(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 620 383**

(51) Int. CI.:

H04L 12/741 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **17.04.2014 PCT/EP2014/057963**

(87) Fecha y número de publicación internacional: **23.10.2014 WO2014170458**

(96) Fecha de presentación y número de la solicitud europea: **17.04.2014 E 14719283 (5)**

(97) Fecha y número de publicación de la concesión europea: **01.03.2017 EP 2984797**

(54) Título: **Consulta de una tabla de reenvío de tráfico**

(30) Prioridad:

19.04.2013 GB 201307130
11.04.2014 GB 201406568

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
28.06.2017

(73) Titular/es:

ENTUITY LIMITED (100.0%)
9a Devonshire Square
London EC2M 4YL, GB

(72) Inventor/es:

ROPER, DR. JEFFREY JOHN

(74) Agente/Representante:

UNGRÍA LÓPEZ, Javier

DESCRIPCIÓN

Consulta de una tabla de reenvío de tráfico

La presente invención se refiere a la consulta de una tabla de reenvío de tráfico.

Las redes de ordenadores forman la base de infraestructura de TI (Tecnología de la Información) en una amplia variedad de contextos. Tales redes de ordenadores comprenden dispositivos interconectados de varios tipos. El objetivo de la red es soportar el flujo de mensajes entre esos dispositivos para entregar información, aplicaciones y servicios, etc., sobre la red. Un número de técnicas están disponibles para la gestión de una red. En este contexto, la gestión de una red incluye la monitorización de la red para identificar puntos de fallo y otras áreas problemáticas, tales como puntos de acceso, y proporcionar información a los administradores y a los usuarios de la red para permitir arreglar los problemas. Hay una serie de herramientas disponibles para proporcionar una topología de red. La topología de una red identifica cómo los dispositivos de la red están física o lógicamente conectados entre sí. Por lo tanto, cualquier dispositivo único particular puede tener una o más conexiones a un dispositivo vecino. Herramientas informatizadas, que "descubren" una red están disponibles, y crean topologías de red que definen la interconexión de los dispositivos en la red, y la naturaleza de esos dispositivos. Esto puede verse en el documento EP 1 916 812 A1.

Sumario

Los inventores han desarrollado un enfoque para la consulta de dispositivos de red para determinar lo que harían con un paquete hipotético (en oposición a la consulta para obtener información específica del protocolo de enrutamiento).

Según un aspecto de la presente invención, se proporciona un método implementado por ordenador de consulta de una tabla de reenvío de tráfico a un dispositivo en una red de ordenadores, teniendo la tabla de reenvío de tráfico entradas que son accesibles mediante una clave, comprendiendo el método: identificar una dirección de reenvío para su uso en la consulta de la tabla de reenvío de tráfico, en el que la dirección de reenvío constituye solo una parte de una clave y en el que una parte restante de la clave es un índice incrustado en la tabla de reenvío de tráfico; combinar la dirección de reenvío de una pluralidad de índices integrados de la tabla de reenvío de tráfico para generar un conjunto de claves para consultar la tabla de reenvío de tráfico; y generar un mensaje de consulta para la tabla de reenvío utilizando al menos una de dicho conjunto de claves.

La técnica definida anteriormente se denomina aquí como "modulación especulativa". Permite generar una lista limitada de claves especulativas para simplificar la consulta de la tabla de reenvío de tráfico, y para reducir el tráfico necesario para este tipo de consultas a través de una red de ordenadores. Mediante la generación de una pequeña lista limitada de claves especulativas, todas las solicitudes se pueden emitir en paralelo (es decir, un número de claves de consulta puede estar contenido en un mensaje de consulta común), en lugar de secuencialmente.

La dirección de reenvío se puede combinar con cada índice incrustado combinando lógicamente una secuencia de bits que representa la dirección de reenvío con una secuencia de bits que representa el índice incrustado.

Cuando la tabla de reenvío es una tabla de enrutamiento, cada índice de la tabla de enrutamiento es una máscara de red. Las claves que se utilizan en el mensaje de consulta se pueden seleccionar de solo claves únicas que se generan mediante la combinación de manera lógica de una secuencia de bits que representa la dirección de reenvío con una secuencia de bits que representa la máscara de red. La dirección de reenvío puede ser una dirección IP (Protocolo de Internet) y la tabla de reenvío puede ser para un dispositivo de enrutamiento de capa 3.

La técnica de modulación especulativa es también aplicable cuando el índice incrustado es un índice de interfaz de una tabla ARP en un dispositivo de reenvío de tráfico, tal como un enrutador.

La invención también proporciona un ordenador configurado para implementar un método de consulta de una tabla de reenvío de tráfico, en el que el ordenador comprende un procesador y una memoria que contiene un programa de ordenador que, cuando es ejecutado por el procesador, realiza las etapas de los métodos definidos anteriormente.

La invención también proporciona un producto de programa de ordenador en forma de un programa de ordenador que, cuando se ejecuta mediante un ordenador, implementa el método definido anteriormente.

La técnica de modulación especulativa es particularmente útil cuando se usa en un enfoque novedoso desarrollado por los inventores para identificar la ruta tomada a una red de dispositivos interconectados para un flujo de mensajes particular. Esa técnica se basa en el uso de una cantidad mínima de datos recogidos "por adelantado" - específicamente solo la topología de la red estática y la ubicación del servidor final (cuyos clientes y los servidores están conectados a conmutadores de acceso/borde), y recoge cualquier otra cosa que se requiera al vuelo y de manera muy selectiva, como se requiere para este tipo de datos muy dinámicos. Para entornos dinámicos modernos, la capacidad de calcular una ruta de extremo a extremo ahora, es decir, en tiempo real, tiene una amplia

aplicabilidad. La recogida de datos y su procesamiento tiene que ser muy rápido para que el algoritmo sea de valor practicable, cuando se utiliza con redes de gran escala, en el mundo real.

El comportamiento en un dispositivo particular se denomina "comportamiento por salto" (PHB). Aquí es donde la técnica de modulación especulativa es particularmente valiosa. PHB por sí solo no puede proporcionar una ruta de extremo a extremo. Sin embargo, tener un paquete que deja el dispositivo en una interfaz específica puede ser útil si no se sabe qué dispositivo e interfaz están conectados a esa interfaz. Mediante el uso de topología de red junto con PHB, se puede lograr el cálculo directo de una trayectoria de extremo a extremo a través de la red mediante un flujo de aplicación.

La consulta que se transmite a cada dispositivo está adaptada para consultar cada dispositivo para determinar la identificación de un puerto de salida que representa los puertos de salida que el dispositivo debe usar para un hipotético mensaje dirigido a un destino identificado por el identificador de destino. Debe tenerse en cuenta que el identificador de destino para cualquier consulta dada puede o no ser un identificador de destino del dispositivo terminal, dependiendo de la ubicación en la red del dispositivo que se consulta. Esto se puede conseguir cuando el dispositivo es un enrutador mediante la consulta de lo que está en su tabla de enrutamiento activo en el momento en que se recibe la consulta. El identificador de destino es la dirección de reenvío que se utiliza para la tabla de enrutamiento o una tabla ARP, por ejemplo, una dirección IP (Protocolo de Internet).

