(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 720 759**

(51) Int. CI.:

G06F 13/14 (2006.01)
G06F 9/44 (2008.01)
H04L 12/26 (2006.01)
H04L 12/931 (2013.01)
H04L 12/813 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **06.03.2013 PCT/US2013/029222**

(87) Fecha y número de publicación internacional: **26.09.2013 WO13142041**

(96) Fecha de presentación y número de la solicitud europea: **06.03.2013 E 13764757 (4)**

(97) Fecha y número de publicación de la concesión europea: **16.01.2019 EP 2828760**

(54) Título: **Descarga de procesamiento de paquetes para virtualización de dispositivos de red**

(30) Prioridad:

21.03.2012 US 201261613824 P
17.07.2012 US 201213551064

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
24.07.2019

(73) Titular/es:

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

(72) Inventor/es:

ZUO, YUE;
FIRESTONE, DANIEL M.;
GREENBERG, ALBERT GORDON;
CHAU, HOYUEN;
DENG, YIMIN;
TUTTLE, BRYAN WILLIAM y
GARG, PANKAJ

(74) Agente/Representante:

CARPINTERO LÓPEZ, Mario

DESCRIPCIÓN

Descarga de procesamiento de paquetes para virtualización de dispositivos de red

Antecedentes

1. Antecedentes y técnica pertinente

Los sistemas informáticos y la tecnología relacionada afectan a muchos aspectos de la sociedad. De hecho, la capacidad del sistema informático para procesar la información ha transformado la forma en que vivimos y trabajamos. Los sistemas informáticos ahora realizan habitualmente una serie de tareas (por ejemplo, procesamiento de textos, programación, contabilidad, etc.) que se han llevado a cabo de forma manual antes de la aparición del sistema informático. Más recientemente, los sistemas informáticos han sido acoplados entre sí y a otros dispositivos electrónicos para formar redes informáticas tanto cableadas como inalámbricas sobre las que los sistemas informáticos y otros dispositivos electrónicos pueden transferir datos electrónicos. En consecuencia, el rendimiento de muchas tareas informáticas se distribuye a través de un número de diferentes sistemas informáticos y/o un número de diferentes entornos informáticos.

Algunos sistemas informáticos se configuran para proporcionar entornos virtualizados para alojar una o más máquinas virtuales. Por ejemplo, los entornos de ejecución para-virtualizados incluyen hipervisores. Los hipervisores proporcionan una partición primaria y una o más particiones secundarias (o máquinas virtuales). La partición primaria se configura para ejecutar un sistema operativo host y para gestionar una pila de virtualización. Cada partición secundaria se configura para ejecutar un sistema operativo invitado correspondiente. Los hipervisores proporcionan también interfaces de software que permiten que las particiones secundarias accedan a dispositivos físicos a través de dispositivos virtuales (controladores) que se ejecutan en los sistemas operativos invitados.

Un escenario común en la virtualización es la gestión de los paquetes de red entre las máquinas virtuales que se están ejecutando en un sistema informático host de virtualización, y la gestión de paquetes de red que fluyen entre las máquinas virtuales y sistemas informáticos remotos desde el sistema informático host. En ese sentido, las pilas de virtualización en sistemas operativos host pueden incluir pilas de virtualización de redes, o conmutadores virtuales. Los conmutadores virtuales se configuran para interceptar, inspeccionar y manipular paquetes de red que se comunican en relación con las máquinas virtuales. Hacer esto puede, sin embargo, ser ineficaz, puesto que hace frecuente y costoso (por ejemplo, en términos de uso de CPU) los cambios de contexto entre el sistema operativo host y los sistemas operativos invitados.

Un reciente desarrollo en la virtualización ha sido la Virtualización de E/S de una Sola Raíz (SRIOV). La SRIOV es una extensión de la arquitectura de bus de Interconexión de componentes periféricos exprés (PCIe) que permite a los dispositivos PCIe comunicarse directamente con las particiones primarias y secundarias. En ese sentido, la SRIOV permite a los dispositivos PCIe exponerse directamente a las máquinas virtuales (a través del hipervisor). Por ejemplo, una tarjeta de interfaz de red física (NIC) compatible con SRIOV puede presentar una función física con la partición host y presentar una o más funciones virtuales para particiones secundarias correspondientes. El sistema operativo host puede, a continuación, incluir un controlador de función física que se comunica con la función física, y cada sistema operativo host puede ejecutar un controlador de función virtual que se comunica con la función virtual correspondiente. La NIC física puede comunicar los paquetes de red directamente con los sistemas operativos invitados (sin pasar por el sistema operativo host), lo que puede mejorar considerablemente el rendimiento de la red.

A pesar de los avances que trae consigo la SRIOV, siguen existiendo algunas ineficiencias en el área de procesamiento de paquetes de red en entornos de virtualización.
Kaushik Ram Kumar y col.: “sNICh: Efficient Last Hop Networking in the Data Center”, Simposio ACM/IEEE sobre Arquitecturas para Redes y Sistemas de Comunicaciones (ANCS), 25 de octubre de 2010, páginas 1 a 12, presenta un concepto de un sNICh, que es una combinación de una tarjeta de interfaz de red y un acelerador de conmutación para servidores virtualizados modernos. La arquitectura sNICh explota su proximidad al servidor separando las tareas de conmutación de la red entre hardware y software de forma eficaz. Esto permite que el sNICh haga frente a la intensidad de recursos de virtualización de software y los límites de escalabilidad de soporte de hardware actual. Para conmutar un paquete entre dos máquinas virtuales invitadas a través del sNICh. El controlador de sNICh en la primera máquina virtual transmite un paquete al hardware sNICh través de su interfaz vNIC. El hardware sNICh identifica un flujo mediante el análisis de los encabezados del paquete. Una tabla de flujo es consultada para buscar una entrada de flujo coincidente. Si la búsqueda no tiene éxito, el paquete se envía a un software backend sNICh dentro de un dominio de gestión a través de una interfaz de control. Si el paquete pertenece a un nuevo flujo, el backend valida el paquete contra todas las reglas ACL. La entrada de flujo se almacena en hardware a través de la interfaz de control. El paquete se re-inyecta en el sNICh de nuevo a través de la interfaz de control. Si la búsqueda en la tabla de flujo es exitosa, se ejecuta la acción de flujo. El paquete se reenvía a la segunda máquina virtual a través de su interfaz vNIC.

El documento US 2010/333189 A1 se refiere a técnicas para hacer cumplir las políticas de seguridad en el tráfico de red. Un paquete procedente de una fuente de paquetes de paquetes se recibe por un ejecutor de regla del vínculo de datos y el ejecutor de regla del vínculo de datos obtiene una regla del vínculo de datos que se aplica a un vínculo

de datos. El vínculo de datos se conecta operativamente a la fuente de paquetes y el vínculo de datos se asocia con una dirección de control de acceso a medios. El ejecutor de regla del vínculo de datos determina si el paquete cumple con la regla del vínculo de datos y descarta el paquete cuando el paquete no cumple con la regla del vínculo de datos.

El documento US 2010/014526 A1 se refiere a un conmutador de hardware para su uso con hipervisores y servidores de cuchilla. El conmutador de hardware permite que ocurra la conmutación entre diferentes SO invitados que se ejecutan en el mismo servidor, o entre diferentes servidores en un sistema IOV multirradiculares, o entre diferentes SO invitados que se ejecutan en el mismo servidor en sistemas IOV unirradiculares.

Breve sumario

El objeto de la presente invención es mejorar los sistemas de la técnica anterior.

Este objeto se resuelve por la materia objeto de las reivindicaciones independientes.

Las realizaciones preferidas se definen por las reivindicaciones dependientes.

La presente invención se extiende a procedimientos, sistemas y productos de programas informático para la descarga de procesamiento de paquetes para la virtualización de dispositivos de redes. Por ejemplo, las realizaciones de la invención proporcionan un modelo de flujo y de regla de paquetes de red genérico que permite que una porción del procesamiento de paquetes de red en un host de máquina virtual para descargarse desde el host a una NIC física. En particular, las realizaciones de la invención permiten que la totalidad o porciones de una o más tablas de flujo en una partición primaria (es decir, el sistema operativo host) se descargue en una NIC física. Si lo hace, permite a la NIC física realizar el procesamiento de paquetes de acuerdo con reglas definidas, al tiempo que aumenta el rendimiento del procesamiento de paquetes de red en entornos de máquina virtual.

En algunas realizaciones, un procedimiento de procesamiento de paquetes de red para una máquina virtual que se ejecuta en un sistema informático incluye una partición host que mantiene uno o más conjuntos de reglas para una máquina virtual. El procedimiento incluye también una NIC física que mantiene una o más tablas de flujo para la máquina virtual. La NIC Física recibe un paquete de red asociado con la máquina virtual, y procesa el paquete de red para la máquina virtual. El procesamiento del paquete de red incluye que la NIC física compare el paquete de red con la una o más tablas de flujo. Cuando el paquete de red coincide con un flujo en la una o más tablas de flujo, la NIC física realiza una acción en el paquete de red basándose en el flujo coincidente. Como alternativa, cuando el paquete de red no coincide con un flujo en la una o más tablas de flujo, la NIC física hace pasar el paquete de red a la partición host para procesarse contra uno o más conjuntos de reglas.

En otras realizaciones, un procedimiento de procesamiento de paquetes de red para una máquina virtual que se ejecuta en el sistema informático incluye un conmutador virtual que mantiene uno o más conjuntos de reglas para una máquina virtual y que mantiene también una o más tablas de flujo para la máquina virtual. El conmutador virtual descarga a al menos una porción de la una o más tablas de flujo en la NIC física. El conmutador virtual procesa un paquete de red para la máquina virtual. El procesamiento del paquete de red incluye que el conmutador virtual reciba el paquete de red de una de la máquina virtual o la NIC física, y que el conmutador virtual haga coincidir el paquete de red con una regla en el uno o más conjuntos de reglas. Basándose en la coincidencia del paquete de red con la regla, el conmutador virtual crea un flujo en la una o más tablas de flujo y descarga el flujo en la NIC física.

Se proporciona este sumario para introducir una selección de conceptos en una forma simplificada que se describen más adelante en la Descripción Detallada. Este sumario no tiene la intención de identificar características clave o características esenciales de la materia reivindicada, ni pretende ser utilizado como una ayuda para determinar el ámbito de la materia reivindicada.

