria virtual y para gestionar traducciones de direcciones de memoria usando estructuras de datos dedicadas, por ejemplo las tablas de páginas. Los sistemas configurados para apoyar la virtualización de hardware normalmente usan una segunda capa de traducciones de direcciones, desde una memoria física invitada vista por cada VM expuesta a la memoria 18 física real del sistema 12 cliente. La segunda traducción de direcciones se consigue normalmente usando hardware acelerado, estructuras y mecanismos de datos dedicadas controladas por el procesador 16, conocido como el segundo nivel de traducción de direcciones (SLAT). Las implementaciones de SLAT populares incluyen tablas de páginas extendidas (EPT) en plataformas Intel®, e indexación de virtualización rápida (RVI)/tablas de páginas anidadas (NPT) en plataformas AMD®. SLAT normalmente permite configurar los permisos de acceso a memoria para cada página de memoria, tal como de lectura/escritura/ejecución.

En algunas realizaciones, el motor 40 de autoanálisis en vivo colabora con el hipervisor 30 para establecer los permisos de ciertas páginas de memoria usando un mecanismo SLAT tal como se describe anteriormente. En uno de dichos ejemplos una página de memoria concreta alberga el código que pertenece a una función OS concreta. Marcar la página respectiva como no ejecutable desencadenará una violación de permiso cuando se haga un intento de ejecutar la respectiva función OS. La violación se puede interpretar mediante el motor 40 de autoanálisis como un indicador de que se ha producido un intento de ejecutar la respectiva función OS.

El procesador 16 se puede configurar para desencadenar un evento de procesador (por ejemplo, una excepción, un fallo, etc.) cuando el software intente acceder a la página respectiva de una manera que viole los permisos de acceso actuales. Un tipo de evento de procesador comprende un evento de salida de VM (VMExit en las plataformas Intel®), en donde el procesador 16 cambia de ejecutar código dentro de la respectiva VM a ejecutar una rutina de manejador fuera de la respectiva VM en respuesta a una violación de un permiso de acceso a memoria. Otra categoría de eventos de procesador comprende una excepción de virtualización (#VE en las plataformas Intel®), en donde el procesador 16 cambia a ejecutar una rutina de manejador dentro de la respectiva VM.

En la configuración ejemplar de la Figura 3-A, un manejador 46a de eventos se ejecuta fuera de la VM invitada monitorizada, en el nivel de privilegio de procesador del hipervisor 30. Dichas realizaciones se pueden apoyar en los eventos de salida de la VM para informar al motor 40 de autoanálisis en vivo sobre la ocurrencia de un evento dentro de la VM 32 invitada. En contraste, en la Figura 3-B, un manejador 46b de eventos se ejecuta dentro de la VM monitorizada, por ejemplo, en el nivel de privilegio de procesador del OS 34 invitado (por ejemplo, anillo 0, modo de núcleo). Dichas configuraciones se pueden basar en las excepciones de virtualización para detectar eventos en la VM. El manejador 46b puede usar un mecanismo de comunicación entre procesos (por ejemplo, usar una sección compartida de la memoria) para enviar información de evento al motor 40 de autoanálisis. Las configuraciones tales como las mostradas en la Figura 3-B pueden ser más eficientes en cuanto a la recopilación de información que aquellas mostradas en la Figura 3-A, ya que el manejador 46b se ejecuta dentro de la VM monitorizada y por lo tanto puede usar funciones y mecanismos del OS 34 invitado para determinar las semánticas de cada evento detectado. Sin embargo, mediante la ejecución dentro de la VM 32 invitada, el manejador 46b puede ser más vulnerable a software malicioso que el manejador 46a.

En lugar de basarse en sólo el motor 40 de autoanálisis en vivo o en la ejecución de software dentro de la VM protegida para analizar los eventos de seguridad, algunas realizaciones de la presente invención despliegan además un segundo, motor 42 de autoanálisis bajo demanda. El motor 40 de autoanálisis en vivo puede detectar la ocurrencia de diversos eventos dentro de la VM invitada monitorizada, pero puede no llevar a cabo un análisis forense sofisticado de dichos eventos, ya que dicho análisis sería computacionalmente demasiado costoso e impactaría de manera negativa la experiencia de usuario. En su lugar, el motor 42 de autoanálisis bajo demanda puede ser invocado para realizar el respectivo análisis forense, pero dicho análisis forense puede ser desencadenado de manera selectiva sólo para un subconjunto de eventos detectados por el motor 40 de autoanálisis en vivo. En algunas realizaciones, la decisión de realizar un análisis forense de un evento es tomada por el servidor 14 de seguridad y comunicada al respectivo sistema 12 cliente.

En algunas realizaciones, el motor 40 de autoanálisis en vivo y el motor 42 de autoanálisis bajo demanda pueden comunicarse el uno con el otro, por ejemplo a través de una sección compartida de memoria y señalización llevada a cabo a través del hipervisor 30. Por ejemplo, mientras se realizan actividades forenses, el motor 42 de autoanálisis bajo demanda puede leer y usar el estado actual del motor 40 de autoanálisis en vivo. A su vez, el motor de introspección en vivo puede recibir notificaciones del motor de introspección bajo demanda, por ejemplo para señalizar que las actividades forenses se están llevando a cabo actualmente.

La Figura 4 muestra una secuencia ejemplar de pasos realizada para configurar la seguridad informática en el sistema 12 cliente según algunas realizaciones de la presente invención. En un escenario típico de protección de una red corporativa contra amenazas de seguridad informática, un administrador de red puede instalar una aplicación de seguridad en cada sistema 12a-d cliente que requiera protección. La aplicación de seguridad puede comprender diversos componentes, tales como el hipervisor 30, los motores 40-42 de autoanálisis, los manejadores 46a-b de eventos, etc. La secuencia ilustrada de pasos se puede llevar a cabo, por ejemplo, mediante una utilidad de instalador de la respectiva aplicación de seguridad. Cuando se instala en ausencia de un entorno de virtualización de hardware, el software de seguridad primero puede asumir el procesador 16 con el nivel de privilegio de procesador más fuerte (por ejemplo, modo raíz, anillo -1) e instalar el hipervisor 30. El hipervisor 30 puede exponer entonces la VM 32 invitada y mover todo el software que se ejecutaba anteriormente en el respectivo sistema cliente para ejecutarse dentro de la VM 32 invitada. El hipervisor 30 puede además configurar la VM 33 de seguridad y configurar un método de compartición de hardware del respectivo sistema cliente entre las VM 32-33. El instalador puede entonces instalar y lanzar el software que se ejecuta dentro de la VM 33 de seguridad, así como el motor 40 de autoanálisis en vivo.

En algunas realizaciones, el hipervisor 30 y/o la VM 33 de seguridad pueden ser lanzados usando un mecanismo de arranque seguro u otra forma de autenticación conocida en la técnica, para asegurar que la VM 33 de seguridad ejecuta software confiable. Configurar la VM 33 de seguridad (paso 204) y/u otro software que se ejecute dentro de la VM 33 de seguridad puede comprender intercambios de autenticación (por ejemplo, verificación hash) con el servidor 14 de seguridad o con otras entidades de autenticación. En una realización ejemplar, el arranque seguro puede emplear un componente de hardware de almacenamiento seguro del sistema 12 cliente, tal como un Módulo de Plataforma de confianza (TPM) en las plataformas Intel®, y emplear además un mecanismo de atestado tal como la Tecnología de Ejecución de Confianza de Intel® (TXT).

En algunas realizaciones, un paso 206 configura un acceso administrativo remoto desde el servidor 14 de seguridad a la VM 33 de seguridad. Dicho acceso puede permitir al servidor 14 de seguridad (de manera automática o ayudado por un operador humano) enviar instrucciones y comandos de manera directa a un sistema cliente protegido, por ejemplo para dar instrucciones al motor 42 de autoanálisis bajo demanda para realizar un tipo particular de análisis forense, o para llevar a cabo una secuencia específica de pasos de limpieza. La configuración del acceso administrativo remoto puede incluir, por ejemplo, la configuración de un túnel (esto es, un canal de comunicación punto a punto seguro) entre el servidor 14 y la VM 33 de seguridad a través del intérprete de comandos seguro(SSH) o protocolos de red privada (VPN). La Figura 5 muestra dicho intercambio ejemplar. Una solicitud 48 de túnel puede ser emitida por bien el sistema 12 cliente o el servidor 14. La solicitud 48 puede comprender un indicador de un protocolo de intercambio, y un conjunto de claves de cifrado. En respuesta, las partes de la comunicación configuran un túnel 49 seguro que comprende una manera específica de cifrar y enrutar paquetes de red entre éstas. En algunas realizaciones, el hipervisor 30 enruta los paquetes de red recibidos desde el servidor 14 de seguridad de manera directa a la VM 33 de seguridad.

La Figura 6 muestra un intercambio de datos ejemplar entre el servidor 14 de seguridad y un sistema cliente protegido según algunas realizaciones de la presente invención. El sistema 12 cliente puede enviar un indicador 50 de eventos para informar al servidor 14 de que un evento se ha producido durante la ejecución del software dentro de la VM 32 invitada. El indicador 50 de eventos puede incluir un indicador de un tipo de evento del respectivo evento, una marca de tiempo del evento, y un identificador (ID de cliente) del respectivo sistema 12 cliente y/o de la VM 32 invitada.

En respuesta, el servidor 14 puede enviar una solicitud 52 de análisis al sistema 12 cliente, la solicitud 52 da instrucciones al motor 42 de autoanálisis bajo demanda para realizar ciertas actividades forenses en el sistema 12 cliente. La solicitud 52 de análisis puede incluir un indicador de una herramienta forense a ser usada por el motor 42 para llevar a cabo la recopilación de datos/análisis de eventos, y/o un indicador de algún tipo de recurso de

seguridad. En algunas realizaciones, se puede enviar una solicitud 52 de análisis al sistema cliente para recibir un indicador de eventos de otro sistema cliente.