La propia consulta se puede acomodar en un mensaje o señal transmitida desde el ordenador monitor al dispositivo que se va a consultar (dispositivo de enfoque). El mensaje de consulta o la señal no constituye el flujo de mensajes para los que la ruta se debe determinar. En cambio, cada consulta contiene un identificador de destino (dirección de reenvío), que consulta la tabla de reenvío de un dispositivo de enfoque para descubrir cómo el dispositivo de enfoque se ocuparía de un mensaje hipotético dirigido a ese destino si tuviera que tomar la decisión en el momento de recibir la consulta. De este modo, el dispositivo de enfoque devuelve un resultado que identifica a un puerto de salida inmediato que habría sido utilizado en ese momento para un mensaje real dirigido a ese destino. Las consultas se pueden transmitir mientras la red está activa y mientras el flujo de mensajes está en posición. Sin embargo, también se pueden transmitir cuando el propio flujo de mensajes no está activo - la técnica se puede utilizar en cualquier contexto.

Cuando la consulta es en la forma de un mensaje o paquete, por ejemplo, el mensaje puede ser un mensaje SNMP con una dirección IP de destino, llevará su propia dirección de destino y se entregará a través de la red desde el ordenador monitor al dispositivo de enfoque. En ese caso, la dirección de destino del mensaje de consulta es la del dispositivo de enfoque. Esto no es lo mismo que el identificador de destino (dirección de reenvío) que se incluye en la propia consulta. En una disposición alternativa, una señal o señales de consulta pueden enviarse desde el ordenador monitor a través de conexiones directas a los dispositivos de enfoque, tal como, por ejemplo, a través de un mecanismo CLI o XML API.

La determinación de la topología de red se puede hacer de muchas maneras. Las técnicas que pueden utilizarse por separado o en combinación para dar una buena representación de la conectividad de la red incluyen, por ejemplo:

- *Cisco Discovery Protocol (CDP)*
- *Link Layer Discovery Protocol (LLDP)*
- *SynOptics Network Management Protocol (SoNMP)*
- *Spanning Tree Protocol (STP)*
- *Traceroute IP*
- *IPv6 Neighbour Discovery*
- *adiciones de usuario* / *modificaciones* / *eliminaciones*

El conocimiento de la topología de una red es muy útil, pero no proporciona una solución a todos los problemas que pueden producirse. Las redes se utilizan cada vez más para proporcionar la infraestructura para soportar la entrega de aplicaciones y servicios entre localizaciones geográficas remotas, ya sea a través de largas distancias o en redes extremadamente complejas con un gran número de dispositivos interconectados. Cada vez más, los administradores y los usuarios de la red están interesados en saber no necesariamente todos los detalles de la red, sino entender el rendimiento de la entrega de aplicaciones y servicios a través de una red. Por lo tanto, la llamada monitorización de "extremo a extremo" se está convirtiendo cada vez más popular. Con la monitorización de "extremo a extremo", las aplicaciones que implican el flujo de mensajes desde un dispositivo fuente a un dispositivo de destino tienen su rendimiento monitorizado a medida que se entregan entre dicha fuente y el dispositivo de destino. Los parámetros de rendimiento se pueden utilizar para estimar o adivinar posibles fallos en la red, a pesar de que no proporcionan ninguna información específica sobre la ubicación de dichos fallos y, por lo tanto, no apuntan directamente a una solución.

A menudo un dispositivo fuente es un servidor que proporciona un servicio particular, y el dispositivo de destino es un terminal de cliente que está conectado al servidor a través de la red y que requiere el uso de dicho servicio. El término "dispositivo" que se utiliza aquí se destina a cubrir todos los dispositivos que se pueden conectar en una red.

El término "servidor" se utiliza para designar un dispositivo que se encarga de la entrega de un servicio o aplicación, y el término "cliente" se utiliza para designar un dispositivo (ya sea basado en usuario u otra máquina o servidor dependiente) que depende de esa aplicación o servicio.

Una dificultad significativa en adivinar dónde podría haber un problema cuando se observa que el rendimiento de una aplicación se está deteriorando es una falta de comprensión acerca de la ruta a través de la red que el flujo de mensajes para esa aplicación podría haber tomado. Las redes dependen de muchos tipos de dispositivos de red (por ejemplo, enrutadores, conmutadores, cortafuegos, equilibradores de carga, etc.) para conectar sus dispositivos de punto final, de manera que es muy difícil decir para cualquier punto final de origen determinado cómo se dirige el mensaje desde ese punto final a través de la red a un punto extremo de destino dado. La complejidad de esta determinación de ruta se ve agravada por el uso de varias rutas alternas, rutas redundantes, equilibrio de carga, etc.

Se han hecho intentos de predecir cómo se dirige un paquete particular a través de una red. Tales predicciones están basadas en un modelo complejo de la topología de la red, junto con indicaciones sobre una base por dispositivo en cuanto a cómo un dispositivo particular se comportará en la red. Los dispositivos de red pueden ser muy sofisticados, y no se han desarrollado un gran número de algoritmos complejos para determinar una estrategia de enrutamiento en cualquier dispositivo particular. Por otra parte, esa estrategia de enrutamiento puede depender del tráfico y de otras consideraciones ambientales que afectan a la red (como la insuficiencia de otros dispositivos, etc.). Los algoritmos complejos que pueden aplicarse mediante un dispositivo para determinar una estrategia de enrutamiento pueden incluir, por ejemplo:

- *Interfaz de entrada y tecnología de interfaz de entrada*
- *Cabeceras de paquetes (L2, L3, MPLS, ATM, etc.)*
- *Rutas estáticas y directamente conectadas*
- *Tablas de enrutamiento compartidas (pleno conocimiento de BGP, OSPF, RIP, EIGRP, etc. - vecinos activos, estados de enlace, costes de ruta, ponderaciones de ruta, etc.)*
- *Tablas de reenvío MAC aprendidas*
- *Listas de control de acceso*
- *Tecnologías de superposición de red (por ejemplo, MPLS, VLANs 802.1q), etc.*
- *Tecnologías de prevención de bucle - por ejemplo, PVSTP*
- *Protocolos de tunelización (MPLS, IPSec, SSL, GRE)*
- *Cargar enlaces redundantes / equilibrados*
- *Puertas de enlace predeterminadas*

Sin embargo, incluso si en el pasado, en principio, era posible predecir que un determinado paquete será enviado al siguiente en un dispositivo particular, esto requería una gran cantidad de datos, que es lento a recoger, y podría estar fuera de fecha dentro de segundos debido a la naturaleza en tiempo real de la operación de los dispositivos de enrutamiento. Por otra parte, la mera adquisición de estos datos coloca una carga significativa en ambos dispositivos de red y en las redes.

El método de identificación de ruta descrito en este documento usando la técnica de modulación especulativa permite un número de técnicas de análisis de red útiles. Permite la determinación de ruta bajo demanda, de modo que un administrador que trata de determinar la ruta de acceso para una aplicación particular puede pedir más o menos instantáneamente al ordenador monitor y recibir un resultado de la ruta de acceso.