Las características y ventajas adicionales de la invención se expondrán en la siguiente descripción, y en parte serán obvias a partir de la descripción, o pueden aprenderse por la práctica de la invención. Las características y ventajas de la invención se pueden realizar y obtener por medio de los instrumentos y combinaciones particularmente señalados en las reivindicaciones adjuntas. Estas y otras características de la presente invención resultarán más evidentes a partir de la siguiente descripción y reivindicaciones adjuntas, o pueden aprenderse por la práctica de la invención como se expone más adelante.

Breve descripción de los dibujos

Para describir la forma en que se pueden obtener las ventajas y características anteriormente mencionadas y otras de la invención, se reproducirá una descripción más particular de la invención brevemente descrita anteriormente, haciendo referencia a realizaciones específicas de la misma que se ilustran en los dibujos adjuntos. Entendiendo que estos dibujos representan solo realizaciones convencionales de la invención y no debe, por tanto, considerarse que limiten su ámbito, la invención se describirá y explicará con especificidad y detalle adicionales a través del uso de los dibujos adjuntos, en los que:

La Figura 1 ilustra una arquitectura informática ilustrativa que facilita la descarga de procesamiento de paquetes

en una NIC física para la virtualización de dispositivos de redes.
La Figura 2 ilustra un diagrama de flujo de un procedimiento ilustrativo para el procesamiento de paquetes de red para una máquina virtual de ejecución en el sistema informático.
La Figura 3 ilustra un diagrama de flujo de un procedimiento de ejemplo alternativo para el procesamiento de paquetes de red para una máquina virtual que se ejecuta en un sistema informático.
La Figura 4 ilustra una arquitectura informática alternativa que facilita la descarga de procesamiento de paquetes en una NIC física para la virtualización de dispositivos de redes.
La Figura 5 ilustra una arquitectura informática ilustrativa que incluye capas de un conmutador virtual multi-capa ilustrativo.

Descripción detallada

La presente invención se extiende a procedimientos, sistemas y productos de programa informático para la descarga de procesamiento de paquetes para la virtualización de dispositivos de redes. Por ejemplo, las realizaciones de la invención proporcionan un modelo de flujo y de regla de paquetes de red genérico que permite que una porción del procesamiento de paquetes de red en un host de máquina virtual se descargue desde el host en una NIC física. En particular, las realizaciones de la invención permiten que la totalidad o porciones de una o más tablas de flujo en una partición primaria (es decir, el sistema operativo host) se descarguen a una NIC física. Hacerlo permite a la NIC física realizar el procesamiento de paquetes de acuerdo con reglas definidas, al tiempo que aumenta el rendimiento del procesamiento de paquetes de red en entornos de máquina virtual.

En algunas realizaciones, un procedimiento de procesamiento de paquetes de red para una máquina virtual que se ejecuta en un sistema informático incluye una partición host que mantiene uno o más conjuntos de reglas para una máquina virtual. El procedimiento también incluye una NIC física que mantiene una o más tablas de flujo para la máquina virtual. La NIC Física recibe un paquete de red asociado con la máquina virtual, y procesa el paquete de red para la máquina virtual. El procesamiento del paquete de red incluye que la NIC física compare el paquete de red con la una o más tablas de flujo. Cuando el paquete de red coincide con un flujo en la una o más tablas de flujo, la NIC física realiza una acción en el paquete de red basándose en el flujo coincidente. Como alternativa, cuando el paquete de red no coincide con un flujo en una o más tablas de flujo, la NIC física hace pasar el paquete de red a la partición host para procesarse contra uno o más conjuntos de reglas.

En otras realizaciones, un procedimiento de procesamiento de paquetes de red para una máquina virtual que se ejecuta en el sistema informático incluye un conmutador virtual que mantiene uno o más conjuntos de reglas para una máquina virtual y que mantiene también una o más tablas de flujo para la máquina virtual. El conmutador virtual descarga a al menos una porción de la una o más tablas de flujo en la NIC física. El conmutador virtual procesa un paquete de red para la máquina virtual. El procesamiento del paquete de red incluye que el conmutador virtual reciba el paquete de red de una de la máquina virtual o la NIC física, y que el conmutador virtual haga coincidir el paquete de red con una regla en el uno o más conjuntos de reglas. Basándose en la coincidencia del paquete de red con la regla, el conmutador virtual crea un flujo en la una o más tablas de flujo y descarga el flujo en la NIC física.

Las realizaciones de la presente invención pueden comprender o utilizar un ordenador de propósito especial o de propósito general que incluye el hardware informático, tal como, por ejemplo, uno o más procesadores y memoria de sistema, como se describe en mayor detalle a continuación. Las realizaciones dentro del ámbito de la presente invención incluyen también medios físicos y otros legibles por ordenador para transportar o almacenar instrucciones ejecutables por ordenador y/o estructuras de datos. Tales medios legibles por ordenador pueden ser cualquier medio disponible al que se puede acceder por un sistema informático de propósito general o de propósito especial. Los medios legibles por ordenador que almacenan instrucciones ejecutables por ordenador son medios (dispositivos) de almacenamiento informático. Los medios legibles por ordenador que llevan instrucciones ejecutables por ordenador son medios de transmisión. Por tanto, a modo de ejemplo, y sin limitación, las realizaciones de la invención pueden comprender al menos dos tipos distintos de medios legibles por ordenador: medios (dispositivos) de almacenamiento informático y medios de transmisión.

Los medios (dispositivos) de almacenamiento informático incluyen RAM, ROM, EEPROM, CD-ROM, unidades de estado sólido (“SSD”) (por ejemplo, basadas en RAM), memoria flash, memoria de cambio de fase (“PCM”), otros tipos de memoria, otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, o cualquier otro medio que pueda utilizarse para almacenar unos medios de código de programa deseados en forma de instrucciones o estructuras de datos ejecutables por ordenador y a los que se puede acceder mediante un ordenador de propósito general o de propósito especial.

Una “red” se define como uno o más vínculos de datos que permiten el transporte de datos electrónicos entre sistemas informáticos y/o módulos y/o en otros dispositivos electrónicos. Cuando la información se transfiere o proporciona a través de una red u otra conexión de comunicaciones (ya sea cableada, inalámbrica o una combinación de tecnología cableada o inalámbrica) a un ordenador, el ordenador considera propiamente la conexión como un medio de transmisión. Los medios de transmisión pueden incluir una red y/o vínculos de datos que pueden utilizarse para llevar unos medios de código de programa deseados en forma de instrucciones o estructuras de datos ejecutables por ordenador y al que puede accederse mediante un ordenador de propósito general o de propósito especial. Combinaciones de los anteriores deberían también incluirse dentro del ámbito de los medios legibles por

ordenador.

Además, tras alcanzar diversos componentes del sistema informático, los medios de código de programa en forma de instrucciones o estructuras de datos ejecutables por ordenador pueden transferirse automáticamente desde los medios de transmisión a los medios (dispositivos) de almacenamiento informático (o viceversa). Por ejemplo, las instrucciones o estructuras de datos ejecutables por ordenador recibidas a través de una red o vínculos de datos pueden tamponarse en la RAM dentro de un módulo de interfaz de red (por ejemplo, una “NIC”), y eventualmente transferirse, después, a la RAM del sistema informático y/o a medios (dispositivos) de almacenamiento informático menos volátiles en un sistema informático. Por lo tanto, se debe entender que los medios (dispositivos) de almacenamiento informático se pueden incluir en los componentes del sistema informático que utilizan también (o incluso principalmente) medios de transmisión.

Las instrucciones ejecutables por ordenador comprenden, por ejemplo, instrucciones y datos que, cuando se ejecutan en un procesador, hacen que un ordenador de propósito general, ordenador de propósito especial, o dispositivo de procesamiento de propósito especial, realice una cierta función o grupo de funciones. Las instrucciones ejecutables por ordenador pueden, por ejemplo, ser binarias, instrucciones de formato intermedio como lenguaje ensamblador, o incluso código fuente. Aunque la materia objeto se ha descrito en un lenguaje específico para las características estructurales y/o actos metodológicos, se debe entender que la materia objeto definida en las reivindicaciones adjuntas no se limita necesariamente a las características o actos descritos, que se han descrito anteriormente. Más bien, las características y actos descritos se divulgan como formas ilustrativas de implementación de las reivindicaciones.

Los expertos en la materia apreciarán que la invención puede ponerse en práctica en entornos informáticos en red con muchos tipos de configuraciones de sistemas informáticos, incluyendo, ordenadores personales, ordenadores de sobremesa, ordenadores portátiles, procesadores de mensajes, dispositivos de mano, sistemas multi-procesador, electrónica de consumo programable o basada en microprocesadores, PC en red, miniordenadores, ordenadores centrales, teléfonos móviles, PDA, tabletas, buscapersonas, enrutadores, conmutadores y similares. La invención puede también ponerse en práctica en entornos de sistemas distribuidos en los sistemas informáticos locales y remotos, que están vinculados (ya sea por vínculos de datos cableados, vínculos de datos inalámbricos, o por una combinación de vínculos de datos cableados e inalámbricos) a través de una red, ambos realizan tareas. En un entorno de sistema distribuido, se pueden colocar módulos de programa en dispositivos de almacenamiento en memoria tanto locales como remotos. En algunas realizaciones, la invención puede ponerse en práctica en relación con NIC físicas compatibles con SRIOV, sin embargo, el ámbito de la invención se extiende más allá de la SRIOV.

Las realizaciones de la invención operan en conexión con un host (por ejemplo, una partición raíz) que ejecuta una o más máquinas virtuales. El host incluye un conmutador virtual que realiza el procesamiento de paquetes (por ejemplo, la inspección y, posiblemente, la manipulación) de los paquetes de red que se envían y/o reciben por las máquinas virtuales. Por ejemplo, las realizaciones de la invención pueden procesar paquetes de Protocolo de Internet (IP), paquetes RDMA sobre Eternet Convergente (RoCE), paquetes de Canal de Fibra sobre Eternet (FCoE), etc. Además, las realizaciones de la invención proporcionan un modelo de regla y de flujo genérico que permite que al menos una porción del procesamiento de paquetes sea descargada desde el host en una NIC física, tal como una NIC de Ethernet, una NIC InfiniBand, u otro tipo de tejido físico. Las realizaciones de la invención permiten, por tanto, el procesamiento de paquetes de forma genérica, eliminando la necesidad de desarrollar diferentes módulos de conmutación virtual para diferentes tipos de procesamiento de paquetes.