En algunas realizaciones, en respuesta a la realización de las actividades forenses solicitadas, el motor 42 de autoanálisis bajo demanda transmite un reporte 54 forense al servidor 14, comprendiendo el reporte 54 el resultado de llevar a cabo las respectivas actividades forenses. El reporte 54 puede incluir, por ejemplo, una lista de objetos de software, una lista de direcciones de memoria, una lista de configuraciones de seguridad o hardware, etc.

En respuesta a la recepción del reporte 54 forense, el servidor 14 puede determinar si el respectivo sistema cliente/VM invitada es vulnerable a un tipo particular de amenaza de seguridad informática, por ejemplo, si el sistema cliente/VM invitada está infectada con software malicioso. Cuando es así, el servidor 14 puede transmitir una alerta 56 de seguridad al respectivo sistema 12 cliente. Algunas realizaciones del servidor 14 transmiten además un indicador 58 de mitigación que comprende, por ejemplo, un indicador de una herramienta de limpieza, o un conjunto de valores de parámetros para configurar un filtro 44 de red.

La Figura 7 muestra una secuencia ejemplar de pasos realizada por el motor 40 de autoanálisis en vivo según algunas realizaciones de la presente invención. El motor 40 se puede configurar para escuchar en busca de al menos dos tipos de mensajes/notificaciones: las notificaciones de los manejadores 46a-b de eventos sobre la ocurrencia de un evento dentro de la VM 32 invitada, y las notificaciones desde la VM 33 de seguridad (por ejemplo, desde un motor 42 de autoanálisis bajo demanda, para señalizar al motor 40 de que se están llevando a cabo actividades forenses actualmente). Al recibir las notificaciones de eventos, en un paso 226, el motor 40 puede realizar un análisis preliminar de dichas notificaciones de eventos. En algunas realizaciones, dichos análisis preliminares implican mínimos costes computacionales, para mantener el impacto sobre la experiencia de usuario tan bajo como sea posible. Por ejemplo, el paso 226 puede filtrar las notificaciones de eventos usando un conjunto de reglas relativamente simple. Sólo ciertos tipos de eventos se pueden reportar al servidor 14 de seguridad (pasos 228-230). En una realización ejemplar, los eventos que son comunicados al servidor 14 de seguridad incluyen, entre otros, un intento de modificar o parchear el núcleo OS, una inyección de un módulo desconocido, y un intento de ejecutar código desde una región de memoria concreta. En respuesta al análisis del evento, en un paso 231 algunas realizaciones del motor 40 dan instrucciones al hipervisor 30 para reanudar la ejecución de la VM 32 invitada. Reanudar la ejecución puede incluir configurar el procesador 16 para emular el evento respectivo. Por ejemplo, cuando el evento comprende la ejecución de una instrucción de procesador concreta, el paso 231 puede comprender emular la instrucción respectiva (por ejemplo, escribir un resultado de la ejecución de la respectiva instrucción de procesador a un registro virtual del procesador virtual de la VM 32 invitada).

Cuando el evento detectado comprende una notificación desde la VM 33 de seguridad (paso 232), algunas realizaciones pueden presentar un mensaje de aviso en un dispositivo de salida del sistema 12 cliente, por ejemplo para informar a un usuario de que el sistema 12 cliente puede experimentar una ralentización temporal debida a actividades forenses o de mitigación de amenaza en curso. En un paso 236 adicional, algunas realizaciones del motor 40 de autoanálisis en vivo pueden colaborar con y ayudar al motor 42 de autoanálisis bajo demanda para llevar a cabo las actividades forenses/de mitigación. Un ejemplo de dicha colaboración incluye al motor 42 de autoanálisis bajo demanda preguntando el estado actual del motor 40 de autoanálisis en vivo.

La Figura 8 muestra una secuencia ejemplar de pasos realizada por un motor 42 de autoanálisis bajo demanda según algunas realizaciones de la presente invención. En una secuencia de pasos 250-252, el motor 42 puede escuchar en busca de solicitudes de análisis desde el servidor 14, por ejemplo a través del túnel 49 seguro establecido entre el servidor 14 y la VM 33 de seguridad (véase la Figura 5). En algunas realizaciones, el hipervisor 30 puede conmutar de manera automática de ejecutar la VM 32 invitada a ejecutar la VM 33 de seguridad y un motor 42 de autoanálisis bajo demanda en respuesta a la recepción de la solicitud 52 de análisis.

La solicitud 52 puede indicar un conjunto de herramientas y/o recursos de seguridad a ser usados en las actividades forenses. Por ejemplo, la solicitud 52 de análisis puede incluir un indicador de ubicación (por ejemplo una dirección de memoria, una ruta de red) que permita la recuperación selectiva de las herramientas respectivas del repositorio 15 de herramientas. En un paso 256, el motor 42 puede acceder a dichas herramientas/recursos a través la red 11 de comunicación. Acceder a una herramienta/recurso puede comprender descargar la respectiva herramienta/recurso desde el repositorio 15 de herramientas en los dispositivos 24 de almacenamiento local del sistema 12 cliente (Figura 2-A). En una realización preferida, el acceso a una herramienta/recurso comprende un motor 42 de autoanálisis bajo demanda que monta las respectivas herramientas/recursos desde sus ubicaciones remotas. El montaje en la presente memoria hace referencia a la creación de una conexión entre el recurso remoto (albergado por el repositorio 15 de herramientas) y un sistema de archivos locales de la VM 33 de seguridad de manera que la VM 33 de seguridad pueda acceder al respectivo recurso a través del sistema de archivos local. Por ejemplo, en un sistema de archivos de Linux®, el montaje se consigue a través del comando “montar”, y dicha conexión comprende un punto de montaje.

En algunas realizaciones, el servidor 14 puede explícitamente dar instrucciones al motor 42 a través del túnel 49 para acceder y/o montar las respectivas herramientas/recursos. En respuesta al acceso a las herramientas/recursos, en una secuencia de pasos 258-260-262, el motor 42 puede llevar a cabo las actividades forenses solicitadas, enviar el reporte 54 forense al servidor 14, y desmontar o de otra manera descartar las respectivas herramientas/recursos.

Descartar las herramientas/recursos al final del análisis forense puede ser beneficioso ya que evita la propagación de software malicioso entre las distintas sesiones de análisis forense, y ya que asegura que los sistemas 12a-c cliente siempre usan la última versión del respectivo recurso disponible en el repositorio 15 de herramientas.

La Figura 9 muestra la secuencia ejemplar de pasos realizada por el servidor 14 de seguridad según algunas realizaciones de la presente invención. El servidor 14 se puede configurar para escuchar en busca de comunicaciones recibidas desde los sistemas 12a-d cliente (pasos 280-282). Cuando dichas comunicaciones comprendan un indicador de evento (esto es, una notificación de que se ha producido un evento en un sistema cliente protegido), algunas realizaciones registran el respectivo evento (paso 286), y determinan según el indicador 50 de eventos si solicitar un análisis forense, y qué tipo de análisis solicitar. En uno de dichos ejemplos cuando el indicador 50 de eventos muestra que se ha producido una inyección de código, el servidor 14 puede solicitar un análisis usando la “captura de aplicación” y/o las herramientas forenses de memoria de Volatibility®. En algunas realizaciones, ciertos tipos de eventos generan solicitudes de análisis complejos, que usan múltiples herramientas y recursos.

La decisión puede ser tomada según un tipo de evento indicado por la notificación actual, y/o según una política de seguridad actualmente en efecto para el respectivo sistema cliente/VM invitada. El proceso de decisión puede incluir por tanto consultar la base de datos 17 cliente en busca de un histórico de eventos y/o una política específica de cliente (paso 290). Otros criterios de decisión pueden incluir una configuración de hardware del respectivo sistema cliente (por ejemplo, si el sistema 12 cliente tiene componentes de hardware específicos, como un tipo particular de procesador, de adaptador de red, etc.). Otros criterios pueden comprender aspectos de software, tales como un tipo de sistema operativo que se ejecuta dentro de la VM 32 invitada, y un tipo de aplicación 36 (servidor web, procesamiento de base de datos, etc.). Dichas estrategias de decisión se basan en el conocimiento de que algún software es vulnerable a amenazas de seguridad informática concretas. El proceso de decisión puede emplear un conjunto de reglas heurísticas, un árbol de decisión, o cualesquiera otros medios conocidos en la técnica. Una realización ejemplar puede usar una red neuronal artificial que recibe datos tales como el tipo de evento y los diversos valores de características de políticas como entrada, y emite un indicador de decisión (por ejemplo, SI/NO) que muestra si solicitar un análisis forense o no.

En algunas realizaciones, la decisión de si solicitar un análisis forense y/o qué tipo de análisis solicitar puede considerar además un histórico de eventos que se producen en el mismo sistema cliente. Dichas realizaciones permiten la correlación de eventos que se producen en distintos momentos en el tiempo, y como tal, puede permitir la detección de malware sofisticado que distribuya una carga maliciosa sobre una multitud de entidades de software (por ejemplo, algunas acciones son realizadas por una primera entidad, y otras acciones por una entidad hija de la primera entidad, o por una segunda entidad que contiene el código inyectado por la primera entidad).