Se permite múltiples descubrimientos de ruta. Es decir, debido a cambios en el entorno de la red, los dispositivos de enrutamiento pueden dirigir un flujo de mensajes de manera diferente en función de esos cambios. Así, un primer conjunto de consultas para identificar una ruta de acceso podría grabar una primera ruta, mientras que un segundo conjunto de consultas podría identificar una segunda ruta, incluso cuando el primero y segundo conjunto de consultas son muy cercanas entre sí en el tiempo. La información sobre varias rutas entre puntos finales comunes (es decir, el mismo dispositivo de origen y el mismo dispositivo de destino) puede presentarse gráficamente o visualmente para mostrar a un usuario no solo la naturaleza de la ruta de acceso, sino el porcentaje de tiempo que cada ruta se adopta para un flujo de mensajes particular. Esto se puede lograr fácilmente porque las propias consultas no representan una sobrecarga significativa a la red y, por lo tanto, múltiples conjuntos de consultas pueden enviarse sin afectar significativamente al rendimiento.

El método permite la detección de cambios rápidos de la ruta legítima. Esto es, un ajuste de la red puede hacer que la ruta cambie y esto puede ser detectado y señalizado a un usuario en una interfaz de usuario gráfica visual.

Cuando hay múltiples rutas entre dispositivos de origen y de destino comunes, las rutas pueden llevar a diferentes latencias. A veces, un dispositivo de enrutamiento que realiza un enrutamiento inteligente puede causar un fenómeno conocido como "variación de ruta", donde un flujo de mensaje particular cambia continuamente de ruta a ruta. Puede ser útil para un administrador de red identificar este tipo de sucesos, debido a las implicaciones de tales cambios de trayectoria en una latencia de extremo a extremo y las implicaciones de esta "rápida variación" en las

conversaciones telefónicas de voz sobre IP, por ejemplo.

El método puede ser utilizado para localizar fallos de ruta. Esto es, de la realización preferida del método, las consultas son enviadas y los resultados se reciben y se analizan para identificar el siguiente dispositivo hasta que un dispositivo se identifica como el dispositivo de destino. A veces, sin embargo, hay un fallo en la red, de manera que la red no emitiría el flujo de mensajes al dispositivo de destino. El método permite la identificación de esa situación, trabajando a lo largo de una ruta de extremo a extremo hasta que la ruta no puede ir más lejos y esta ubicación de red puede ser notificada a un administrador.

Además, el método puede permitir la posibilidad de reiniciarse en un dispositivo posterior en esa ruta, usando estimaciones basadas en la topología de red. El método de identificación de ruta a continuación se puede adoptar de nuevo hasta que el dispositivo de destino se alcanza desde el punto del fallo. De esta forma, las porciones de la red para las que el ordenador de monitorización no tiene visibilidad (por ejemplo, dispositivos que no tienen interfaz de gestión apropiada, o que pertenecen a una organización diferente) pueden circunnavegarse y el análisis de la ruta continúa.

El método también permite una identificación de enrutamiento asimétrico. No es poco común el flujo de mensajes entre un dispositivo de origen y un dispositivo de destino para adoptar diferentes rutas, dependiendo de su dirección. Esto es, una ruta hacia delante puede ser utilizada desde el dispositivo de origen al dispositivo de destino para el flujo de mensajes, y una ruta de retorno desde el dispositivo de destino al dispositivo de origen, que es diferente.

La ruta se registra en una memoria o almacén en el ordenador monitor o es accesible por el ordenador monitor. El registro de la ruta comprende un conjunto de dispositivos e interfaces conectadas. Esto se puede presentar en forma de un inventario ordenado de los dispositivos (componentes de la red) entre los dos puntos finales. Esto permite la monitorización de la disponibilidad de la ruta de la red, incluyendo la notificación de eventos, generación de informes, SLAs (acuerdos de nivel de servicio); gestión proactiva de la red, incluyendo la presentación de informes sobre los dispositivos que fallan, CPU de dispositivo alta, baja memoria del dispositivo, congestión de los puertos, etc., y el análisis de impacto (planificación de la capacidad, análisis "qué pasaría si").

Es una ventaja significativa que una asignación entre la aplicación o servicio prestado por la red y los propios dispositivos o componentes de la red se pueda determinar a través de la identificación de la ruta. Esto representa un importante paso adelante en la gestión de redes.

Para una mejor comprensión de la presente invención y para mostrar cómo la misma puede llevarse a efecto, se hará ahora referencia, a modo de ejemplo, a los dibujos adjuntos, en los que:

la figura 1 es un diagrama esquemático de una red;
las figuras 2a a 2c son una ilustración esquemática de un algoritmo de descubrimiento de ruta en proceso;
la figura 3 es un diagrama de flujo para un algoritmo de descubrimiento de ruta;
la figura 4 muestra una ruta descubierta;
la figura 5 es la estructura de una tabla de enrutamiento lineal;
la figura 6 ilustra un conjunto de resultados que surgen de la combinación de una dirección de destino con múltiples máscaras de ruta;
la figura 7 muestra una estructura de una tabla ARP;
la figura 8 es un diagrama esquemático de un ordenador monitor;
la figura 9 es un diagrama esquemático de un enrutador de capa 3;
la figura 10 es un diagrama esquemático de un conmutador de capa 2;
las figuras 11a a 11d son un diagrama de flujo de un servicio ejecutado en el ordenador monitor;
la figura 12 es un diagrama de flujo de un programa de ejecución en bucle;
la figura 13 es un diagrama de flujo que muestra un proceso de terminación del programa de ejecución en bucle;
la figura 14 es un diagrama de flujo que muestra la Opción C del programa de ejecución en bucle;
la figura 15 es un diagrama de flujo que muestra la Opción S del programa de ejecución en bucle;
la figura 16 es un diagrama de flujo que muestra la continuación del proceso de la Opción S;
la figura 17 es un diagrama de flujo que muestra el proceso en la Opción del programa de ejecución en bucle;
la figura 18 es un diagrama de flujo que muestra una continuación del proceso de la figura 17;
la figura 19 es un diagrama de flujo que muestra el proceso de la Opción c del programa de ejecución en bucle;
la figura 20 es un diagrama de flujo que muestra el proceso de la opción A;
la figura 21 ilustra un proceso para obtener un indicio de VLAN;
la figura 22 es un proceso para la obtención de un puerto conectado y un dispositivo conectado para almacenarse en un registro de ruta;
la figura 23 es un diagrama de flujo que muestra un proceso iterativo para encontrar una ruta que se utiliza en algunas de las opciones anteriores;
las figuras 24, 25 y 26 ilustran tres procesos de preparación;
la figura 27 es un diagrama de flujo que ilustra un proceso de descubrimiento de ruta que se utiliza en el proceso de descubrimiento de rutas iterativo de la figura P; y
la figura 28 ilustra un proceso para la búsqueda de una entrada de base de datos de reenvío de un dispositivo de

enfoque.