En particular, las realizaciones incluyen la descarga de una o más tablas de flujo (o porciones de las mismas) en una NIC física (tal como una NIC física compatible con SRIOV). En ese sentido, un puente virtual en la NIC física se habilita para llevar a cabo el procesamiento de paquetes, similar al conmutador virtual en el host. Por ejemplo, si un paquete se recibe en la NIC física, el puente virtual puede hacer coincidir el paquete con un flujo descargado. El puente virtual en la NIC física puede, a continuación, adoptar la acción apropiada para el flujo sin la participación del host. Si lo hace, elimina las ineficiencias asociadas con tener que hacer todo el procesamiento de paquetes de reglas/flujos en el host.

Haciendo referencia a continuación a las Figuras, la Figura 1 ilustra una arquitectura 100 informática ilustrativa que facilita la descarga de procesamiento de paquetes en una NIC física para la virtualización de dispositivos de redes. Como se ilustra, la arquitectura 100 informática incluye el host 102, la máquina 108 virtual y la NIC 110 física.

El host 102 proporciona un entorno de virtualización. Por ejemplo, el host 102 puede incluir una partición primaria (que ejecuta un sistema operativo host) y una o más particiones secundarias. Cada partición secundaria puede ser vista como proporcionando un entorno de hardware virtualizado para la ejecución de una máquina virtual correspondiente, tal como la máquina 108 virtual. En algunas realizaciones, el host 102 utiliza una parte de un entorno de computación en la nube que proporciona máquinas virtuales a los inquilinos.

Cada máquina virtual (incluyendo la máquina 108 virtual) ejecuta una o más aplicaciones virtualizadas, tales como un sistema operativo, software de aplicación, etc. Como se representa, la máquina 108 virtual incluye una pila 108a de red (por ejemplo, una pila TCP/IP), un controlador 108b de NIC virtual, y un controlador 108c de función virtual. Al utilizar la pila 108a de red, el controlador 108b de NIC virtual, y el controlador 108c de función virtual, la máquina 108

virtual es capaz de enviar y/o recibir paquetes de red y otra información a través del host 102 sobre un bus 116 virtual y/o a través de la NIC 110 física sobre la trayectoria 114 de datos.

La NIC 110 física comprende hardware físico que es capaz de virtualizarse y que se conecta a otros sistemas informáticos y/o redes utilizando una o más interfaces externas (por ejemplo, la interfaz 126 externa representado). Aunque solo se representa una NIC física, la arquitectura informática puede incluir cualquier número de NIC físicas. La NIC 110 física incluye un puente 112 virtual. El puente 112 virtual une funciones virtuales y las funciones físicas en la NIC 110 física y lleva a cabo la inspección y la manipulación de paquete. El puente 112 virtual trabaja con un conmutador 104 virtual en el host 102 para regular el tráfico de red, como se describe en mayor detalle más adelante. En ese sentido, la NIC 110 física puede exponer una o más funciones virtuales a una o más máquinas virtuales que están siendo alojadas en el host 102. Además, la NIC 110 física puede exponer una o más funciones físicas al host 102.

Por ejemplo, la Figura 1 representa que la NIC 110 física presenta la función 122 física al host 102. La Figura 1 representa también que host 102 incluye un controlador 124 de función física correspondiente, y que la trayectoria 118 de datos conecta la función 122 física en la NIC 110 física y el controlador 124 de función física en el host 102. En ese sentido, la función 122 física y el controlador 124 de función física pueden funcionar para el intercambio de paquetes de red entre la NIC 110 física y el host 102. Por ejemplo, el controlador 124 de función física puede comunicarse con conmutador 104 virtual en el host 102, y la función 122 física puede comunicarse con el puente 112 virtual en la NIC 110 física.

La Figura 1 representa también que la NIC 110 física presenta la función 120 virtual a la máquina 108 virtual, que corresponde con controlador 108c de función virtual. La trayectoria 114 de datos conecta la función 120 virtual en la NIC 110 física y el controlador 108c de función virtual en la máquina 108 virtual. La NIC 110 física puede presentar más de una función virtual a la máquina 108 virtual, y/o puede presentar funciones virtuales adicionales a las máquinas virtuales adicionales. Por lo general, cada máquina virtual puede acceder directamente a una función virtual asignada. Por ejemplo, una máquina virtual puede utilizar su controlador de función virtual para comunicar los paquetes de red con una función virtual asignada a la NIC 110 física sin la intervención del host 102. Si lo hace, puede reducir el uso del procesador y la latencia de red. Por ejemplo, la máquina 108 virtual y la NIC 110 física pueden comunicarse directamente mediante la función 120 virtual y el controlador 108c de función virtual a través de la trayectoria 114 de datos.

Como se ha indicado anteriormente, la NIC 110 física puede, en algunas realizaciones, comprender hardware PCIe que es compatible con SRIOV. En tales realizaciones, una o más de la función 120 virtual o función 122 física pueden comprender funciones PCIe. Sin embargo, se apreciará que los principios descritos en la presente memoria pueden aplicarse a una variedad de dispositivos de hardware, y no se limitan a dispositivos compatibles con SRIOV o a dispositivos PCIe.

En algunas realizaciones, una o más máquinas virtuales alojadas en el host 102 se pueden asociar con reglas (de entradas y/o salientes) y posiblemente flujos (de entradas y/o salientes) de acuerdo con un modelo de regla/flujo genérico. Como se representa, el host 102 incluye un conmutador 104virtual. El conmutador 104 virtual se configura para inspeccionar y manipular paquetes de red que se envían de y reciben por cualquier máquina virtual alojada de acuerdo con el modelo de regla/flujo genérico. Por ejemplo, basándose en las reglas y flujos definidos, el conmutador 104 virtual puede permitir los paquetes, bloquear paquetes, volver a enrutar paquetes, realizar NAT, o realizar cualquier otra inspección/manipulación de paquete apropiada para las tecnologías y dispositivos de redes que se estén utilizando.

Como se utiliza en la presente memoria, una regla define una política de flujo de paquetes (o una porción de la misma) basándose en una o más condiciones de la regla y una o más acciones de regla. En algunas realizaciones, las reglas son específicas de una máquina virtual particular. Las reglas se pueden definir por los administradores, o se pueden definir por los sistemas de nivel superior. En algunas realizaciones, las reglas son estáticas, o relativamente estáticas. En algunas realizaciones, las reglas se almacenan en conjuntos de reglas y se configuran para su coincidencia lineal.

Las condiciones de regla se pueden definir utilizando tuplas, que incluyen campos y valores coincidentes. Las tuplas pueden comprender cualquier combinación de campos apropiados para el uno o más protocolos de red y el uno o más dispositivos de hardware en uso. Las tuplas pueden incluir, por ejemplo, la dirección de red de origen y/o destino (por ejemplo, dirección IP, cuando se utiliza IP), el puerto de origen y/o destino, el protocolo (por ejemplo, Protocolo de Control de Transmisión (TCP), Protocolo de datagramas de usuario (UDP)), la dirección de hardware de origen y/o destino (por ejemplo, dirección MAC de Ethernet), o combinaciones de los mismos. Por ejemplo, una condición de regla ilustrativa puede definirse de acuerdo con una de cinco tuplas como ‘192.168.0. *, *, *, *, TCP’, que se correspondería con cualquier paquete de red en la red 192.168.0.* que tiene cualquier dirección IP de origen, cualquier puerto de origen, cualquier dirección IP de destino, cualquier puerto de destino, y que utiliza el protocolo TCP. En algunas realizaciones, las tuplas pueden referirse no solo al flujo, sino también a la condición del paquete. Por ejemplo, las tuplas pueden incluir campos relacionados con el tipo de servicio IP (ToS). Una persona con experiencia ordinaria en la materia reconocerá que otras tuplas son también posibles, incluyendo tuplas relacionadas con tecnologías de redes aún por desarrollar.

Las acciones de regla pueden comprender cualquier operación de enrutamiento y/o manipulación de paquete apropiada. Por ejemplo, algunas acciones de regla ilustrativas pueden incluir negar, permitir, la traducción de direcciones de red (NAT), mapear, medir, desencapsular, encapsular, etc. Un experto en la materia reconocerá que una variedad de otras acciones de regla es posible, incluyendo acciones relacionadas con tecnologías de red aún por desarrollar.

Las reglas se pueden utilizar para definir un rico conjunto de políticas de procesamiento de paquetes. Por ejemplo, al utilizar condiciones de regla (tuplas) y acciones de regla, una regla puede especificar que los paquetes UDP de una dirección IP particular están permitidos. En otro ejemplo, una regla puede especificar que los paquetes TCP enviados a cualquier destino con un puerto especificado están sujetos a NAT. Combinando el ejemplo de cinco tuplas anterior ilustrativo con una acción de 'permitir', una regla ilustrativa se puede definir como 'permitir 192.168.0. *, *, *, *, TCP' lo que significa que cualquier paquete de red en la red 192.168.0.* tiene cualquier dirección IP de origen, cualquier puerto de origen, cualquier dirección IP de destino, cualquier puerto de destino, y se debería permitir la utilización del protocolo TCP.

Como se utiliza en la presente memoria, un flujo es un estado dinámico que se crea basándose en reglas. Por ejemplo, cuando un paquete de red coincide con una regla, un flujo puede crearse basándose en la regla. En ese sentido, similar a las reglas, los flujos pueden también definirse en términos de condiciones (tuplas) y acciones. Los flujos almacenan un contexto sobre conexiones de red, y se pueden utilizar para determinar cómo manejar un paquete actual en una corriente o un contexto basándose en los paquetes anteriores en la corriente o contexto. Los flujos pueden someterse a un tiempo de espera. En algunas realizaciones, los flujos se almacenan en una o más tablas de flujo, tal como una tabla de flujo de entrada y/o una tabla de flujo de salida. Por ejemplo, cuando un paquete de red coincide con la regla 'permitir 192.168.0. *, *, *, *, TCP' ilustrativa, un flujo coincidente se puede crear en una o más tablas de flujo apropiadas. En algunas realizaciones, los flujos se indexan basándose en tuplas de un flujo (por ejemplo, utilizando uno o más hashes.