En algunas realizaciones, el proceso de decisión puede considerar un histórico de eventos que se producen en sistemas cliente distintos del emisor de la notificación 50 de eventos. En uno de dichos ejemplos, el servidor 14 puede consultar un registro de eventos en busca de información sobre si un tipo específico de evento se ha producido recientemente en otros sistemas cliente. Un estallido de dicha actividad puede señalizar una ola de malware que se está propagando entre los sistemas cliente (amenaza del día cero), o un ataque coordinado llevado a cabo por múltiples sistemas cliente.

Cuando el evento notificado garantiza un análisis forense (paso 292), un paso 294 selecciona los recursos y/o herramientas forenses a ser usadas por el respectivo sistema cliente para llevar a cabo el análisis forense requerido. La selección se puede hacer según un conjunto de reglas, un árbol de decisión, etc. En un ejemplo, en respuesta a la recepción de una notificación de que se ha producido una inyección de código, el servidor 14 puede solicitar un análisis usando la herramienta de “captura de aplicación”. En algunas realizaciones, ciertos tipos de eventos generan solicitudes de análisis complejos, usando múltiples herramientas y recursos.

Cuando la comunicación recibida desde el sistema 12 cliente comprende el reporte 54 forense (paso 298), el servidor 14 puede analizar el respectivo reporte (paso 300) y posiblemente corroborar el reporte 54 con otros reportes del mismo o de otros sistemas cliente, y determinar si cualquiera de los sistemas 12a-c cliente es vulnerable a una amenaza de seguridad informática (por ejemplo, probablemente bajo el ataque de una entidad maliciosa). Cuando lo es, el servidor 14 puede enviar un aviso al administrador y/o una alerta 56 de seguridad al sistema cliente afectado. En un paso 306 adicional, el servidor 14 puede enviar un indicador 58 de mitigación al respectivo sistema cliente, por ejemplo a través de un túnel 49 seguro. El indicador 58 de mitigación puede incluir comandos, instrucciones, y/o un indicador de un conjunto de herramientas de limpieza.

En las realizaciones ejemplares descritas anteriormente (Figuras 7-8-9), las decisiones concernientes a si solicitar un análisis forense y/o qué tipo de análisis solicitar son tomadas por el servidor 14 de seguridad. En una realización alternativa, el motor 42 de autoanálisis bajo demanda recibe la solicitud 52 de análisis y el indicador 50 de eventos de manera directa desde el motor 40 de autoanálisis en vivo, por ejemplo a través de un mecanismo de comunicación entre procesos gestionado por el hipervisor 30. El motor 42 de autoanálisis bajo demanda puede seleccionar entonces un conjunto de herramientas/recursos forenses a usar según el indicador 50 de eventos y/u otros criterios tal como se describió anteriormente. En dichas realizaciones, la decisión de acceder (por ejemplo, montar) a los respectivos recursos es tomada por tanto en el sistema cliente protegido.

Algunas realizaciones de la presente invención fortalecen más la seguridad empleando el filtro 44 de red para regular la comunicación entre el sistema 12 cliente y el servidor 14 de seguridad o entre el sistema 12 cliente y otras entidades conectadas a la red 11 de comunicación. En un ejemplo, la VM 33 de seguridad se puede configurar para tener un uso exclusivo del adaptador o los adaptadores 26 del sistema 12 cliente. El hipervisor 30 puede enrutar entonces toda la comunicación hacia/desde la VM 32 invitada a través del filtro 44 de red de la VM 33. En algunas realizaciones, cuando el servidor 14 determina que el sistema 12 cliente es vulnerable o es probable que esté bajo ataque de una entidad maliciosa, el servidor 14 puede dar instrucciones al filtro 44 de red para bloquear toda la comunicación hacia/desde la VM 32 invitada, para evitar que la VM 32 invitada conecte a otros sistemas cliente a través de la red local, o para evitar que la VM 32 invitada acceda a Internet. En una realización alternativa, el filtro 44 de red puede bloquear o de otra manera regular la comunicación entre la VM 32 invitada y otra entidad mientras el motor 42 de autoanálisis bajo demanda realiza las actividades forenses.

Algunas realizaciones de la presente invención permiten proteger un número relativamente grande de sistemas cliente (por ejemplo, una red corporativa) contra amenazas de seguridad informática tales como el malware, el spyware, el adware, y a intrusión no autorizada. Algunas realizaciones pueden permitir además una gestión fácil y de confianza de la seguridad desde un punto central, por ejemplo desde un servidor de seguridad acoplado de manera comunicativa con los sistemas cliente protegidos. Dicha centralización puede ser particularmente ventajosa para las aplicaciones de computación en la nube tales como la gestión de torres de servidores web y de infraestructuras de escritorio virtuales, en donde cientos de máquinas virtuales pueden operar de manera simultánea en una única plataforma hardware.

En algunas realizaciones, cada cliente protegido opera un motor de autoanálisis en vivo y un motor de autoanálisis bajo demanda. El motor de autoanálisis en vivo detecta la ocurrencia de ciertos eventos dentro de una máquina virtual protegida expuesta sobre el respectivo sistema cliente y comunica la ocurrencia a un servidor de seguridad remoto. A su vez, el servidor puede seleccionar una herramienta forense de un repositorio de herramientas según el tipo de evento del evento reportado, y/o según las particularidades de hardware y/o software del respectivo cliente. El motor de autoanálisis bajo demanda puede recuperar la herramienta forense y ejecutarla para realizar un análisis forense del evento (por ejemplo, para descubrir un contexto del respectivo evento). Un resultado del análisis forense puede ser comunicado entonces al servidor de seguridad, que puede usar la información para determinar si el respectivo cliente está bajo ataque por software malicioso o un intruso.

Algunas realizaciones de la presente invención se basan en la percepción de que las soluciones de seguridad informática normales para proteger un gran número de clientes no responden bien a los ataques. Incluso cuando las soluciones existentes son capaces de detectar el ataque, el análisis y la remediación pueden requerir que un operador humano sea enviado al sistema cliente afectado, especialmente en el caso de amenazas sofisticadas tales como el ransomware y la intrusión maliciosa. En contraste, en algunas realizaciones de la presente invención, la recopilación de datos personalizada/investigación de los hechos (análisis forense) y la mitigación de amenazas (por ejemplo, limpieza de malware) es realizada de manera automática sobre la máquina de cliente, bajo la instrucción directa del servidor de seguridad. Una realización ejemplar de la presente invención establece un canal de comunicación punto a punto seguro entre un cliente protegido y el servidor de seguridad, lo que permite a un operador humano configurar de manera remota el respectivo cliente, e incluso dirigir las operaciones forenses/de limpieza en el respectivo cliente desde un terminal remoto.

Agregando los resultados de detección de eventos y análisis forense de múltiples clientes, algunas realizaciones pueden permitir una rápida detección de malware anteriormente desconocido (también conocido en la técnica como ataque de día cero). La detección de amenazas del lado del servidor convencional normalmente requiere una sustancial potencia de computación del lado del servidor. En contraste a los escenarios de detección cliente-servidor típicos, en algunas realizaciones de la presente invención algunas operaciones relacionadas con la seguridad costosas tales como el filtrado de eventos y el análisis forense son llevadas a cabo por el sistema cliente en sí. Algunas realizaciones por lo tanto usan una potencia de computación distribuida de múltiples máquinas clientes para realizar algunas operaciones de detección de amenazas y análisis costosas, en lugar de descargar la mayor parte de la carga computacional al servidor de seguridad.

Algunas realizaciones de la presente invención se basan además en la observación de que no todos los eventos que se producen en un sistema informático son intrínsecamente informativos o indicativos de una amenaza a la seguridad informática. El mismo tipo de evento (por ejemplo, acceder a una URL, abrir un archivo de disco, etc.) puede indicar malicia en algunos escenarios, mientras que puede ser totalmente benigno en otros escenarios. En uno de dichos ejemplos, un evento puede no ser indicativo de malicia cuando se produce de manera aislada, pero puede ser indicativo de malware cuando se produce como parte de una secuencia específica de eventos. Por ejemplo, escribir a un archivo de disco puede ser una operación benigna cuando se produce de manera aislada (esto es, muchos procesos y aplicaciones acceden al disco de manera legítima). Sin embargo, el evento de escritura puede ser sospechoso cuando la entidad que realiza la escritura es un recipiente de código inyectado desde otra entidad. En algunas realizaciones de la presente invención, el motor de autoanálisis en vivo pre filtra los eventos detectados según diversos criterios, y sólo reporta un subconjunto seleccionado de dichos eventos al servidor de seguridad. Esta estrategia reduce más la carga computacional en el servidor.

En aplicaciones de seguridad informática convencionales, las herramientas de seguridad del lado del cliente (por ejemplo, las rutinas anti malware, las bases de datos de firmas, etc.) residen en el sistema informático cliente, y se mantienen actualizadas mediante actualizaciones de software periódicas. En contraste, en algunas realizaciones de la presente invención, las herramientas de seguridad se mantienen en un repositorio centralizado (por ejemplo, un servidor dedicado en una red corporativa). Los sistemas cliente pueden recuperar sólo las herramientas que son necesarias, bajo demanda, cuando es mandado por el servidor de seguridad, eliminando así la necesidad de actualizaciones masivas de software y una administración de red costosa en tiempo. En una realización preferida, los sistemas cliente acceden de manera remota a herramientas de seguridad montando el repositorio de herramientas centralizado remoto en un sistema de archivos del cliente.