La figura 1 es un diagrama esquemático de una red. La red se extiende sobre un número de diferentes ubicaciones geográficas. En cada ubicación geográfica final hay dispositivos de punto final y dispositivos o nodos de red. Los dispositivos de red incluyen enrutadores y conmutadores. El núcleo de la red comprende una pluralidad de dispositivos de red. Teniendo en cuenta la ubicación geográfica marcada los terminales de cliente 2 pueden actuar como dispositivos de punto final. Del mismo modo, un servidor 4 puede actuar como un dispositivo de punto final y la impresora 6 se puede considerar un dispositivo de punto final. Dispositivos similares se muestran en las ubicaciones geográficas de París y Nueva York con diferentes diseños (Nueva York que muestran un conjunto de servidores o centro de datos). Se debe tener en cuenta que en la ubicación de Nueva York una pluralidad de servidores 8 representan la aplicación clave o dispositivos de punto final de servicio.

Se debe apreciar que la red mostrada en la figura 1 se da a modo de un ejemplo. Hay una amplia variedad de posibles redes y la presente invención se puede utilizar en cualquier red de dispositivos interconectados. En particular, la naturaleza de los dispositivos de punto final y dispositivos o nodos de red específicos puede variar. En la red particular que se divulga, los dispositivos de red pueden ser dispositivos de capa 3 o de capa 2.

El modelo OSI (Interconexión de sistemas abiertos) define siete capas dentro de los que los protocolos de los sistemas de comunicación se pueden caracterizar. El algoritmo de búsqueda de ruta descrito aquí calcula las rutas de red utilizando información disponible en las capas 2 y 3.

Los dispositivos que operan en la capa 2 (la capa de enlace de datos) tienen un conocimiento de los dispositivos inmediatamente adyacentes y tienen la responsabilidad de obtener paquetes de un dispositivo de capa 2 con el siguiente dispositivo de capa 2 (basado en la capa 2 de direcciones MAC (control de acceso de medios)).

Los dispositivos que operan en la capa 3 (la capa de red) son responsables de la propagación de los paquetes desde un punto en una red a otro punto de la red - a menudo a muchas decenas o cientos de dispositivos de separación. Para calcular los dispositivos que deben participar en una ruta de capa 3 dada (en este documento se refiere como los saltos de la capa 3), los dispositivos de la capa 3 intercambian información de enrutamiento y usan protocolos de enrutamiento para calcular la(s) ruta(s) más deseable(s). Para pasar los paquetes entre dispositivos de la capa 3 consecutivos en una ruta, se utilizan dispositivos que operan en la capa 2; a menudo con muchos dispositivos de la capa 2 (en adelante denominados como saltos de la capa 2) entre cada dispositivo de la capa 3.

De este modo, grandes redes se subdividen en múltiples segmentos con eficacia, típicamente cada uno conteniendo múltiples dispositivos de la capa 2, conectados por dispositivos de la capa 3.

La figura 9 es un diagrama muy esquemático de un dispositivo de enrutamiento de la capa 3. El dispositivo comprende un controlador 90, por ejemplo, en forma de un código de control de ejecución de microprocesador, firmware o cualquier otra implementación adecuada. El controlador 90 puede acceder a una tabla de enrutamiento 92 que se describe en más detalle más adelante con referencia a la figura 5. El dispositivo de enrutamiento de la capa 3 tiene puertos Pi/Po. Cada puerto está conectado a un enlace físico, como se ilustra en la red de la figura 1. En esta notación, Pi designa un puerto de "entrada" y Po designa un puerto de "salida". Esto es por conveniencia de notación, en la práctica, los dispositivos no suelen tener puertos que se dedican como puertos de entrada o de salida - si son de entrada o de salida depende de los datos que se transfieren en el momento. La mayoría de los puertos funcionan como salida y entrada todo el tiempo.

Los paquetes que llegan a un puerto de entrada Pi pueden tener sus identificadores de destino, por ejemplo, IP (direcciones de protocolo de internet), leídos por el controlador 90 a través de un bus 94. El controlador 90 accede a la tabla de enrutamiento 92 y, basándose en la información derivada de la misma, controla un conmutador de enrutamiento 96 al que se dirige el paquete entrante. El conmutador de enrutamiento 96 entonces dirige el paquete entrante a un puerto de salida Po apropiado, en función de la información de la tabla de enrutamiento. El dispositivo de enrutamiento incluye una tabla de asignación 91, que asigna direcciones de la capa 3 a la capa 2 para su enrutamiento posterior. La operación de estos dispositivos de enrutamiento se conoce en la técnica y, por lo tanto, no se describirá adicionalmente en este documento. Se observa en este contexto que la tabla de enrutamiento se puede consultar mediante paquetes desde el ordenador monitor que llegan por medio de los enlaces en el puerto de entrada Pi mediante la interceptación de dichos paquetes al controlador 90. Tales paquetes de consulta no se suministran al conmutador de enrutamiento 96 para más de enrutamiento, pero en cambio generan una respuesta que se emite desde el dispositivo de enrutamiento y se devuelve a la entidad de consulta a través de la red desde un puerto de salida. En este caso, dicha entidad de consulta es el ordenador monitor 16. Todos los paquetes transmitidos por la red (incluyendo los paquetes de consulta) contienen una fuente y una dirección de destino - el paquete de consulta tiene una dirección de origen correspondiente al ordenador monitor y una dirección de destino correspondiente al dispositivo que se está consultando. Cuando tiene que enviarse la respuesta, las direcciones de origen y de destino se intercambian para hacer que la dirección de origen el dispositivo que se está consultando y la dirección de destino el ordenador monitor.

La figura 10 es una versión muy esquematizado de un conmutador de capa 2. De manera similar a un dispositivo de

enrutamiento de capa 3, el conmutador de capa 2 tiene puertos Pi/Po, cada uno conectado a un enlace físico, tal como se muestra, por ejemplo, en la red de la figura 1. Como se mencionó anteriormente, los puertos no están generalmente dedicados como entrada o salida. Los paquetes entrantes en un puerto de entrada Pi se dirigen a un conmutador 100, que puede acceder a una base de datos de reenvío de capa 2 (FDB) 102 para determinar cómo dirigir los paquetes basados en identificadores de destino (normalmente cabeceras) en los paquetes. Una base de datos de reenvío de capa 2 asigna un identificador de un paquete entrante a un puerto de salida en el que se debe reenviar el paquete. Como ya se ha explicado anteriormente, de acuerdo con el modelo OSI, los identificadores para los dispositivos de enrutamiento de capa 3 son direcciones IP, mientras que los identificadores para los dispositivos de capa 2 son direcciones MAC.

Al igual que con los dispositivos de la capa 3, los de la capa 2 son conocidos en la técnica y, por lo tanto, no se describirán más en este documento. Sin embargo, se observa que, de nuevo, como en los dispositivos de la capa 3, pueden recibir una consulta en un paquete en un puerto de entrada Pi y generar una respuesta a esa consulta en la salida del conmutador de la capa 2 en un puerto de salida Po. Por lo tanto, los propios paquetes de consulta no se dirigen al conmutador, pero en lugar de generar una respuesta que se devuelve al dispositivo de consulta, en este caso al ordenador monitor 16.

Un controlador del conmutador 101 en el conmutador es responsable de reenviar el tráfico y generar respuestas.

Algunos dispositivos más recientes pueden realizar la función de la capa 3 y de la capa 2.