A lo largo de estas líneas, la Figura 1 representa que el conmutador virtual incluye los estados 106 de la máquina 108 virtual, que pueden incluir diversos tipos de estados, tales como el conjunto 106a de reglas de salida representado, conjunto 106b de reglas de entrada, la tabla 106c de flujo de salida, y la tabla 106d de flujo de entrada. El Conjunto 106a de reglas de salida define una o más reglas que se aplican a los paquetes que se envían por la máquina 108 virtual y el conjunto 106b de reglas de entrada define una o más reglas que se aplican a los paquetes que se reciben en nombre de la máquina 108 virtual. Cuando un paquete coincide con una regla de un conjunto de reglas correspondiente, un flujo puede crearse en la tabla 106c de flujo de salida y/o tabla 106d de flujo de entrada. Se apreciará que en algunas circunstancias los estados 106 pueden incluir un subconjunto de los estados representados.

Como un ejemplo, cuando el conmutador 104 virtual recibe un paquete de red que está asociado con la máquina 108 virtual (por ejemplo, de la NIC 110 física o de una máquina virtual alojada) que no coincide con un flujo en una tabla (106c, 106d) de flujo apropiada, el conmutador 104 virtual puede examinar el conjunto de reglas apropiado (es decir, el conjunto 106b de reglas de entrada para un paquete que se recibe en nombre de la máquina 108 virtual o el conjunto 106a de reglas de salida para un paquete a enviarse por la máquina 108 virtual) para una regla coincidente. Si el conmutador 104 virtual encuentra una regla coincidente, el conmutador 104 virtual puede adoptar una acción apropiada en el paquete como se define por la regla (por ejemplo, permitir/bloquear/NAT, etc.).

Si el conmutador 104 virtual encuentra una regla coincidente, el conmutador 104 virtual puede crear también un flujo (o un par de flujos) en la tabla 106c de flujo de salida y/o en la tabla 106d de flujo de entrada para su uso en el procesamiento de los paquetes subsiguientes en la corriente/contexto. Por ejemplo, cuando el paquete coincide con una regla el en conjunto 106a de reglas de salida, el conmutador 104 virtual puede crear un flujo en tabla 106c de flujo de salida y/o tabla 106d de flujo de entrada (como se representa por las flechas que conectan el conjunto 106a de reglas de salida y la tablas 106c, 106d de flujo). Como alternativa, cuando el paquete coincide con una regla en el conjunto 106b de reglas de entrada, el conmutador 104 virtual puede crear un flujo en la tabla 106c de flujo de salida y/o en la tabla 106d de flujo de entrada (como se representa por las flechas entre el conjunto 106b de reglas de entrada y la tablas 106c, 106d de flujo). Se apreciará que mediante la creación de flujos en la tabla de flujo de dirección opuesta, la conmutación virtual puede implementar Firewalls con estado.

El conmutador 104 virtual puede también descargar uno o más estados de flujo en la memoria 112a caché de flujo de salida y/o la memoria 112b caché de flujo de entrada en el puente 112 virtual de la NIC 110 física, tal como se representa por la flecha discontinua entre tabla 106c de flujo de salida y la memoria 112a caché de flujo de salida y la flecha discontinua entre la tabla 106d de flujo de entrada de la memoria 112b caché de flujo de entrada. Por ejemplo, el conmutador 104 virtual puede enviar una o más solicitudes a través de la trayectoria 118 de datos a la NIC 100 física solicitando la creación de los flujos en las memorias 112a caché, 112b de flujo. En algunas circunstancias, la descarga del estado de flujo en la NIC 110 física permite que el puente 112 virtual realice el procesamiento de paquetes separado de un conmutador 104 virtual, reduciendo de este modo el uso del procesador al host 102. Por ejemplo, después de descargar un flujo en la NIC 110 física, la NIC 110 física puede recibir un paquete posterior de la misma corriente (por ejemplo, desde la máquina 108 virtual sobre la trayectoria 114 de datos o desde otro sistema informático a través de la interfaz 126 externa). En esta circunstancia, el puente 112 virtual puede hacer coincidir el paquete subsiguiente con un estado de flujo en la memoria 112a caché, 112b de flujo

apropiada, y llevar a cabo la acción definida en el propio flujo, sin necesidad de enviar en primer lugar el paquete al conmutador 104 virtual.

Utilizando la configuración anterior, la máquina 108 virtual puede utilizar el controlador 108c de función virtual para enviar un paquete de red de salida a la función 120 virtual de la NIC 110 física sobre la trayectoria 114 de datos. Al recibir el paquete de red, el puente 112 virtual busca la memoria 112a caché de flujo de salida para un flujo coincidente. Si el puente 112 virtual encuentra un flujo coincidente en la memoria 112a caché de flujo de salida, el puente 112 virtual adopta la acción definida en el flujo. Por ejemplo, el puente 112 virtual puede realizar una operación de manipulación de paquete y/o puede reenviar el paquete de red a una máquina virtual de destino o a otro sistema informático a través de la interfaz 126 externa.

De lo contrario, si el puente 112 virtual no encuentra un flujo coincidente en la memoria 112a caché de flujo de salida, pueden ocurrir dos acciones alternativas. En una primera realización, el puente 112 virtual rechaza el paquete de red a la máquina 108 virtual (por ejemplo, sobre la trayectoria 114 de datos). La máquina 108 virtual reenvía el paquete de red al conmutador 104 virtual a través del bus 116 virtual. En una segunda realización, el puente 112 virtual utiliza la función 122 física para enviar el paquete de red al controlador 124 de función física a través de la trayectoria 118 de datos. El controlador 124 de función física enruta, a su vez, el paquete de red hacia el conmutador 104 virtual. En cualquier realización, después de que el conmutador 104 virtual recibe el paquete de red, el conmutador 104 virtual intenta hacer coincidir el paquete de red con un flujo en tabla 106c de flujo de salida. Si el paquete de red no coincide con un flujo en la tabla 106c de flujo de salida, el conmutador 104 virtual intenta hacer coincidir el paquete de red con una regla del conjunto 106a de reglas de salida. Si se encuentra una regla coincidente en el conjunto 106a de reglas de salida, el conmutador 104 virtual adopta una acción apropiada (por ejemplo, permitir/bloquear/NAT, etc.) como se define por la regla coincidente, y se puede crear uno o más flujos en una o ambas tablas 106c/106d de flujo y potencialmente en una o ambas memorias 112a caché/112b de flujo.

También utilizando la configuración anterior, la NIC 110 física puede recibir un paquete de red de entrada en nombre de la máquina 108 virtual (por ejemplo, desde otra máquina virtual a través de una función virtual correspondiente o desde otro sistema informático a través de la interfaz 126 externa). Una vez recibido el paquete de red, el puente 112 virtual busca la memoria 112b caché de flujo de entrada para un flujo coincidente. Si el puente 112 virtual encuentra un flujo coincidente en la memoria 112b caché de flujo de entrada, el puente 112 virtual adopta la acción apropiada (por ejemplo, permitir/bloquear/NAT, etc.) definida en el flujo. Por ejemplo, el puente 112 virtual puede utilizar la función 120 virtual y la trayectoria 114 de datos para enviar el paquete al controlador 108c de función virtual en la máquina 108 virtual. Si el puente 112 virtual no encuentra un flujo coincidente en la memoria 112b caché de flujo de entrada, el puente 112 virtual reenvía el paquete al conmutador 104 virtual en el host 102 utilizando la función 122 física y la trayectoria 118 de datos o la función 120 virtual y la trayectoria 114 de datos. El conmutador 104 virtual procesa el paquete como se ha descrito anteriormente en el contexto de un paquete de red de salida.

Se apreciará que la memoria 112a caché de flujo de salida y la memoria 112b caché de flujo de entrada pueden representar solo una porción o un subconjunto de las tablas de flujo total (es decir, la tabla 106c de flujo de salida y la tabla 106d de flujo de entrada). Por ejemplo, la NIC 110 física puede tener una memoria limitada debido a los costes u otras restricciones de diseño. En ese sentido, el almacenamiento de solo una porción de las tablas 106c/106d de flujo en memorias cachés 112a/112b de flujo disminuye la cantidad de memoria necesaria para descargar tablas de flujo en la NIC 110 física. Debido a que la memoria 112a caché de flujo de salida y la memoria 112a caché de flujo de entrada no pueden incluir todos los datos de estado de flujo un fallo de memoria caché puede ocurrir cuando se procesa un paquete en el puente 112 virtual. Cuando se produce un fallo en la memoria caché, el puente 112 virtual reenvía el paquete al conmutador 104 virtual para su procesamiento adicional. Se apreciará que diferentes tipos de políticas de sustitución/refrescamiento de memorias caché se pueden emplear. Por ejemplo, el estado de flujo se puede colocar en la NIC 110 física después de que se produce un fallo en la memoria caché, las entradas pueden caducar de la NIC 110 física después de una cantidad predefinida de inactividad, etc.

Además, en algunas realizaciones, solo ciertos tipos de flujos se almacenan en la NIC 110 física. Por ejemplo, el puente 112 virtual puede soportar la realización de solo tipos limitados de operaciones/acciones. En ese sentido, solo los flujos relacionados con operaciones/acciones soportadas por un puente 112 virtual se pueden almacenar en la NIC 110 física. En estas realizaciones, cualquier operación/acción adicional se maneja en el conmutador 104 virtual.

La Figura 2 ilustra un diagrama de flujo de un procedimiento 200 ilustrativo para el procesamiento de paquetes de red para una máquina virtual que se ejecuta en el sistema informático. El procedimiento 200 se describirá con respecto a los componentes y los datos de la arquitectura 100 informática.

El procedimiento 200 incluye un acto de un conmutador virtual en una partición host de mantener uno o más conjuntos de reglas para una máquina virtual (acto 202). Por ejemplo, el conmutador 104 virtual en el host 102 puede mantener los estados 106 de la máquina 108 virtual. Los estados 106 pueden incluir uno o ambos del conjunto 106a de reglas de salida para la máquina 108 virtual o el conjunto 106b de reglas de entrada para la máquina 108 virtual. Los estados 106 pueden incluir también una o más tablas de flujo de la máquina 108 virtual, como la tabla 106c de flujo de salida y la tabla 106d de flujo de entrada. Aunque no se representa, el conmutador 104 virtual puede almacenar los estados (por ejemplo, conjuntos de reglas, tablas de flujo, etc.) para otras máquinas

virtuales adicionales.