Mantener la mayoría de las herramientas de seguridad en un repositorio central, a diferencia del cliente protegido o el servidor de seguridad, asegura que los clientes tengan acceso a la versión más reciente de las respectivas herramientas de seguridad. Otra ventaja significativa de dichas configuraciones es que permiten que el software del lado del cliente y del servidor sea sustancialmente menor que en los sistemas de seguridad convencionales. Dichos componentes más livianos (por ejemplo, el hipervisor 30, los motores 40-42 de autoanálisis en vivo y bajo demanda) son más fáciles de desarrollar, mantener, y desplegar a los clientes. Requieren también menos actualizaciones que las soluciones de seguridad todo en uno que incluyen herramientas forenses y/o de mitigación de amenazas. Además, los componentes más livianos exponen una relativamente menor superficie de ataque al software malicioso y a los piratas informáticos

En realizaciones preferidas, el motor de autoanálisis bajo demanda se ejecuta dentro de una máquina virtual dedicada, separada de la VM invitada protegida. Junto a la seguridad aumentada, dichas configuraciones tienen otras ventajas importantes, por ejemplo la facilidad de despliegue y una portabilidad aumentada. La ejecución de una herramienta de seguridad dentro de una máquina virtual puede permitir a un desarrollador desplegar una única versión de la respectiva herramienta a todos los sistemas cliente protegidos, y permite además usar entornos de desarrollo relativamente sofisticados y/o entornos de trabajo tales como Python® o .Net®, por ejemplo. En una de dichas realizaciones ejemplares, todas las VM de seguridad se pueden configurar de manera uniforme para ejecutar el OS Linux® y un conjunto de librerías de Python®, mientras que las herramientas forenses/de mitigación de amenazas se pueden desplegar en Python®. Por contraste en una realización alternativa, el motor de autoanálisis bajo demanda puede ejecutarse en el nivel de privilegio de procesador del hipervisor 30, por ejemplo, junto a o integrado con el motor 40 de autoanálisis en vivo. Sin embargo, en dichas configuraciones, el motor de autoanálisis bajo demanda y/o las herramientas de seguridad pueden necesitar ser codificadas/compiladas/probadas de manera específica para las particularidades del hardware y/o software del respectivo sistema cliente, que pueden impactar de manera sustancial a los costes de desarrollo y el tiempo necesario hasta llegar al mercado del software de seguridad.

Otra ventaja de la ejecución del motor 42 de autoanálisis bajo demanda de manera separada del motor 40 de autoanálisis en vivo (por ejemplo, en una máquina virtual separada) es que en dichas configuraciones el motor 42 puede ejecutarse de manera concurrente con la VM invitada protegida. En las soluciones de seguridad típicas, la ejecución de la máquina virtual protegida se suspende durante la duración del evento de análisis (proceso forense de memoria, etc.). En contraste, en algunas realizaciones de la presente invención, el análisis de un evento se realiza de manera fuera de línea, esto es, mientras que la máquina virtual protegida continua la ejecución. Dichas configuraciones reducen el impacto de las operaciones de seguridad informática, creando una mejor experiencia de usuario.

Queda claro para alguien experto en la técnica que las realizaciones anteriores pueden ser alteradas de muchas maneras sin salir del alcance de la invención. Por consiguiente, el alcance de la invención debería quedar determinado por las reivindicaciones siguientes.

REIVINDICACIONES

1. Un sistema (12) informático

que comprende un procesador de hardware y una memoria, configurado el procesador de hardware para ejecutar un hipervisor (30)

configurado para exponer

una máquina (32) virtual y una VM (33) de seguridad distinta de la VM (32) invitada, estando dicho procesador configurado para además ejecutar un motor (40) de autoanálisis en vivo y un motor (42) de autoanálisis bajo demanda, en donde:

el motor de autoanálisis en vivo se ejecuta fuera de las VM invitada y de seguridad, y en donde la ejecución del motor de autoanálisis en vivo comprende el procesador de hardware, en respuesta a la detección de una ocurrencia de un evento dentro de la VM invitada, que transmite un indicador del evento a un sistema informático servidor remoto sobre una red de comunicación; y

el motor de autoanálisis bajo demanda se ejecuta dentro de la VM de seguridad, y en donde la ejecución del motor de autoanálisis bajo demanda comprende llevar a cabo los pasos de:

en respuesta al motor de autoanálisis en vivo transmitir el indicador del evento al sistema informático servidor remoto, recibir una solicitud de análisis desde el sistema informático servidor remoto, indicando la solicitud de análisis una herramienta de seguridad que reside en un repositorio de herramientas remoto configurado para distribuir las herramientas de seguridad a una pluralidad de clientes que incluye el sistema informático cliente, comprendiendo la herramienta de seguridad software configurado para analizar la ocurrencia del evento, seleccionada la herramienta de seguridad por el sistema informático servidor remoto según un tipo de evento del evento,

en respuesta a la recepción de la solicitud de análisis, identificar la herramienta de seguridad según la solicitud de análisis,

en respuesta a identificar la herramienta de seguridad, recuperar de manera selectiva la herramienta de seguridad desde el repositorio de herramientas, en donde la recuperación de la herramienta de seguridad comprende conectarse al repositorio de herramientas central a través de la red de comunicación,

en respuesta a recuperar de manera selectiva la herramienta de seguridad, ejecutar la herramienta de seguridad, y

en respuesta a ejecutar la herramienta de seguridad, transmitir un resultado de la ejecución de la herramienta de seguridad al sistema informático servidor remoto.

2. El sistema informático cliente de la reivindicación 1, en donde la ejecución del motor de auto análisis bajo demanda comprende además los pasos de:

en respuesta a la transmisión del resultado al sistema informático servidor remoto, recibir desde el sistema informático servidor remoto un indicador de una herramienta de mitigación que reside en el repositorio de servidor remoto, comprendiendo la herramienta de mitigación software configurado para incapacitar el software malicioso que se ejecuta en el sistema informático cliente; y

en respuesta a la recepción del indicador de la herramienta de mitigación, recuperar y ejecutar la herramienta de mitigación.

3. El sistema informático cliente de la reivindicación 1, en donde la ejecución del motor de autoanálisis en vivo comprende además los pasos de:

en respuesta a la detección de la ocurrencia del evento, determinar según un tipo de evento del evento si se satisface una condición de elegibilidad de evento; y

en respuesta, transmitir el indicador del evento al sistema informático servidor remoto sólo cuando se satisface la condición de elegibilidad de evento.

4. El sistema informático cliente de la reivindicación 1, en donde la recuperación de la herramienta de seguridad desde el repositorio de herramientas remoto comprende montar el repositorio de herramientas remoto en un sistema de archivos de la VM de seguridad.

5. El sistema informático cliente de la reivindicación 1, en donde la VM de seguridad comprende además un filtro de red, en donde el hipervisor se configura además para enrutar tráfico de red entre la VM invitada y una parte remota a través del filtro de red, y en donde el filtro de red se configura, en respuesta a la recepción de una alerta de

seguridad desde el sistema informático servidor remoto, para restringir el tráfico de red entre la VM invitada y la parte remota.

6. El sistema informático cliente de la reivindicación 1, en donde el resultado de la ejecución de la herramienta de seguridad comprende una captura de memoria de una aplicación que se ejecuta dentro de la VM invitada y/o una captura de memoria de un núcleo de un sistema operativo de la VM invitada.

7. El sistema informático cliente de la reivindicación 1, en donde el resultado de la ejecución de la herramienta de seguridad comprende una lista de entidades de software que se ejecutan dentro de la VM invitada.

8. El sistema informático cliente de la reivindicación 1, en donde el resultado de la ejecución de la herramienta de seguridad comprende un indicador de una configuración de hardware del sistema informático cliente.

9. El sistema informático cliente de la reivindicación 1, en donde:

el hipervisor se configura para establecer un canal de comunicación punto a punto seguro entre el sistema informático servidor remoto y la VM de seguridad; y

el motor de autoanálisis bajo demanda se configura para recibir la solicitud de análisis y transmitir el resultado a través del canal de comunicación punto a punto seguro.

10. Un ordenador (14)

servidor configurado para realizar las transacciones de seguridad informática con una pluralidad de sistemas cliente, (12a-d) teniendo el sistema (14) informático de servidor un procesador de hardware configurado para llevar a cabo los pasos de:

en respuesta a la recepción de un indicador de evento desde un sistema cliente de la pluralidad de sistemas cliente, siendo el indicador de evento indicativo de una ocurrencia de un evento dentro de una máquina (32) virtual invitada que se ejecuta en el sistema (12) cliente, seleccionar una herramienta de seguridad que reside en un repositorio (15) de herramientas remoto

configurado para distribuir

herramientas de seguridad a la pluralidad de sistemas cliente, comprendiendo la herramienta de seguridad software configurada para analizar la ocurrencia del evento, en donde la selección de la herramienta de seguridad es realizada según un tipo de evento del evento;

en respuesta a la selección de la herramienta de seguridad, transmitir una solicitud de análisis al sistema cliente a través de una red de comunicación, comprendiendo la solicitud de análisis un identificador de la herramienta de seguridad; y

en respuesta a la transmisión del indicador de la herramienta de seguridad, recibir desde el sistema cliente un resultado de la ejecución de la herramienta de seguridad en el sistema cliente;

en donde el sistema cliente se configura para ejecutar un hipervisor, (30) un motor de autoanálisis en vivo, (40) y un motor de autoanálisis bajo demanda, (42) en donde:

el hipervisor se configura para exponer la VM (32) invitada y una VM (33) de seguridad distinta de la VM invitada, (32) en donde el motor de autoanálisis bajo demanda se ejecuta dentro de la VM de seguridad, y en donde el motor de autoanálisis en vivo se ejecuta fuera de las VM invitada y de seguridad,

el motor de autoanálisis en vivo se configura, en respuesta a la detección de la ocurrencia del evento, para transmitir el indicador de evento al sistema informático de servidor, y recibir la solicitud de análisis que obliga al motor de autoanálisis bajo demanda a:

en respuesta a la recepción de la solicitud de análisis, identificar la herramienta de seguridad según la solicitud de análisis,

en respuesta a la identificación de la herramienta de seguridad, recuperar de manera selectiva la herramienta de seguridad desde el repositorio de herramientas, en donde la recuperación de la herramienta de seguridad comprende que el sistema cliente se conecte al repositorio de herramientas remoto a través de la red de comunicación, y

en respuesta a la recuperación de la herramienta de seguridad, ejecutar la herramienta de seguridad para producir el resultado.