Las siguientes realizaciones descritas de la presente invención proporcionan un método de identificación de una ruta recorrida por un flujo de mensajes entre un dispositivo de origen dado y un dispositivo de destino dado. Por ejemplo, el punto final X podría considerarse como un dispositivo de origen y el punto final Y se podría considerar un dispositivo de destino. Ante una red de la figura 1, que es, como ya hemos comentado anteriormente, una tarea lejos de ser trivial para establecer qué ruta se adoptará a través de la red entre esos puntos finales en un momento dado y bajo un conjunto dado de condiciones ambientales. La figura 1 muestra un ordenador monitor 16 que ejecuta un programa de descubrimiento de ruta que permite descubrir esta ruta y registrarse. La figura 8 es una versión muy esquemática de un ordenador monitor 16. El ordenador 16 comprende un microprocesador 80 que puede acceder a la memoria 82 en la que se almacena el código para su ejecución por el procesador. En el presente caso, el código incluye el programa de descubrimiento de ruta. La memoria 82 también almacena un registro de ruta 81, ya que se crea mediante el programa de descubrimiento de ruta. El ordenador tiene una interfaz de usuario 84 que puede incluir un dispositivo de entrada de usuario, tal como un ratón o un teclado y una pantalla para mostrar información a un usuario. En particular, como se describe en más detalle en este documento, alerta después del programa de descubrimiento de ruta o información relativa al programa de descubrimiento de ruta se pueden mostrar a un usuario en la interfaz de usuario 84. Las figuras 2a a 2c ilustran etapas de la ruta como se describirá ahora.

En un nivel alto, el algoritmo utiliza la noción de un "dispositivo de enfoque", que es el dispositivo que actualmente se está consultando en cuanto a dónde se enviaría un paquete hipotético siguiente (es decir, qué interfaz enviaría el paquete hipotético). Comenzando en el dispositivo de origen, el algoritmo se desplaza hacia el dispositivo terminal (es decir, el destino final del paquete) mediante la evaluación de cada dispositivo de enfoque, a su vez - si el dispositivo está funcionando en la capa 3, se consulta para qué interfaz (puerto de salida) se usaría para enviar paquetes con destino al siguiente salto en la capa 3 (NHL3); si el dispositivo está operando en la capa 2, se consulta para qué interfaz (puerto de salida) se usaría para enviar paquetes con destino a la siguiente dirección de capa de salto 2 de la capa 3 (MAC) (NHL2). Usando la respuesta del dispositivo de enfoque en conjunción con una topología de red, se puede determinar el siguiente dispositivo en la ruta. De esta manera, el algoritmo trabaja a lo largo de la ruta de la capa 3, usando dispositivos de la capa 2 para desplazarse entre los nodos consecutivos de la capa 3.

Antes de comenzar el algoritmo de preparación, se localizan el dispositivo de origen y el dispositivo terminal. Esto puede no ser sencillo y técnicas para lograr esto se describen más adelante.

De acuerdo con el algoritmo de preparación, se localiza el primer salto. La ruta se siembra y el recuento de bucle se establece en cero. El límite de bucle regula el número de veces que se ejecuta un bucle de identificación de ruta (explicado más adelante).

Encuentro del primer salto en la capa 3

El primer salto se localiza al encontrar el siguiente salto inicial (el siguiente salto desde el dispositivo de origen) en la capa 3 (NHL3). En la siguiente explicación, se utiliza frecuentemente el término "consulta". Las consultas se generan y se estructuran como se describe en más detalle más adelante. El propósito de una consulta es localizar una dirección del siguiente salto y el puerto de salida desde un dispositivo de enfoque al que se dirige la consulta. La dirección NHL3 inicial puede determinarse mediante la consulta en primer lugar de un dispositivo de origen X utilizando la dirección IP de destino. Es decir, se hace un intento para consultar la tabla de enrutamiento en el dispositivo de origen para NHL3 y el puerto de salida. Si no se encuentra ninguna ruta, y el dispositivo de origen tiene un conmutador de la capa 3, este conmutador de acceso del Nivel 3 se consulta para NHL3 utilizando la dirección IP de destino. Si esto no tiene éxito, la puerta de enlace predeterminada en el dispositivo de origen se

consulta para determinar la NHL3. Si esto no tiene éxito, se realiza una consulta utilizando la dirección IP de destino al conmutador de acceso de la puerta de enlace predeterminada. Si no se encuentra la dirección NHL3, esto se considera como un fallo. Esto no significa que el algoritmo ha fallado, sino que un punto de fallo en la ruta de acceso puede haber sido identificado en este punto. Alternativamente, puede haber otras razones por las cuales no se ha encontrado ninguna NHL3.

Sembrar la ruta

Para sembrar la ruta, se añade el dispositivo de origen a la ruta cuando ha sido localizado. La interfaz de salida del dispositivo de origen se localiza y se añade a la ruta. Si NHL3 se encuentra en la tabla de enrutamiento en el dispositivo de origen, la interfaz de salida del dispositivo de origen para esta dirección NHL3 se añade a la ruta. Como se explica más adelante, puede comprobarse la dirección de la capa 2 (NHL2) correspondiente a la dirección de la capa 3 (NHL3). Si no se encuentra ningún puerto de salida para NHL3 en la tabla de enrutamiento en el dispositivo de origen, se utiliza la tabla de reenvío de la capa 2 en el dispositivo de origen para NHL2 para encontrar el puerto de salida. Si se encuentra, a continuación, el puerto de salida se añade a la ruta.

Información general sobre el algoritmo de descubrimiento de la ruta

La consulta enviada desde el ordenador monitor 16 al dispositivo de origen X se muestra como una flecha directa en la figura 2a, pero, de hecho, podría implementarse en la red de la figura 1 mediante el ordenador monitor 16 que emite un mensaje o paquete dirigido al dispositivo de origen X. Como se ha explicado, la consulta pregunta al dispositivo de origen para el siguiente salto de IP (y puerto de salida) para la IP terminal (IP de destino), que es la dirección de la capa 3 del punto de destino Y. El objetivo es provocar que el dispositivo de origen X suministre una respuesta que incluye NHL3 y el puerto de salida para NHL3 (la dirección IP del terminal). Véase la etapa S1 de la figura 3 y de la figura 2a.

Como se explicó anteriormente, puede haber situaciones en las que el dispositivo de origen no puede suministrar la información necesaria. Otras posibilidades mencionadas anteriormente para obtener el primer dispositivo de "enfoque" incluyen consulta del conmutador de acceso conectado a la información de enrutamiento de la capa 3 (en caso de que el conmutador de acceso sea un conmutador de la capa 3), si esto falla el algoritmo consulta el conmutador de acceso conectado a una puerta de enlace predeterminada y la dirección IP de la puerta de enlace predeterminada se utiliza como la primera NHL3.