El procedimiento 200 incluye también un acto de una NIC física de mantener una o más tablas de flujo para la máquina virtual (acto 204). Por ejemplo, la NIC 110 física puede almacenar la memoria 112a caché de flujo de salida y/o la memoria 112b caché de flujo de entrada para la máquina virtual 208. Las memorias cachés de flujo pueden incluir todas, o solo una parte de, las tablas de flujo en el host 102. Aunque no se representa, la NIC 110 física puede almacenar tablas de flujo para otras máquinas virtuales adicionales.

El procedimiento 200 incluye también un acto de la NIC física de recibir un paquete de red asociado con la máquina virtual (acto 206). Por ejemplo, la NIC física puede recibir un paquete de red desde otro sistema informático a través de la interfaz 126 externa, puede recibir un paquete de red desde la máquina 108 virtual a través de la función 120 virtual, o puede recibir un paquete de red desde otra máquina virtual en el host 220 a través de otra función virtual que está asociada con esa máquina virtual.

El procedimiento 200 incluye también un acto de procesar un paquete de red para la máquina virtual (acto 208). Por ejemplo, el puente 112 virtual puede procesar un paquete de red recibido desde la máquina 108 virtual o recibido en nombre de la máquina 108 virtual (es decir, un paquete de red que es recibido por la máquina 108 virtual).

El acto 208 incluye un acto de la NIC física de comparar el paquete de red con la una o más tablas de flujo (acto 210). Por ejemplo, el puente 112 virtual puede comparar el paquete de red con la memoria 112a caché de flujo de salida si el paquete está siendo enviado por la máquina 108 virtual, o el puente 112 virtual puede comparar el paquete de red con la memoria 112b caché de flujo de entrada si el paquete se recibe en nombre de la máquina 108 virtual.

El acto 208 incluye también, cuando el paquete de red coincide con un flujo en la una o más tablas de flujo, un acto de la NIC física de realizar una acción sobre el paquete de red basándose en el flujo coincidente (acto 212). Por ejemplo, si el paquete de red coincide con un flujo en la memoria 112a caché de flujo de salida o en la memoria 112b caché de flujo de entrada, el puente 112 virtual puede realizar una acción especificada en el flujo (por ejemplo, permitir, rechazar, NAT, etc.).

El acto 208 incluye también, cuando el paquete de red no coincide con un flujo en la una o más tablas de flujo, un acto de la NIC física de hacer pasar el paquete de red al host para su procesamiento frente a uno o más conjuntos de reglas (Acto 214). Por ejemplo, si el paquete de red no coincide con un flujo en la memoria 112a caché de flujo de salida o en la memoria 112b caché de flujo de entrada, el puente 112 virtual puede enviar el paquete al conmutador 104 virtual en el host 102 para su procesamiento adicional. En algunas realizaciones, el puente 112 virtual envía el paquete de red directamente al host 102 utilizando la función 122 física y la trayectoria 118 de datos. En otras realizaciones, el puente 112 virtual envía el paquete de red indirectamente al host 102 utilizando la función 120 virtual y la trayectoria 114 de datos (es decir, a través de la máquina 108 virtual y a través del bus 116 virtual).

Cuando se recibe, el host 102 puede hacer pasar el paquete de red al conmutador 104 virtual. El conmutador 104 virtual puede, a su vez, comparar el paquete de red con el estado 106 (es decir, tablas de flujo, conjuntos de reglas) y adoptar cualquier acción apropiada. Por ejemplo, si el paquete de red coincide con un flujo en el host 102, el conmutador 104 virtual puede adoptar la acción apropiada (por ejemplo, permitir, rechazar, NAT, etc.) y, potencialmente, actualizar una memoria caché de flujo en la NIC 110 física. Si el paquete de red no coincide con un flujo en el host 102 (o si no existe flujo apropiado) el conmutador 104 virtual puede comparar el paquete de red con un conjunto de reglas apropiado, adoptar una acción apropiada como se especifica en cualquier regla coincidente, y, potencialmente, crear uno o más flujos nuevos (por ejemplo, en los estados 206 y la NIC 110 física).

La Figura 3 ilustra un diagrama de flujo de un procedimiento 300 ilustrativo adicional para el procesamiento de paquetes de red para una máquina virtual que se ejecuta en un sistema informático. El procedimiento 300 se describirá con respecto a los componentes y los datos de la arquitectura 100 informática.

El procedimiento 300 incluye un acto de un acto de un conmutador virtual en una partición host de mantener uno o más conjuntos de reglas para una máquina virtual (acto 302). Por ejemplo, el conmutador 104 virtual en el host 102 puede mantener los estados 106 de la máquina 108 virtual. Los estados 106 pueden incluir uno o ambos del conjunto 106a de reglas de salida para la máquina 108 virtual o el conjunto 106b de reglas de entrada para la máquina 108 virtual.

El procedimiento 300 incluye también un acto de un acto del conmutador virtual de mantener una o más tablas de flujo para la máquina virtual (acto 304). Por ejemplo, los estados 106 pueden incluir uno o ambas de la tabla 106c de flujo de salida para la máquina 108 virtual o de la tabla 106d de flujo de entrada para la máquina 108 virtual.

El procedimiento 300 incluye también un acto del conmutador virtual de descargar al menos una porción de la una o más tablas de flujo en la NIC física (acto 306). Por ejemplo, el conmutador 104 virtual puede descargar uno o más flujos de la tabla 106c de flujo de salida en la memoria 112a caché de flujo de salida. Además o como alternativa, el conmutador 104 virtual puede descargar uno o más flujos de la tabla 106d de flujo de entrada en la memoria 112b caché de flujo de entrada.

El procedimiento 300 incluye también un acto del conmutador virtual de procesar un paquete de red para la máquina virtual (acto 308). Por ejemplo, el conmutador 104 virtual puede procesar un paquete de red recibido de o en nombre de la máquina 108 virtual.

El acto 308 incluye un acto del conmutador virtual de recibir el paquete de red de una de la máquina virtual o la NIC física (acto 310). Por ejemplo, el conmutador 104 virtual puede recibir el paquete de red, ya sea de la máquina 108 virtual a través del bus 216 virtual, o de la NIC 110 física a través de la trayectoria 118 de datos (y el controlador 124 de función física).

El acto 308 incluye también un acto del conmutador virtual de hacer coincidir el paquete de red con una regla en el uno o más conjuntos de reglas (acto 312). Por ejemplo, si el paquete de red está siendo enviado por la máquina 108 virtual, el conmutador 104 virtual puede hacer coincidir el paquete contra el conjunto 106a de reglas de salida. Como alternativa, si el paquete de red se recibe en nombre de la máquina 108 virtual, el conmutador 104 virtual puede hacer coincidir el paquete contra el conjunto 106b de reglas de entrada.

El acto 308 incluye también, basándose en la coincidencia del paquete de red con la regla, un acto del conmutador virtual de crear un flujo en la una o más tablas de flujo (acto 314). Por ejemplo, después de hacer coincidir el paquete de red en contra de una regla en un conjunto 106a de reglas de salida o conjunto 106b de reglas de entrada, el conmutador virtual puede crear uno o más flujos basándose en la regla en tabla 106c de flujo de salida y/o tabla 106d de flujo de entrada.

El acto 308 incluye también, basándose en la coincidencia del paquete de red con la regla, un acto del conmutador virtual de descargar el flujo en la NIC física (acto 316). Por ejemplo, basándose en la regla coincidente, el conmutador 104 virtual puede descargar el flujo de la memoria 112a caché de flujo de salida y/o de la memoria 112b caché de flujo de entrada.

La Figura 4 ilustra una arquitectura 400 informática alternativa que facilita la descarga del procesamiento de paquetes en una NIC física para la virtualización de dispositivos de redes, y que proporciona una o más posibles optimizaciones sobre la arquitectura 100 informática. En algunas realizaciones, la arquitectura 400 informática se puede combinar con la arquitectura 100 informática. Como se ilustra, la arquitectura 400 informática incluye componentes que son análogos a los componentes de la arquitectura 100 informática, como el host 402, la máquina 408 virtual y la NIC 410 física. En la arquitectura 400 informática, sin embargo, el controlador 408b de NIC virtual en la máquina 408 virtual incluye la lista 408d de flujo de salida. Por lo tanto, el controlador 408b de NIC virtual mantiene información acerca de algunos o todos los flujos de salida. En ese sentido, incluso antes de enviar un paquete de red a la función 420 virtual con el controlador 408c función virtual, el controlador 408b de NIC virtual puede determinar si el paquete coincide con una red de flujo de salida basándose en la lista 408d de flujo de salida. Si se encuentra una coincidencia, entonces el paquete de red se puede hacer coincidir también con un flujo en la memoria caché 412a de flujo de salida (si el flujo se ha descargado en la NIC 410 física). Cuando el paquete no coincide con un flujo basándose en la lista 408d, 408b de flujo de salida virtual, el controlador de NIC puede reenviar el paquete directamente al conmutador 404 virtual, sin necesidad de enviar el paquete a la primera NIC 410 física.

En algunas situaciones, la máquina 408 virtual puede ser una entidad no fiable. Por lo tanto, si el paquete en última instancia, puede o no enviarse a un destino, está todavía determinado por el puente 412 virtual en la NIC 410 física y/o conmutador 404 virtual. Por ejemplo, incluso si existe un flujo en la lista 408d de flujo de salida y la máquina 408 virtual envía un paquete de red a la NIC 410 física, el puente 412 virtual verifica aún el paquete contra la memoria caché 412a de flujo de salida.

En algunas realizaciones, puede ser deseable almacenar solo una porción de los flujos de salida en la lista 408d de flujo de salida. Por ejemplo, cierta información de los flujos de salida puede ser confidencial (por ejemplo, una dirección IP que se utilizará para NAT) y, como se ha señalado, la máquina 408 virtual puede ser una entidad no fiable. En ese sentido, la lista 408d de flujo de salida puede contener una lista de flujo (es decir, la información condicional utilizada para hacer coincidir un paquete con un flujo) sin información de acción. Por lo tanto, la lista 408d de flujo de salida solo podrá contener suficiente información para permitir que el controlador 408b de NIC virtual adopte la decisión de si se debe enviar el paquete a la NIC 410 física o al host 402.