11. El ordenador servidor de la reivindicación 10, caracterizado además porque el procesador de hardware que lleva a cabo los pasos de:

en respuesta a la recepción del resultado, detectar una condición del sistema cliente según el resultado, consistente la condición seleccionada de un grupo de una intrusión del sistema cliente y una infección del sistema cliente con software malicioso; y

en respuesta a la detección de la condición, transmitir una alerta de seguridad al sistema cliente, la alerta de seguridad indicativa de la condición.

12. El ordenador servidor de la reivindicación 11, caracterizado además porque el procesador de hardware lleva a cabo los pasos de:

en respuesta a la detección de la condición, seleccionar una herramienta de mitigación de la condición del repositorio de herramientas remoto según la condición; y

en respuesta a la selección de la herramienta de mitigación de la condición, formular la alerta de seguridad para incluir un indicador de la herramienta de mitigación.

13. El ordenador servidor de la reivindicación 10, en donde la VM comprende además un filtro de red, en donde el hipervisor se configura además para enrutar el tráfico de red entre la VM invitada y una parte remota a través del filtro de red, y en donde la alerta de seguridad se configura para provocar que el filtro de red restrinja el tráfico de red entre la VM invitada y la parte remota.

14. El ordenador servidor de la reivindicación 10, en donde el resultado de la ejecución de la herramienta de seguridad comprende una captura de memoria de una aplicación que se ejecuta dentro de la VM invitada y/o una captura de memoria de un núcleo de un sistema operativo de la VM invitada.

15. El ordenador servidor de la reivindicación 10, en donde el resultado de la ejecución de la herramienta de seguridad comprende una lista de entidades de software que se ejecutan dentro de la VM invitada.

16. El ordenador servidor de la reivindicación 10, en donde el resultado de la ejecución de la herramienta de seguridad comprende un indicador de una configuración de hardware del sistema informático cliente.

17. El ordenador servidor de la reivindicación 10, en donde:

el hipervisor se configura para establecer un canal de comunicación punto a punto seguro entre el ordenador servidor y la VM de seguridad; y

el procesador de hardware se configura además para enviar la solicitud de análisis y para recibir el resultado de la ejecución de la herramienta de seguridad a través del canal de comunicación punto a punto seguro.

18. Un producto de programa informático que comprende un conjunto de instrucciones que, al ser ejecutadas en un procesador de hardware de un sistema informático cliente, (12) provocan que el sistema (12) informático cliente cree un hipervisor (30) configurado para exponer una máquina virtual (32) invitada y una VM (33) de seguridad distinta de la VM (32) invitada,

en donde las instrucciones son las causantes de que el sistema informático cliente ejecute un motor (40) de autoanálisis en vivo y un motor de autoanálisis bajo demanda, (42) en donde

el motor de autoanálisis en vivo se ejecuta fuera de las VM invitada y de seguridad, en donde la ejecución del motor de autoanálisis en vivo comprende que el procesador de hardware, en respuesta a la detección de una ocurrencia de un evento dentro de la VM invitada, transmita un indicador del evento a un sistema informático servidor remoto a través de una red de comunicación; y

el motor de autoanálisis bajo demanda se ejecuta dentro de la VM de seguridad, en donde la ejecución del motor de autoanálisis bajo demanda comprende que el procesador de hardware tenga que llevar a cabo los pasos de:

en respuesta al motor de autoanálisis en vivo transmitir el indicador del evento al sistema informático servidor remoto, recibir una solicitud de análisis desde el sistema informático servidor remoto, indicando la solicitud de análisis una herramienta de seguridad que reside en un repositorio de herramientas remoto configurado para distribuir las herramientas de seguridad a una pluralidad de clientes que incluye el sistema informático cliente, comprendiendo la herramienta de seguridad software configurado para analizar la ocurrencia del evento, la herramienta de seguridad seleccionada por el sistema informático servidor remoto según un tipo de evento del evento.

en respuesta a la recepción de la solicitud de análisis, identificar la herramienta de seguridad según la solicitud de análisis,

en respuesta a la identificación de la herramienta de seguridad, recuperar de manera selectiva la herramienta de seguridad desde el repositorio de herramientas, en donde la recuperación de la herramienta de seguridad comprende conectar con el repositorio de herramientas central a través de la red de comunicación,

en respuesta a la recuperación de manera selectiva de la herramienta de seguridad, ejecutar la herramienta de seguridad, y

en respuesta a la ejecución de la herramienta de seguridad, transmitir un resultado de la ejecución de la herramienta de seguridad al sistema informático servidor remoto.

19. El sistema informático cliente de la reivindicación 1, en donde el evento comprende un elemento seleccionado de un grupo consistente de un intento de modificar una función de OS (34) y un intento por una entidad de software de inyectar código dentro de otra entidad de software.

20. El sistema informático cliente de la reivindicación 1, en donde la detección del evento comprende detectar una violación de un permiso de acceso definido para la sección de la memoria.

21. El sistema informático cliente de la reivindicación 1, en donde la VM de seguridad comprende además un filtro de red, en donde el hipervisor se configura además para enrutar el tráfico de red entre la VM invitada y una parte remota a través del filtro de red, y en donde el filtro de red se configura, en respuesta a la solicitud de análisis, para restringir el tráfico de red entre la VM invitada y la parte remota durante la duración de la ejecución de la herramienta de seguridad.

22. El sistema informático cliente de la reivindicación 1, en donde la ejecución del motor de autoanálisis bajo demanda comprende el procesador de hardware que ejecuta además los pasos de:

en respuesta a la ejecución de la herramienta de seguridad, eliminar la herramienta de seguridad de una memoria del sistema informático de cliente;

en respuesta a la eliminación de la herramienta de seguridad, recibir otra solicitud de análisis del sistema informático de servidor, indicando la otra solicitud de análisis otra herramienta de seguridad del repositorio de herramientas;

en respuesta a la recepción de la otra herramienta de análisis, recuperar de manera selectiva la otra herramienta del repositorio de herramientas;

en respuesta a la recuperación de la otra herramienta de seguridad, ejecutar la herramienta de seguridad para producir un segundo resultado; y

transmitir el segundo resultado al sistema informático servidor.

23. El sistema informático cliente de la reivindicación 1, caracterizado además por el procesador de hardware, en respuesta a la recepción de la solicitud de análisis, presentando una notificación en un dispositivo de salida del sistema informático cliente, indicando la notificación una ralentización de la VM invitada.

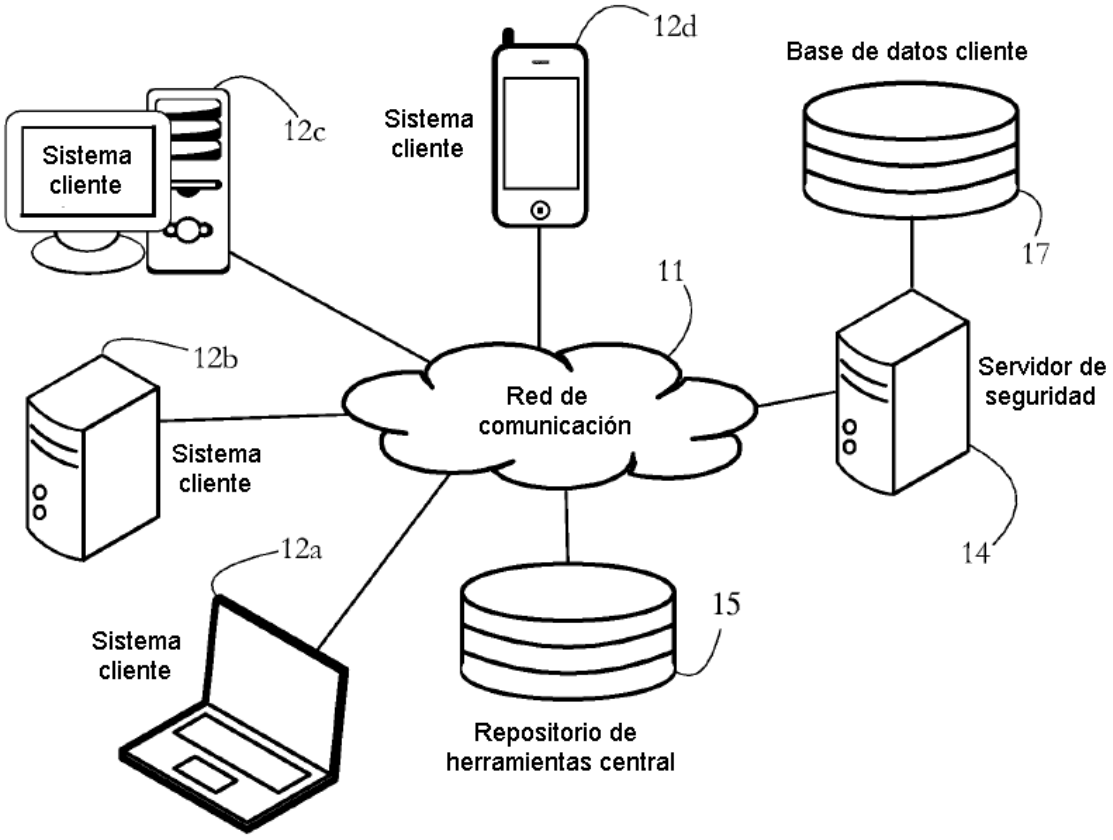
12d
Base de datos cliente
Sistema cliente
12c
Sistema cliente
17
11
12b
Red de comunicación
Servidor de seguridad
Sistema cliente
14
12a
15
Sistema cliente
Repositorio de herramientas central