En la etapa S2, la siguiente dirección de salto de la capa 2 (MAC) se resuelve a partir de la dirección NHL3 y NHL2 se establece en esta dirección MAC. Esto se puede lograr mediante la consulta de una tabla de asignación 91 que asigna L3 a direcciones L2. Una de esta tabla de asignación es una tabla ARP (otras incluyen "asignación directa" y descubrimiento vecino). Este puede ser el dispositivo de origen ARP, después del dispositivo de salto L3 ARP o ARP de caché global usando una consulta de ARP que se describe más adelante. El puerto de salida identificado en la etapa S1 se añade al registro de ruta S1A. En la etapa S3, el conmutador de red de origen (y el puerto) se encuentra usando una ubicación del servidor final en caché (a partir de consultas del conmutador CAM), y se definen como el dispositivo de enfoque. En la etapa S4, el conmutador de red terminal se encuentra utilizando la ubicación del servidor final en caché (a partir de consultas del conmutador CAM). El conmutador de origen se añadirá al registro de ruta.

El método está ahora listo para entrar en un bucle de identificación de ruta. En la etapa S5 se determina si NHL2 se conoce. Si lo es, el bucle se desplaza a la etapa S5A. Si no lo es, el proceso realiza la etapa S5B para resolver NHL2 mediante una consulta de ARP en el dispositivo de enfoque o el dispositivo NHL3. La generación de una consulta para correlacionar una dirección de capa 3 con una dirección de la capa 2 se describe con más detalle más adelante con referencia a la figura 7. En resumen, para el dispositivo que se consulta, se obtiene una lista de índices de interfaz (ifindices) de la topología de la red o caminando iflndex desde la tabla de interfaz del propio dispositivo. Cada iflndex para el dispositivo se combina con la dirección NHL 3 para generar un conjunto de claves para incluir en la consulta al dispositivo. Por lo tanto, una consulta que contenga estas claves se formula y se transmite al dispositivo de enfoque. El dispositivo de enfoque produce cero o una respuesta satisfactoria.

Si las dos técnicas anteriores para resolver NHL2 fallan, se accede al ARP global. En la etapa S5A, se determina si la dirección NHL3 está en el dispositivo de enfoque actual o no.

Si NHL3 no está en el dispositivo actual, en la etapa S6, el proceso despacha una consulta para encontrar la entrada FDB de la capa 2 para NHL2 para obtener el puerto de salida. La generación de una consulta en la capa 2 se describe más adelante. Si tiene éxito, se añade el puerto de salida al registro de ruta (S6A), la topología de caché 3 se utiliza para encontrar el puerto y el dispositivo en el extremo del enlace (S7), el dispositivo se añade a la ruta (S7A), y el dispositivo de enfoque se establece en el dispositivo que acaba de situarse en el extremo del enlace (HOP L2). Las etapas S6A, S7 y S7A pueden citarse como un HOP L2. En este punto, véase la figura 2b. En la etapa S5A, el dispositivo de enfoque es el dispositivo A. Este recibe una consulta para encontrar la entrada de la capa 2 FDB y devuelve el puerto de salida. El dispositivo que se determina que es en el extremo de ese enlace es el dispositivo B (figura 2c) que recibe una consulta con NHL3 todavía establecida en la dirección IP de destino.

Si no se ha encontrado una entrada de la capa 2 FDB, o si está en S5A, se determinó que la NHL3 estaba alojada en el dispositivo de enfoque, en la etapa S8, se realizó una consulta de ruta para determinar si la ruta L3 se encuentra en el dispositivo de enfoque en la dirección IP de destino. La consulta de la ruta puede ser una consulta de una sola ruta o de ruta recursiva - que se describen más adelante. Esto establece un siguiente salto IP y una interfaz de salida. Si no se encuentra la ruta L3, una ruta rota se indica y se detiene el proceso - S8A. En la etapa S9 (HOP L3), la interfaz de tabla de enrutamiento de salida se añade a la ruta, NHL3 se ajusta a la nueva dirección IP de salto siguiente, y el proceso consulta el dispositivo para determinar la dirección de la capa 2 de NHL3. Si NHL2 no se puede resolver, NHL2 se ajusta a "desconocida".

En la etapa S10, la dirección NHL3 actual se compara con la dirección IP de destino. Si NHL3 no es la IP de destino (es decir, el algoritmo de identificación de ruta aún no está en el segmento L2 final), en la etapa S11, se utiliza la topología en caché para encontrar el puerto y el dispositivo en el extremo del enlace, el dispositivo se añade al registro de ruta y el foco se ajusta en este dispositivo. El proceso entonces consulta (S12) si el dispositivo de enfoque es el dispositivo terminal. Si el dispositivo de enfoque no es el dispositivo terminal, el proceso vuelve a la etapa S5, pero utilizando la NHL3 y la NHL2 establecidas en la etapa 9.

Terminación

El algoritmo termina cuando se alcanza el dispositivo terminal y el puerto terminal y el servidor de destino se añaden a la ruta. Otras condiciones de terminación evitan que el algoritmo esté en bucle de manera indefinida. En cada iteración de la ruta, una iteración comienza estableciendo un indicador conmutado a falso y un indicador dirigido a falso. Cuando se produce un salto L2 (S7) el indicador de conmutación se establece en verdadero; cuando se produce un salto L3 (S9) el indicador dirigido se establece en verdadero. Como ya se ha mencionado, el puerto de salida se determina a partir de un dispositivo de enfoque y la topología de la red se utiliza para encontrar el dispositivo conectado y el puerto de entrada del dispositivo conectado. Para cada iteración, la combinación de:

"Dispositivo de enfoque, NHL2, NHL3" se almacena.

Si el dispositivo de enfoque, NHL2 o NHL3 han cambiado y la nueva combinación de "dispositivo de enfoque, NHL2, NHL3" se ha visto antes, se activa un evento de bucle detectado y el bucle se detiene. Si no se ha alcanzado el límite de bucle, y se ha producido el enrutamiento o la conmutación (es decir, los indicadores dirigidos o conmutados son verdaderos) y el dispositivo de enfoque no es igual al dispositivo terminal, se vuelve a iterar. Cada vez se evalúa si se ha alcanzado el límite de bucle de iteración. Si es así, el algoritmo termina.

Cuando cesa la iteración, si el dispositivo de enfoque es el dispositivo terminal, se añade el dispositivo terminal a la ruta. Si el dispositivo de enfoque no es el dispositivo terminal, pero el algoritmo se ha detenido, se informa de un error como el algoritmo de búsqueda de la ruta se habrá terminado en una situación inesperada. Si el dispositivo terminal es un conmutador de acceso, se añade el puerto de salida del conmutador de acceso desde "localizar destino" (S4) a la ruta y el dispositivo de destino derivado del puerto de salida del conmutador de acceso se añade a la ruta - el algoritmo termina entonces. Si el dispositivo terminal es igual al dispositivo de destino, el algoritmo termina. El detalle del algoritmo se describirá ahora con más detalle.

Ejemplo específico

La figura 4 muestra un resultado de la operación del algoritmo de identificación de ruta. Es decir, se proporciona la ruta para la que un paquete de datos desde el dispositivo de origen X dirigido al dispositivo de destino Y se haría cargo de la red en el momento en el que el algoritmo de identificación de ruta consulta la red. La ruta se muestra para incluir dispositivos A-J que forman parte del registro de ruta. El registro de ruta incluye los puertos de entrada y salida de cada uno de dichos dispositivos.