La Figura 5 ilustra una de arquitectura 500 informática ilustrativa que incluye capas de un conmutador virtual multi-capa ilustrativo. Por ejemplo, el conmutador 104 virtual puede incluir capas de reglas y flujos para cada máquina virtual. Cada capa se muestra incluyendo un conjunto independiente de conjuntos de reglas y tablas de flujo. Tal como se representa, por ejemplo, las capas pueden incluir la capa 502 y la capa 504. Los paquetes de red atraviesan las capas en una de dos direcciones basándose en si el paquete de red está siendo enviado o recibido. Cuando un paquete de red está siendo recibido en nombre de una máquina virtual, por ejemplo, el paquete puede atravesar las capas de la capa de inferior hacia arriba (es decir, de la capa 504 a la capa 502, como se representa por las flechas 506). En contraste, cuando un paquete de red está siendo enviado desde una máquina virtual, el paquete puede atravesar las capas de la capa superior hacia abajo (es decir, de la capa 502 a la capa 504, como se representa por las flechas 508).

En algunas realizaciones, cada capa coincide con un paquete de red con su propio conjunto de flujos/reglas y

adopta cualquier acción apropiada antes de reenviar el paquete a la capa siguiente. Por ejemplo, un paquete puede desencapsularse en la capa 504 y someterse después a una operación de NAT en la capa 502. En algunas realizaciones, el paquete deja de atravesar las capas y se descarta si se adopta una acción de “bloquear”. Si bien el conmutador 104 virtual puede incluir capas de tablas de flujo, estos flujos se almacenan normalmente en forma plana cuando se descargan en la NIC 110 física.

Por consiguiente, la presente invención proporciona un modelo de regla y de flujo genérico que posibilita que se descarguen flujos en una NIC física. La descarga de los flujos posibilita que se realice algo de procesamiento de paquetes en la NIC física y elimina la necesidad de enviar algunos paquetes al conmutador virtual de un host para su procesamiento. En ese sentido, la presente invención puede reducir la latencia y el uso de CPU asociados con el procesamiento de paquetes de red para máquinas virtuales.

Las realizaciones descritas han de considerarse en todos los aspectos solamente como ilustrativas y no restrictivas. El ámbito de la invención se indica, por tanto, por las reivindicaciones adjuntas más que por la descripción anterior.

REIVINDICACIONES

1. Un procedimiento de procesamiento de paquetes de red para una máquina (108) virtual que se ejecuta en un sistema (100) informático, incluyendo el sistema informático uno o más procesadores y memoria de sistema, incluyendo también el sistema informático una tarjeta (110) de interfaz de red, NIC, física, y ejecutando una partición (102) host, comprendiendo el procedimiento:

un acto de un conmutador (104) virtual en la partición host de mantener (202) una pluralidad de conjuntos (106a; 106b) de reglas para una máquina virtual, incluyendo el conmutador virtual una pluralidad de capas (502, 504) de conjuntos de reglas, incluyendo cada capa un conjunto independiente de conjuntos de reglas;
un acto de la NIC física de mantener (204) una o más tablas (112a; 112b) de flujo para la máquina virtual;
un acto de la NIC física de recibir (206) un paquete de red asociado con la máquina virtual; y
un acto de procesar (208) el paquete de red para la máquina virtual, incluyendo:

un acto de la NIC física de comparar (210) el paquete de red con la una o más tablas de flujo,
cuando el paquete de red coincide con un flujo en la una o más tablas de flujo, un acto de la NIC física de realizar (212) una acción sobre el paquete de red basándose en el flujo coincidente,
cuando el paquete de red no coincide con un flujo en la una o más tablas de flujo, un acto de la NIC física de hacer pasar (214) el paquete de red a la partición host para su procesamiento contra la pluralidad de conjuntos de reglas, y
cuando el paquete de red no coincide con un flujo en la una o más tablas de flujo, un acto del conmutador virtual en la partición host de comparar el paquete de red con una regla de la pluralidad de conjuntos de reglas, incluyendo que el paquete de red atraviese las capas y que cada capa haga coincidir el paquete de red con su propio conjunto de reglas.

2. El procedimiento de acuerdo con la reivindicación 1, en el que, cuando el paquete de red coincide con una regla en el uno o más conjuntos de reglas, un acto de la partición host de realizar una acción sobre el paquete de red basándose en la regla coincidente.

3. El procedimiento de acuerdo con la reivindicación 2, en el que, cuando el paquete de red coincide con una regla en el uno o más conjuntos de reglas, un acto de la partición host de crear uno o más flujos en la NIC física en la una o más tablas de flujo.

4. El procedimiento de acuerdo con la reivindicación 1, que comprende además un acto de la partición host de mantener una o más tablas de flujo para la máquina virtual, y en el que la una o más tablas de flujo mantenidas en la NIC física comprenden un subconjunto de la una o más tablas de flujo mantenidas en la partición host.

5. Un medio legible por ordenador que tiene almacenadas instrucciones ejecutables por ordenador que, cuando se ejecutan por uno o más procesadores de un sistema (100) informático, hacen que el sistema informático realice un procedimiento de procesamiento de paquetes de red para una máquina (108) virtual que se ejecuta en el sistema informático, comprendiendo el procedimiento:

un acto de un conmutador (104) virtual en una partición (102) host del sistema de informático de mantener (302) una pluralidad de conjuntos (106a; 106b) de reglas para una máquina virtual, incluyendo el conmutador virtual una pluralidad de capas (502, 504) de conjuntos de reglas, incluyendo cada capa un conjunto independiente de conjuntos de reglas;
un acto del conmutador virtual de mantener (304) una o más tablas (106a; 106d) de flujo para la máquina virtual;
un acto del conmutador virtual de descargar (306) al menos una porción de la una o más tablas de flujo en una tarjeta (110) de interfaz de red, NIC, física; y
un acto del conmutador virtual de procesar (308) un paquete de red para la máquina virtual, incluyendo:

recibir (310), por el conmutador virtual, el paquete de red a partir de una de la máquina virtual o la NIC física;
hacer coincidir (312), por el conmutador virtual, el paquete de red con una regla en la pluralidad de conjuntos de reglas, incluyendo que el paquete de red atraviese las capas y que cada capa haga coincidir el paquete de red con su propio conjunto de reglas; y
basándose en la coincidencia de los paquetes de red con la regla:

crear (314), por el conmutador virtual, un flujo en la una o más tablas de flujo; y
descargar (316), por el conmutador virtual, el flujo en la NIC física.

6. El producto de programa informático de acuerdo con la reivindicación 5, en el que el acto de mantener la pluralidad de conjuntos de reglas para una máquina virtual comprende un acto de mantener un conjunto de reglas de entrada y un conjunto de reglas de salida.

7. El producto de programa informático de acuerdo con la reivindicación 5, en el que el acto del conmutador virtual de procesar un paquete de red para la máquina virtual incluye también:

realizar, por el conmutador virtual, al menos una acción en el paquete de red basándose en la regla.

8. El producto de programa informático de acuerdo con la reivindicación 7, en el que la al menos una acción comprende una o más de una inspección de paquete o una operación de manipulación de paquete.

9. Un sistema (100) informático, que comprende:

uno o más procesadores;
memoria de sistema;
una tarjeta (110) de interfaz de red, NIC, física; y
uno o más medios de almacenamiento informático que tienen almacenadas en los mismos instrucciones ejecutables por ordenador que, cuando se ejecutan por el uno o más procesadores, ejecutan un conmutador (104) virtual, configurándose el conmutador virtual para:

ejecutarse dentro de una partición (102) host en el sistema informático;
mantener (302) un conjunto (106b) de reglas de entrada y un conjunto (106a) de reglas de salida para una máquina (108) virtual, incluyendo el conmutador virtual una pluralidad de capas (502, 504) de conjuntos de reglas, incluyendo cada capa un conjunto de reglas de entrada correspondiente y un conjunto de reglas de salida correspondiente;
mantener (304) una tabla (106d) de flujo de entrada y una tabla (106c) de flujo de salida para la máquina virtual;
descargar (306) al menos una porción de una o más de la tabla de flujo de entrada o la tabla de flujo de salida en un puente (112) virtual de la NIC física; y
procesar (308) un paquete de red para la máquina virtual, incluyendo:

recibir (310) el paquete de red de una o más de la máquina virtual o la NIC física;
hacer coincidir (312) el paquete de red con una regla en uno del conjunto de reglas de entrada o el conjunto de reglas de salida, incluyendo que el paquete de red atraviese las capas y que cada capa haga coincidir el paquete de red con su propio conjunto de reglas; y
basándose en la coincidencia del paquete de red con la regla:

crear (314) un flujo en una o más de la tabla de flujo de entrada o la tabla de flujo de salida en el conmutador virtual; y
descargar (316) el flujo en una o más de una memoria (112b) caché de flujo de entrada o una memoria (112a) caché de flujo de salida en el puente virtual de la NIC física.