FIG. 1

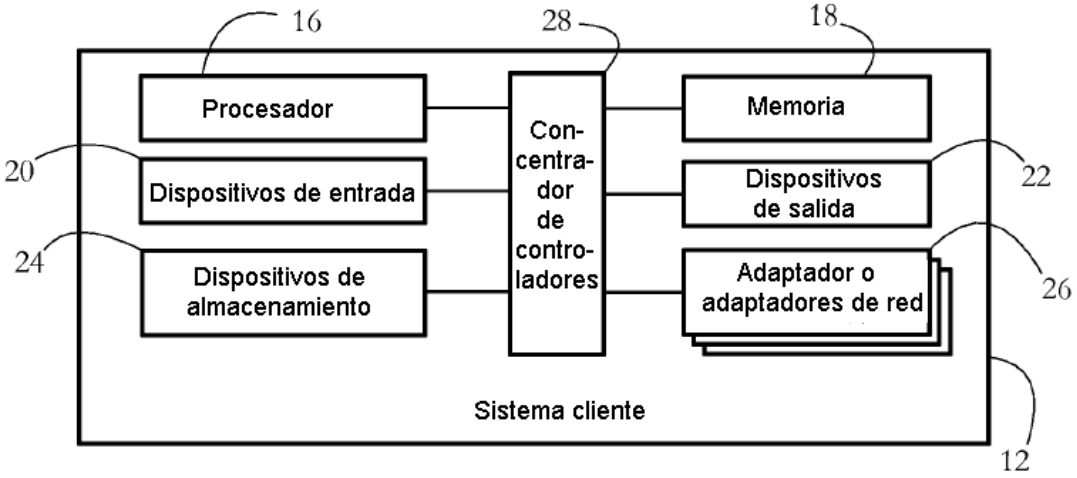
16
28
18
Procesador
Con-
centra-
dor
de
contro-
ladores
Memoria
20
Dispositivos de entrada
Dispositivos
de salida
22
24
Dispositivos de
almacenamiento
Adaptador o
adaptadores de red
26
Sistema cliente
12

FIG. 2-A

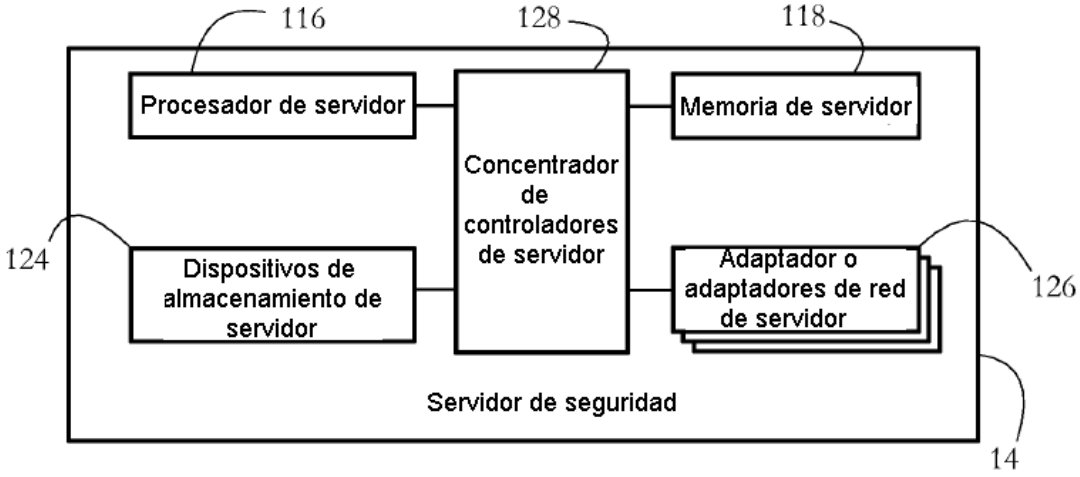
116
128
118
Procesador de servidor
Concentrador
de
controladores
de servidor
Memoria de servidor
124
Dispositivos de
almacenamiento de
servidor
Adaptador o
adaptadores de red
de servidor
126
Servidor de seguridad
14

FIG. 2-B

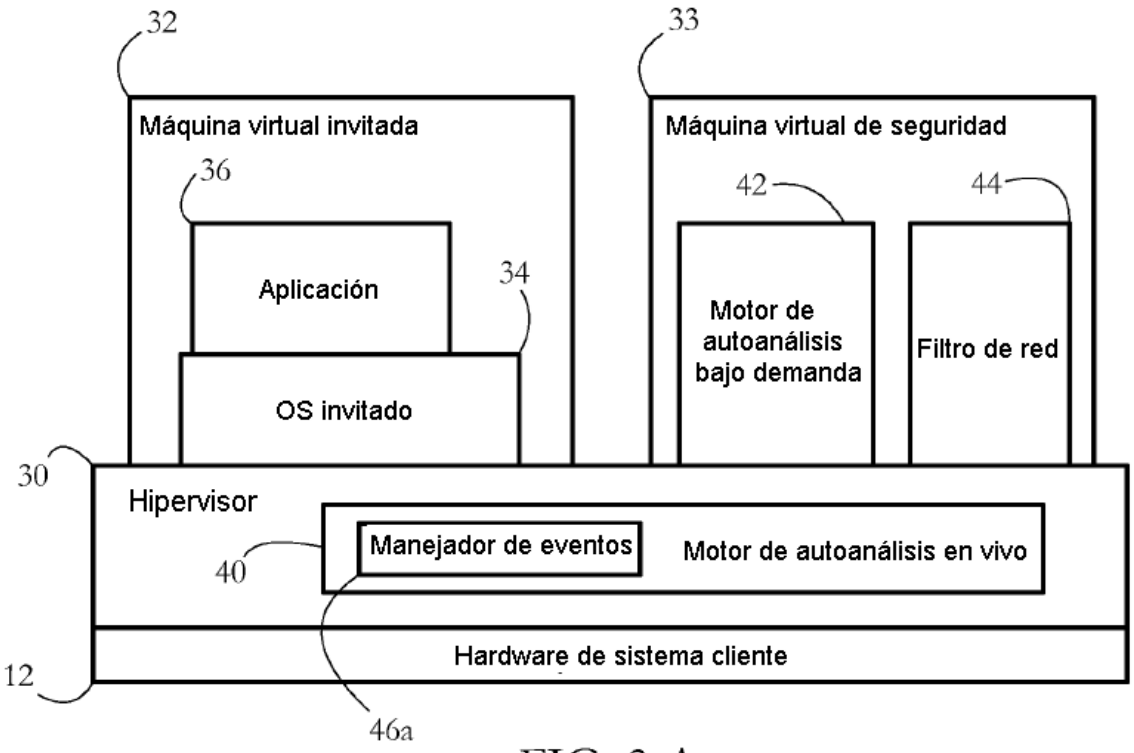
32
33
Máquina virtual invitada
Máquina virtual de seguridad
36
42
44
Aplicación
34
Motor de autoanálisis bajo demanda
Filtro de red
OS invitado
30
Hipervisor
Manejador de eventos
40
Motor de autoanálisis en vivo
Hardware de sistema cliente
12
46a

FIG. 3-A

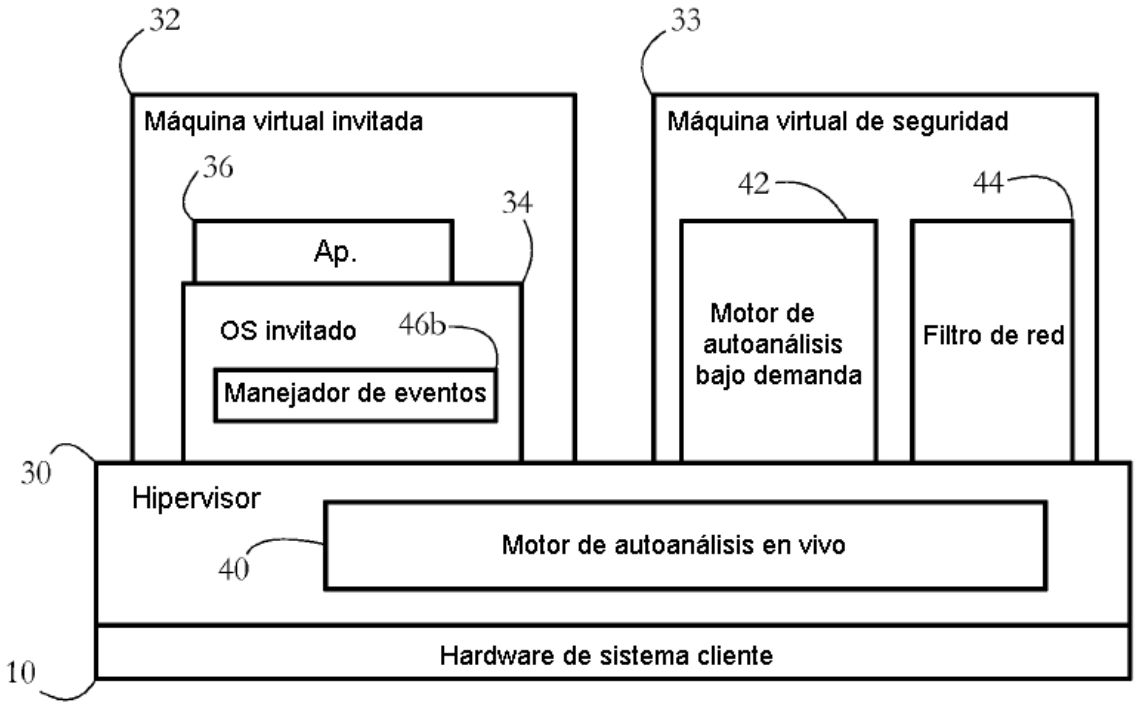
32
33
Máquina virtual invitada
Máquina virtual de seguridad
36
34
42
44
Ap.
OS invitado
46b
Motor de autoanálisis bajo demanda
Filtro de red
Manejador de eventos
30
Hipervisor
40
Motor de autoanálisis en vivo
Hardware de sistema cliente
10