Mirando de nuevo a la red original de la figura 1, la primera parte del registro de ruta que se muestra en la figura 4 se puede ver que se deriva de la red de la figura 1, donde las letras correspondientes se han utilizado para indicar los dispositivos seleccionados por el conmutador anterior o dispositivo de enrutamiento. Cuando opera el algoritmo de identificación de ruta, el dispositivo B de enrutamiento había determinado enviar el paquete al conmutador C. Sin embargo, sin el uso de la presente invención, habría sido extremadamente difícil de trabajar en tiempo real. El dispositivo de enrutamiento B de manera similar tenía una opción para encaminar el paquete al enrutador F en la red central. Mediante la consulta del dispositivo de enrutamiento B en tiempo real (o más o menos tiempo real), basado en paquetes hipotéticos dirigidos al destino Y, el dispositivo de enrutamiento B devuelve la decisión que habría hecho si hubiera llegado un verdadero paquete con esa dirección. Determinando que el dispositivo de enrutamiento B enviará el paquete al conmutador C y, a continuación, se establece que el conmutador C se ha conectado en el extremo alejado de su dispositivo de enrutamiento del puerto de salida D, C y D se han añadido al registro de la ruta 81. De esta manera, el algoritmo de identificación de paquetes se ha intensificado a través de la ruta que el paquete hipotético habría tomado en el momento que el algoritmo de identificación de ruta consulta los dispositivos en la red. El dispositivo de enrutamiento D del cuadro adyacente representa la configuración de NHL3 y NHL2, es decir, NHL3 se establece en la dirección IP del dispositivo E que se ha establecido como el dispositivo de extremo para el dispositivo de enrutamiento D basado en la tabla de enrutamiento actualmente activa en D, y la NHL2 se ha

establecido como la dirección MAC para el dispositivo E mediante el dispositivo de consulta D para su entrada ARP para el dispositivo E.

Topología de la red

Como se mencionó previamente, la topología de la red incluye la interconectividad del dispositivo de red y la ubicación del servidor final. La topología de la red 3 puede proporcionarse mediante un servidor de topología que proporciona detalles de las conexiones de puerto a puerto. Por lo tanto, cuando un puerto de salida se identifica en un dispositivo, el puerto de entrada del dispositivo conectado puede determinarse mediante la conexión de puerto a puerto identificada en la topología. Ambos puertos de salida y de entrada se pueden añadir al registro de ruta. El servidor de topología también proporciona un CAM global, un ARP global y credenciales del dispositivo. Además, para cada dispositivo registrado en la topología hay preferiblemente una lista de índice de interfaz (IfIndex), y una lista de VLAN (red de área local virtual). Los dispositivos VLAN no se han descrito aún. Se describen también en el presente documento. Cuando se devuelve una respuesta al ordenador monitor 16, el ordenador monitor consulta la topología 3 en el orden siguiente cuando la gestión de la capa 2 responde. En este contexto, una respuesta de la capa 2 es una respuesta que ha identificado un puerto de salida de un dispositivo conmutador de la capa 2. El orden de la consulta es CDP, LLDP, STP y SONMP, IPv6 ND.

Localizar el dispositivo de origen

Como se mencionó anteriormente, la ubicación del primer dispositivo en la ruta (el dispositivo conectado al dispositivo de origen) no es necesariamente sencilla. En una realización, el ordenador monitor 16 implementa el algoritmo para tratar en primer lugar de encontrar el origen como un servidor conectado y, si esto falla, trata de encontrar el origen como un dispositivo de red. Cuando se trata de encontrar la fuente como un servidor conectado, consulta el dispositivo de origen para la dirección de la capa 2 (MAC) para la IP de origen. Esto se puede lograr de la misma manera como la consulta en un dispositivo de enfoque como se describe anteriormente en la etapa S5B. Es decir, el proceso envía una consulta para encontrar la entrada ARP para la dirección IP de origen.

Si no hay una dirección de la capa 2 desde el dispositivo de origen, se consulta la tabla de caché ARP global en el servidor de topología. En la realización descrita, estas se denominan como tablas ARP, pero se pueden utilizar cualquier tabla que asigne direcciones de capa 3 a capa 2. Si se encuentra una dirección MAC que corresponde a la dirección IP de origen, el servidor de topología se consulta para el origen de ubicación de IPs MAC mediante la consulta de tablas de reenvío globales en caché de la capa 2 en el servidor de topología para encontrar puertos que han visto tráfico desde esta dirección MAC. Se espera que el servidor de topología vuelva a una ubicación MAC de origen única mediante la eliminación de múltiples coincidencias (la MAC de origen vista en muchos puertos), mediante el filtrado de puertos marcados como troncos, puertos con un número excesivo de MACs (entradas FDB de puertos de conmutación de acceso tienen típicamente una sola dirección MAC "vista"), puertos con topología entre la red (por ejemplo, si un puerto tiene información de adyacencia CDP no puede ser un puerto en un conmutador de acceso), etc.

Si la fuente no se puede encontrar como un servidor conectado, se hacen intentos para encontrar el origen como un dispositivo de red. Esto se puede lograr mediante la consulta del servidor de topología para todas las direcciones IP que se encuentran en todos los dispositivos de red gestionados para ver si la dirección IP se encuentra en un dispositivo de red. Si es así, que el dispositivo de red se establece como el dispositivo de enfoque.

Localizar el dispositivo de destino

Consideraciones similares se aplican a la ubicación del dispositivo de destino. En primer lugar, se hacen intentos para encontrar el dispositivo de destino como un servidor conectado, y si eso no funciona, se hacen intentos para encontrar el destino como un dispositivo de red. Para encontrar el dispositivo de destino como un servidor conectado, el dispositivo de destino es consultado por su dirección de la capa 2, o las tablas de asignación de la capa 3 a la capa 2 en caché global se consultan en el servidor de topología (del mismo modo que para el dispositivo de origen descrito anteriormente). Entonces se consultan las tablas de reenvío de la capa 2 en caché global en el servidor de topología para encontrar puertos que han visto tráfico de esta MAC (de nuevo, como se ha descrito anteriormente con referencia a la ubicación del dispositivo de origen).

Para encontrar el destino como un dispositivo de red si lo anterior falla, el servidor de topología pueden consultarse para todas las direcciones IP que se encuentran en todos los dispositivos gestionados para ver si la dirección IP se encuentra en un dispositivo de la red. El dispositivo de red puede establecerse como el dispositivo terminal.

Utilidad por salto

Para implementar el algoritmo de identificación de ruta, el ordenador monitor 16 ejecuta un programa de ordenador como se ha descrito. Este programa informático proporciona una utilidad que se encarga de consultas "por salto". Es decir, el algoritmo de identificación se basa en el envío de una consulta desde el ordenador monitor a un dispositivo de enfoque y la recepción desde el dispositivo de enfoque de un puerto de salida que se puede utilizar para acceder

a la topología. Esto no puede necesariamente alcanzarse mediante una sola consulta. Como se describió anteriormente, el algoritmo requiere un siguiente salto inicial en la capa 3 (NHL3). La utilidad intenta consultar una tabla de enrutamiento en el dispositivo de origen para NHL3 y el puerto de salida, utilizando la dirección IP de destino. Si no se encuentra ninguna ruta, se consulta la tabla de enrutamiento en el conmutador de acceso en caso de que sea un conmutador de la capa 3 (que es el primer dispositivo conectado al dispositivo de origen para NHL3). Si no se encuentra ninguna ruta, el dispositivo de origen se consulta para la puerta de enlace predeterminada para NHL3. Si no se encuentra ninguna ruta, el primer dispositivo se consulta para una puerta de enlace predeterminada.