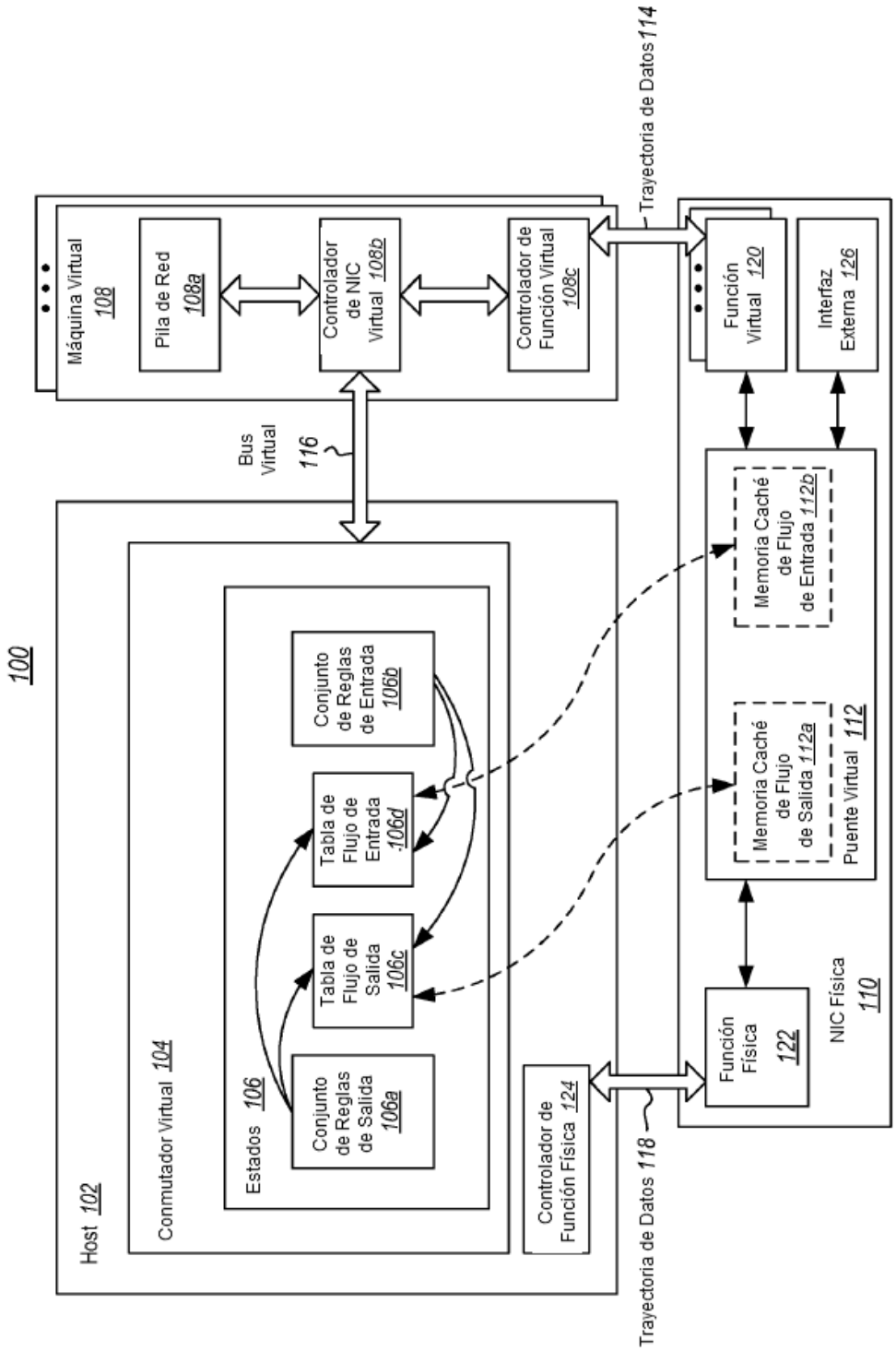
100
Host 102
Máquina Virtual 108
Pila de Red 108a
Controlador de NIC Virtual 108b
Controlador de Función Virtual 108c
Bus Virtual 116
Trayectoria de Datos 114
Función Virtual 120
Interfaz Externa 126
Conmutador Virtual 104
Estados 106
Conjunto de Reglas de Entrada 106b
Tabla de Flujo de Entrada 106d
Tabla de Flujo de Salida 106c
Conjunto de Reglas de Salida 106a
Memoria Caché de Flujo de Entrada 112b
Memoria Caché de Flujo de Salida 112a
Puente Virtual 112
Controlador de Función Física 124
Trayectoria de Datos 118
Función Física 122
NIC Física 110

Figura 1

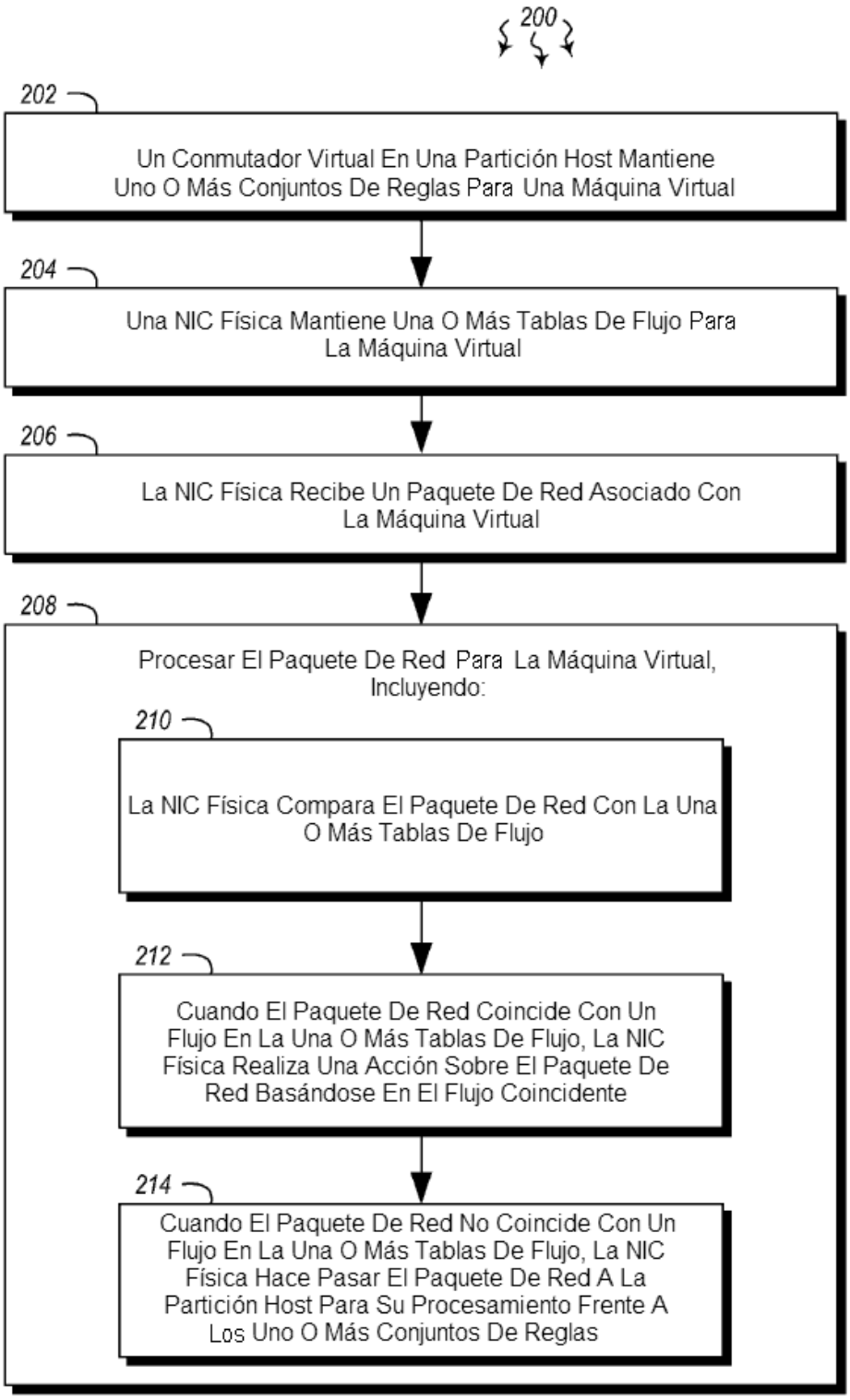
200
202
Un Conmutador Virtual En Una Partición Host Mantiene Uno O Más Conjuntos De Reglas Para Una Máquina Virtual
204
Una NIC Física Mantiene Una O Más Tablas De Flujo Para La Máquina Virtual
206
La NIC Física Recibe Un Paquete De Red Asociado Con La Máquina Virtual
208
Procesar El Paquete De Red Para La Máquina Virtual, Incluyendo:
210
La NIC Física Compara El Paquete De Red Con La Una O Más Tablas De Flujo
212
Cuando El Paquete De Red Coincide Con Un Flujo En La Una O Más Tablas De Flujo, La NIC Física Realiza Una Acción Sobre El Paquete De Red Basándose En El Flujo Coincidente
214
Cuando El Paquete De Red No Coincide Con Un Flujo En La Una O Más Tablas De Flujo, La NIC Física Hace Pasar El Paquete De Red A La Partición Host Para Su Procesamiento Frente A Los Uno O Más Conjuntos De Reglas

Figura 2

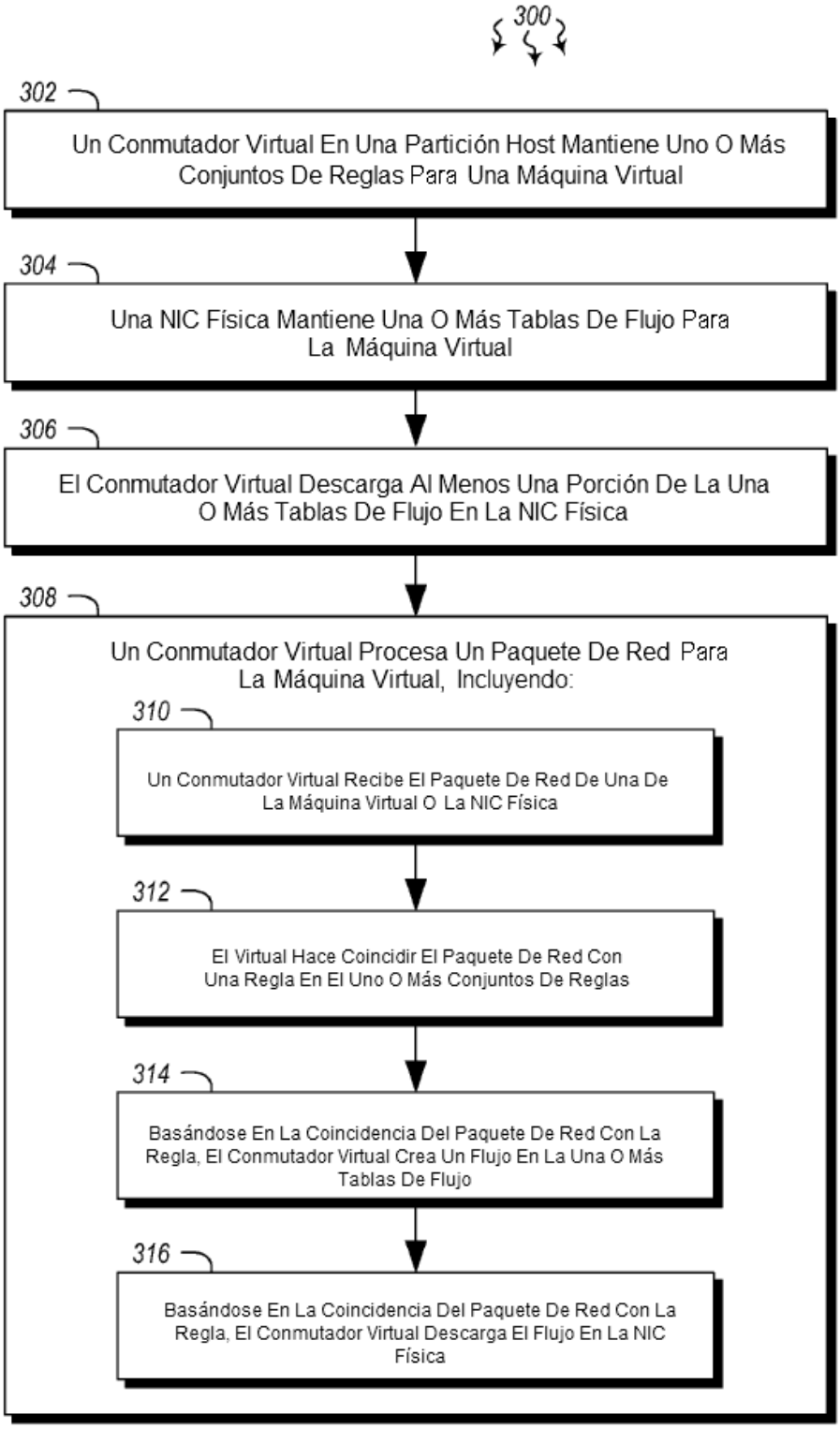
300
302
Un Conmutador Virtual En Una Partición Host Mantiene Uno O Más Conjuntos De Reglas Para Una Máquina Virtual
304
Una NIC Física Mantiene Una O Más Tablas De Flujo Para La Máquina Virtual
306
El Conmutador Virtual Descarga Al Menos Una Porción De La Una O Más Tablas De Flujo En La NIC Física
308
Un Conmutador Virtual Procesa Un Paquete De Red Para La Máquina Virtual, Incluyendo:
310
Un Conmutador Virtual Recibe El Paquete De Red De Una De La Máquina Virtual O La NIC Física
312
El Virtual Hace Coincidir El Paquete De Red Con Una Regla En El Uno O Más Conjuntos De Reglas
314
Basándose En La Coincidencia Del Paquete De Red Con La Regla, El Conmutador Virtual Crea Un Flujo En La Una O Más Tablas De Flujo
316
Basándose En La Coincidencia Del Paquete De Red Con La Regla, El Conmutador Virtual Descarga El Flujo En La NIC Física