FIG. 3-B

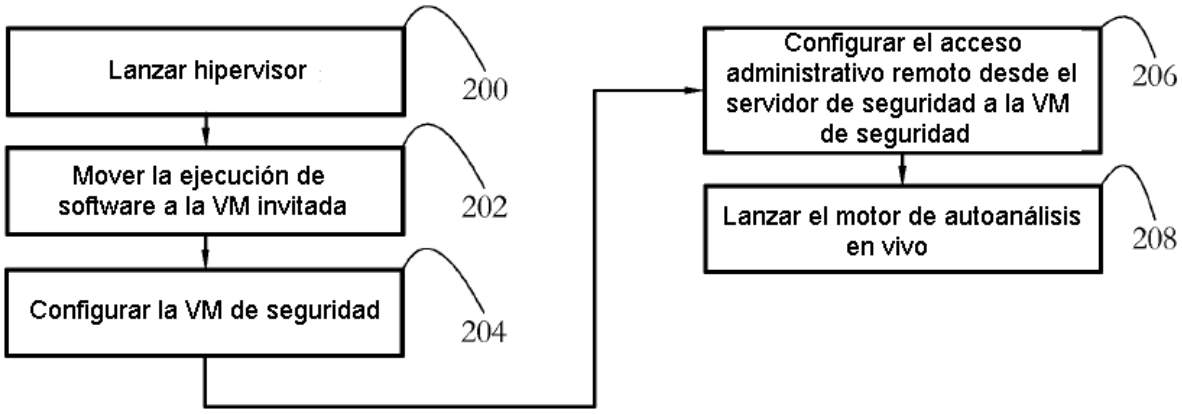
Lanzar hipervisor
200
Mover la ejecución de software a la VM invitada
202
Configurar la VM de seguridad
204
Configurar el acceso administrativo remoto desde el servidor de seguridad a la VM de seguridad
206
Lanzar el motor de autoanálisis en vivo
208

FIG. 4

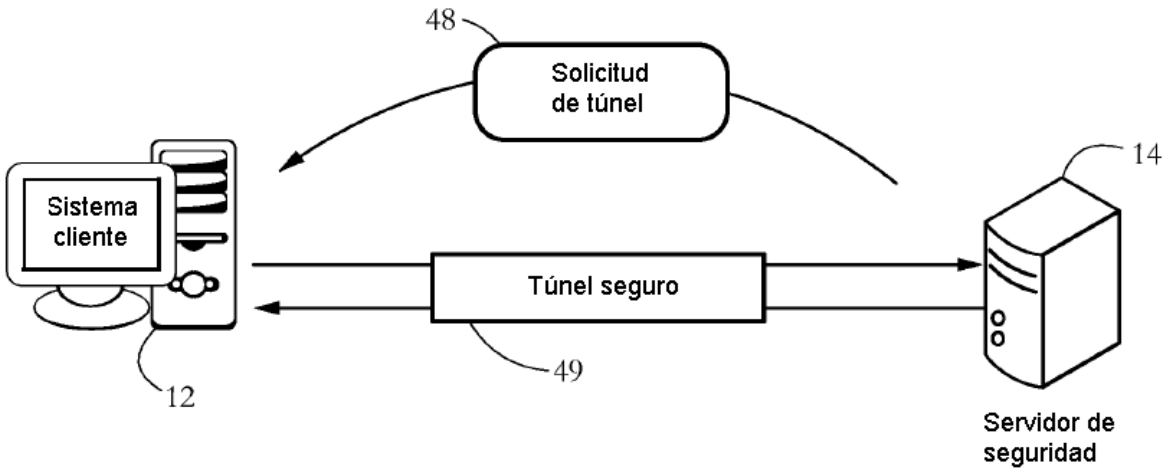
48
Solicitud de túnel
14
Sistema cliente
Túnel seguro
49
12
Servidor de seguridad

FIG.5

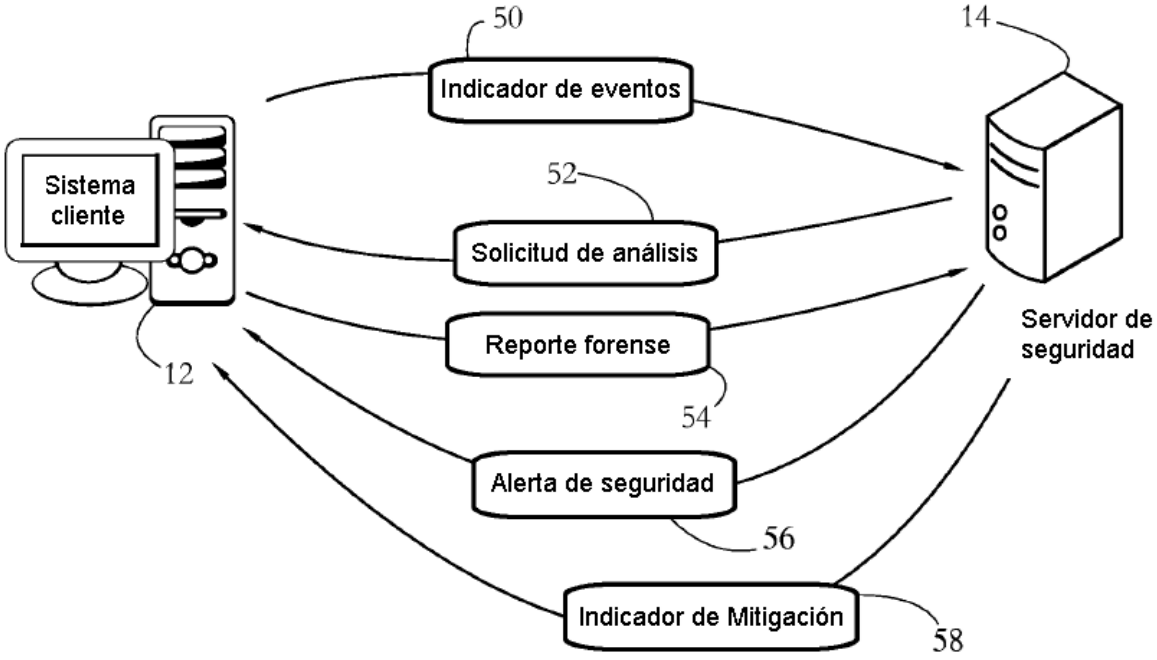
50
14
Indicador de eventos
52
Sistema cliente
Solicitud de análisis
Reporte forense
Servidor de seguridad
12
54
Alerta de seguridad
56
Indicador de Mitigación
58

FIG. 6

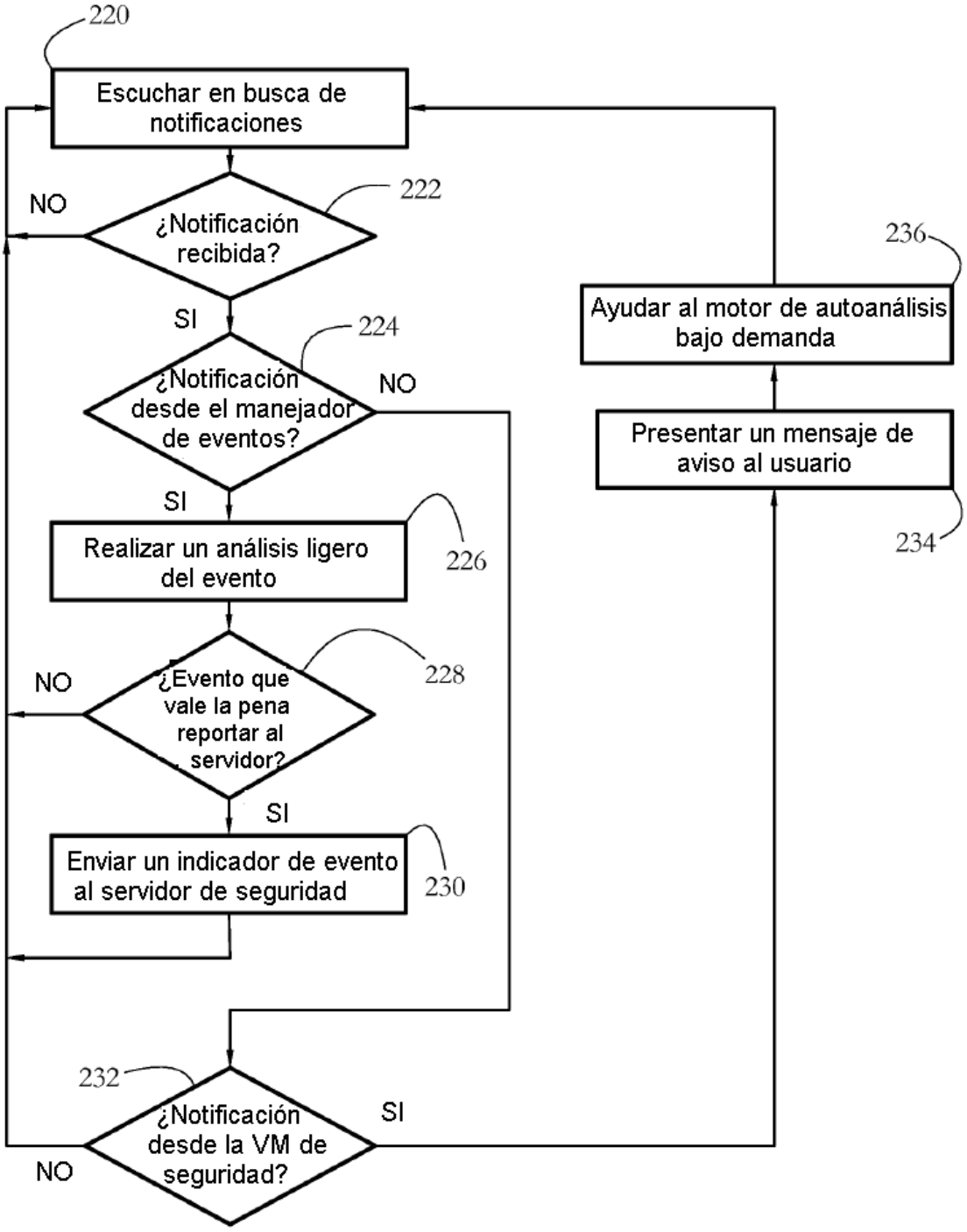
220
Escuchar en busca de notificaciones
222
¿Notificación recibida?
NO
SI
224
¿Notificación desde el manejador de eventos?
NO
SI
Realizar un análisis ligero del evento
226
228
¿Evento que vale la pena reportar al servidor?
NO
SI
Enviar un indicador de evento al servidor de seguridad
230
232
¿Notificación desde la VM de seguridad?
NO
SI
Presentar un mensaje de aviso al usuario
234
236
Ayudar al motor de autoanálisis bajo demanda