Figura 3

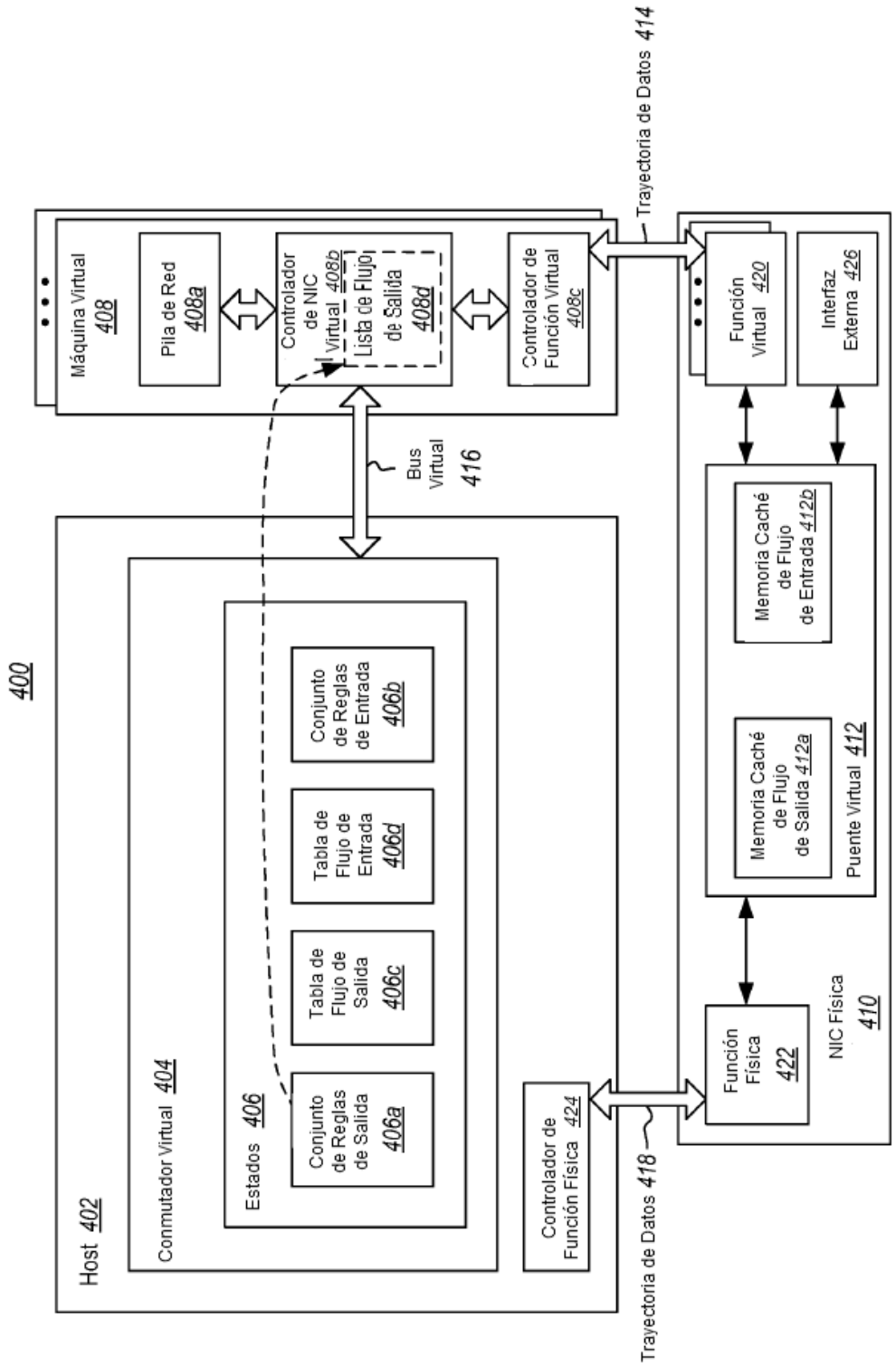
400
Host 402
Conmutador Virtual 404
Estados 406
Conjunto de Reglas de Salida 406a
Tabla de Flujo de Salida 406c
Tabla de Flujo de Entrada 406d
Conjunto de Reglas de Entrada 406b
Controlador de Función Física 424
Trayectoria de Datos 418
Máquina Virtual 408
Pila de Red 408a
Controlador de NIC Virtual 408b
Lista de Flujo de Salida 408d
Controlador de Función Virtual 408c
Bus Virtual 416
Trayectoria de Datos 414
NIC Física 410
Función Física 422
Puente Virtual 412
Memoria Caché de Flujo de Salida 412a
Memoria Caché de Flujo de Entrada 412b
Función Virtual 420
Interfaz Externa 426

Figura 4

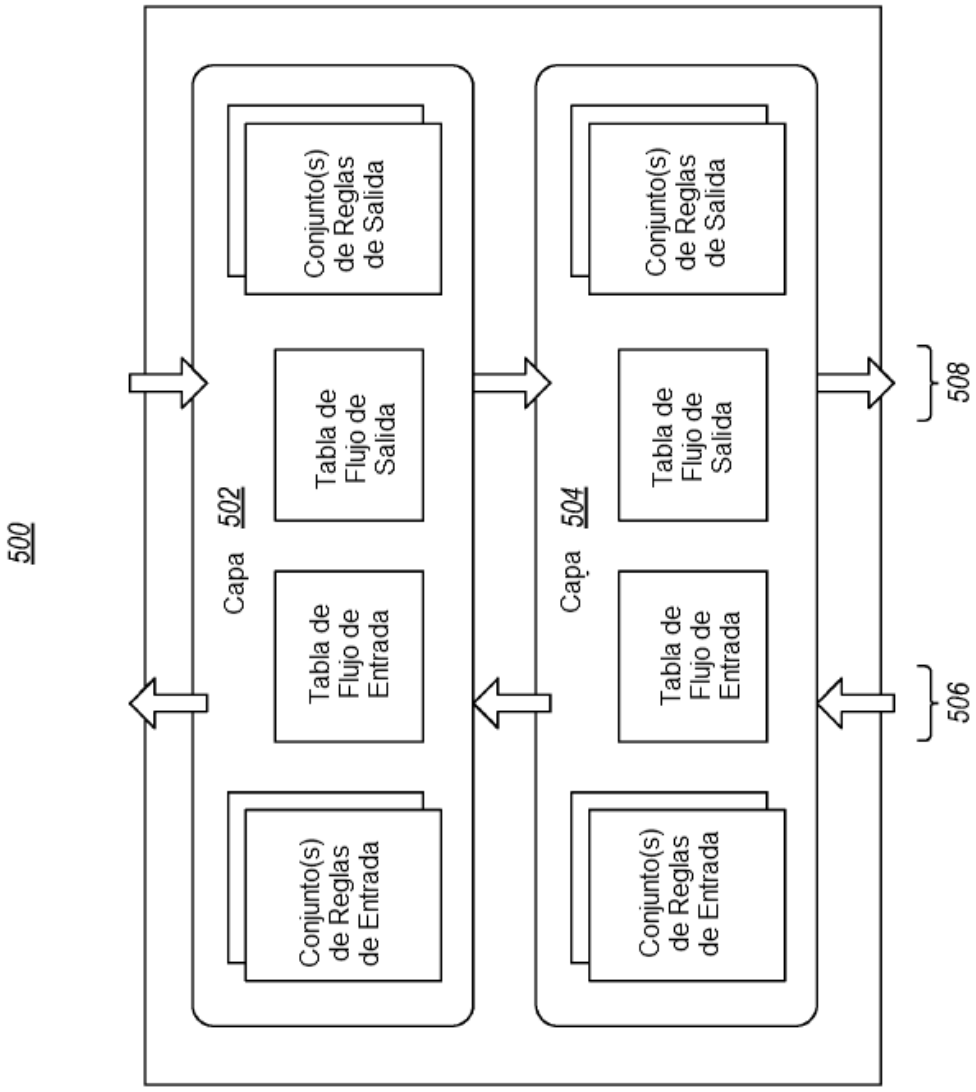
500
Capa 502
Conjunto(s) de Reglas de Entrada
Tabla de Flujo de Entrada
Tabla de Flujo de Salida
Conjunto(s) de Reglas de Salida
Capa 504
Conjunto(s) de Reglas de Entrada
Tabla de Flujo de Entrada
Tabla de Flujo de Salida
Conjunto(s) de Reglas de Salida
506
508

Figura 5