FIG. 7

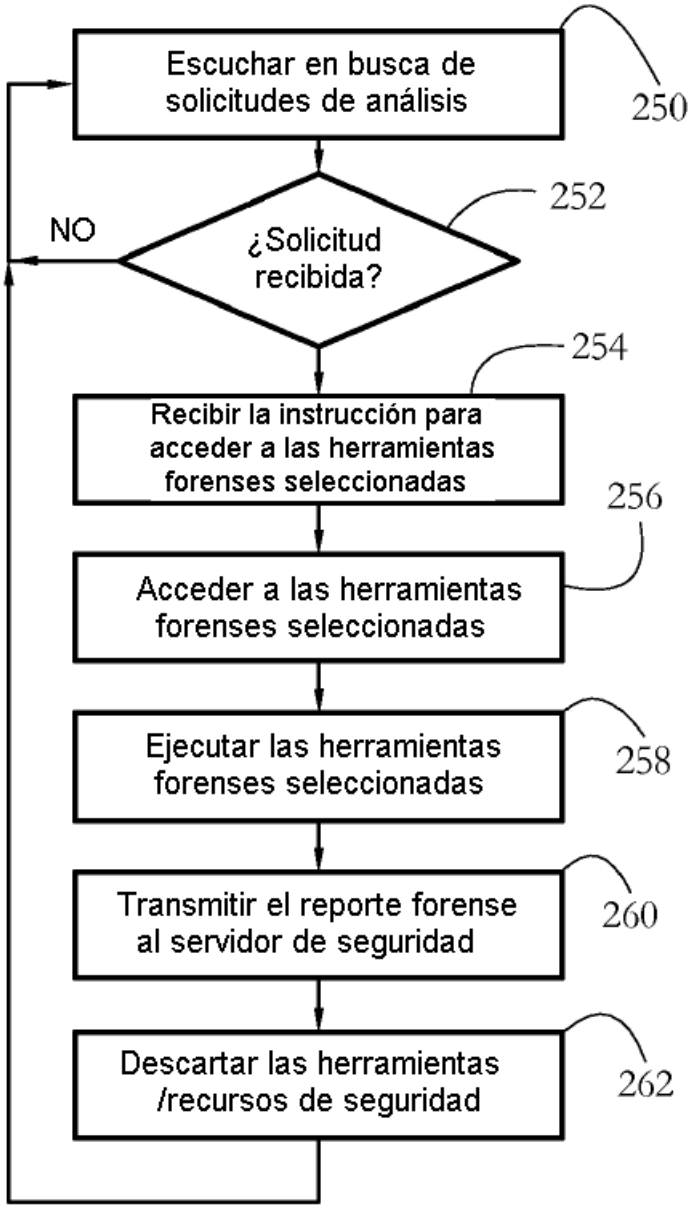
Escuchar en busca de solicitudes de análisis
250
¿Solicitud recibida?
252
NO
Recibir la instrucción para acceder a las herramientas forenses seleccionadas
254
Acceder a las herramientas forenses seleccionadas
256
Ejecutar las herramientas forenses seleccionadas
258
Transmitir el reporte forense al servidor de seguridad
260
Descartar las herramientas /recursos de seguridad
262

FIG. 8

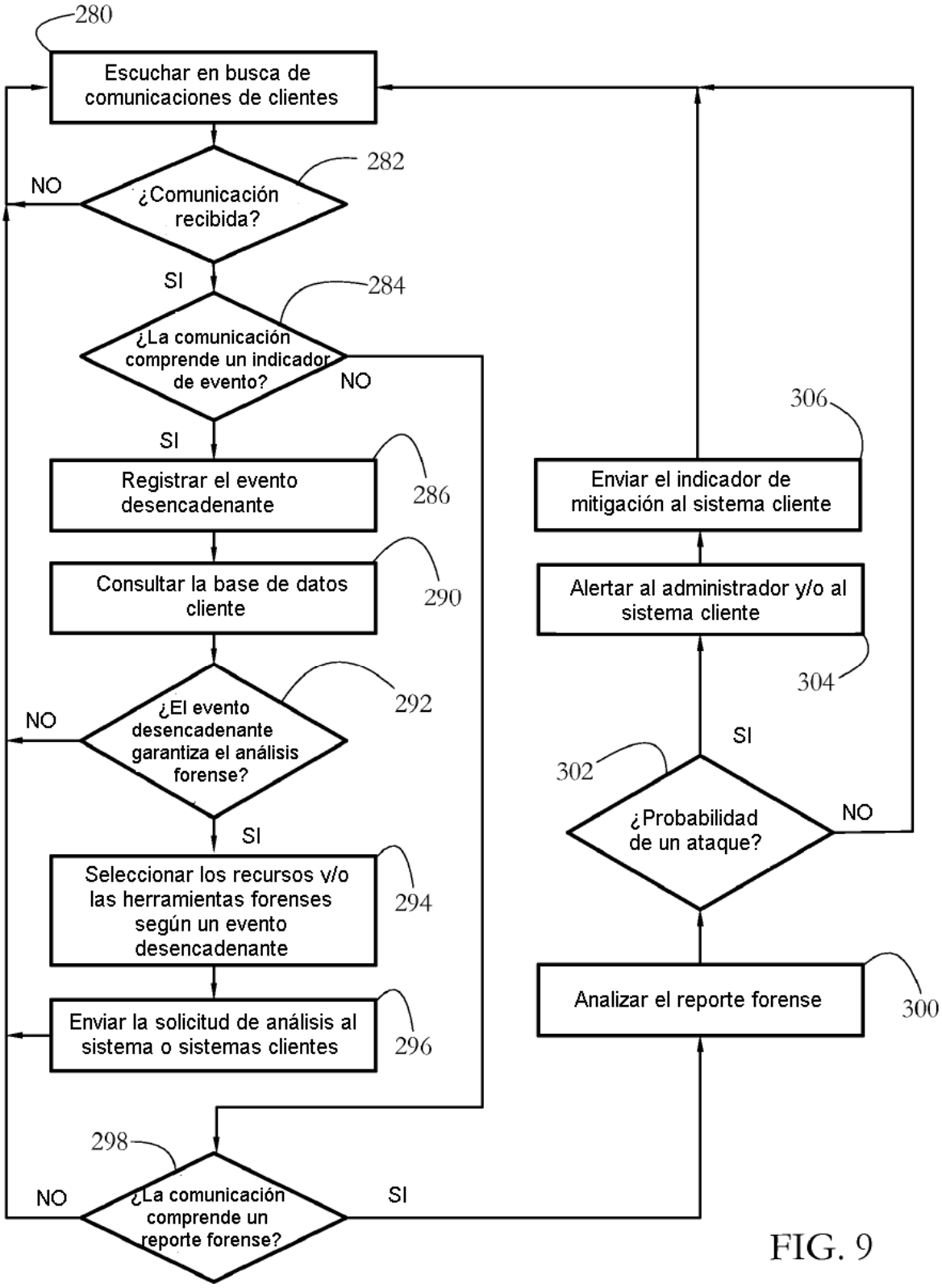
280
Escuchar en busca de comunicaciones de clientes
282
¿Comunicación recibida?
NO
SI
284
¿La comunicación comprende un indicador de evento?
NO
SI
Registrar el evento desencadenante
286
Consultar la base de datos cliente
290
¿El evento desencadenante garantiza el análisis forense?
292
NO
SI
Seleccionar los recursos y/o las herramientas forenses según un evento desencadenante
294
Enviar la solicitud de análisis al sistema o sistemas clientes
296
298
¿La comunicación comprende un reporte forense?
NO
SI
Analizar el reporte forense
300
¿Probabilidad de un ataque?
302
NO
SI
Alertar al administrador y/o al sistema cliente
304
Enviar el indicador de mitigación al sistema cliente
306

FIG. 9