Para consultar una tabla de enrutamiento para encontrar NHL3 (como se describió anteriormente), una ruta se encuentra para la dirección IP en cuestión (la dirección IP 'buscada') mediante la consulta del dispositivo de enrutamiento utilizando una técnica de modulación especulativa que se describirá más adelante. Si se encuentra la ruta, pero no se especifica un puerto de salida, la siguiente dirección IP de salto se devuelve y se utiliza como NHL3. Si no se encuentra la ruta con una interfaz de salida iflndex mayor que cero, el puerto de salida se devuelve con la dirección de NHL3 y se añade el puerto de salida a la ruta. Si la ruta se encuentra con la interfaz de salida ifIndex igual a cero, la utilidad se reitera mediante el establecimiento de la IP solicitada al siguiente salto de IP (a partir de la consulta anterior) y la búsqueda de la ruta para la IP solicitada consultando el dispositivo que utiliza modulación especulativa (como se describe posteriormente). Esto se repite hasta que el iflndex devuelto sea distinto de cero.

La etapa de encontrar la ruta para la IP buscada utiliza la técnica de modulación especulativa para devolver una entrada de ruta. Si se encuentra la entrada de la ruta, la utilidad sondea la dirección del siguiente salto desde ipRouteNextHop.NetworkAddress. La utilidad también sondea la interfaz de salida desde ipRouteIflindex.NetworkAddress y la encuesta para ipRouteType.NetworkAddress. Si ipRouteType es "directo", la IP buscada se establece en el siguiente salto, como un tipo de ruta IP directa indica que está conectada directamente al segmento de red.

Es posible que varias coincidencias serán devueltas a partir de una tabla de enrutamiento en un dispositivo. En ese caso, es apropiado determinar si se están utilizando múltiples rutas, por ejemplo, cuando un dispositivo es responsable de tráfico de equilibrio de carga. Si activamente solo se utiliza una única ruta, la ruta activa debe determinarse. Si se utilizan múltiples rutas, la ruta podría dividirse en este punto y el registro de ruta podría contener los resultados del algoritmo buscador de ruta aplicado a todos y cada ruta que se encuentra desde este punto en adelante. En muchos casos, múltiples opciones de enrutamiento en un dispositivo son indicativas de un dispositivo que de forma inteligente se dirige en base a varias métricas. Estas métricas también se pueden consultar y regresar para el registro en el ordenador monitor.

La utilidad también es responsable de encontrar el siguiente salto inicial en la capa 2 mediante la consulta de la tabla de asignación 91 de la capa 3 a la capa 2 en el dispositivo de enfoque. Si no se encuentra la dirección de la capa 2, donde el dispositivo de enfoque es el dispositivo de origen, la utilidad consulta el conmutador de acceso (si se trata de un conmutador de la capa 3 debe proporcionar una asignación de la capa 3 a la capa 2). Si no se encuentra la dirección de la capa 2, la utilidad consulta las tablas ARP en caché global en el servidor de topología 3. Una consulta de una dirección de la capa 2 en un dispositivo se realiza como se ha explicado anteriormente con referencia a la etapa S5B.

Si la dirección NHL 3 no está en el dispositivo de enfoque, la utilidad sondea el dispositivo de enfoque para un puerto de salida para la dirección de la capa 2 NHL2. La etapa de sondeo del dispositivo de enfoque para el puerto de salida NHL2 incluye sondeo específico de la VLAN (red de área local virtual). Es decir, incluye la etapa de establecer en qué VLANs participa el dispositivo de acuerdo con la topología 3 y cómo se registra en el dispositivo. Estas VLANs se utilizan para ayudar a encontrar entradas de la tabla de reenvío de VLANs específicas (FDBs a menudo se dividen de acuerdo a qué VLAN está relacionada - por ejemplo, para protocolo de árbol de extensión VLAN (PVSTP) es necesario para realizar las consultas FDB en el contexto de cada VLAN para tratar de encontrar una coincidencia).

Si no se encuentra el puerto de salida desde la capa 2 FDB (usando una VLAN específica o la VLAN nativa), entonces la utilidad intenta encontrar qué cabezas de interfaz hacia NHL2 desde los registros ARP sondean para ipNetToMediaPhysAddress 71 (figura 7). Es decir, la utilidad intenta aprender qué interfaz de relación de la capa 2 a la capa 3 se aprendió.

Una vez que la utilidad ha encontrado un puerto de salida usando la dirección de la capa 2, se añade el puerto de salida al registro de ruta y utiliza el servidor de topología 3 para encontrar el puerto remoto conectado al puerto de salida. Este puerto remoto se registra como el puerto de entrada en el siguiente dispositivo.

Canales de puerto/puertos multiplexados

Si no se encuentra un puerto remoto, o el nombre del puerto de salida exige el uso de los puertos de capas más altas o más bajas a continuación, la utilidad comprueba los puertos de la capa inferior o los puertos de la capa superior. Es decir, puede haber un escenario en el que hay una asignación de las salidas de ruta de acceso virtual a puertos físicos. Para que el algoritmo de identificación de ruta de acceso tenga éxito, tiene que identificar un puerto

de salida físico para acceder al servidor de topología. En un escenario donde la comprobación de los puertos de capa inferior revela la presencia de puertos de capa inferior, estos puertos de capa inferior se pueden usar como los puertos de salida y se accede al servidor de topología para encontrar los puertos remotos (puertos de entrada del dispositivo siguiente) unido a los puertos de salida. En este punto, la ruta se divide en varias rutas separadas, cada una de las cuales se traza de forma independiente desde este punto.

Si se identifican los puertos de capa superior, el puerto de capa superior se utiliza para el puerto de salida. El servidor de topología se utiliza para encontrar el puerto remoto conectado a este puerto de salida de capa superior.

Siguiente salto

Se establecen los indicadores dirigidos y conmutados a falso. Usando el servidor de topología o consultas directas con el dispositivo de enfoque, se determina si el dispositivo de enfoque aloja la dirección IP NHL3 en cualquiera de sus puertos. Si lo hace el servidor de la dirección IP NHL3, entonces la utilidad luego pasa a consultar la tabla de enrutamiento del dispositivo de enfoque para rutas a la IP de destino mediante el uso de la técnica de modulación especulativa. Si la utilidad localiza una ruta candidata, la siguiente dirección de la capa 2 NHL2 se establece mediante la consulta del dispositivo de enfoque (o tablas ARP en caché global) para la asignación de la capa 3 a la capa 2 y el indicador dirigido se establece en verdadero. Si NHL3 es igual a la IP de destino, a continuación, indica que la utilidad ha alcanzado el último dispositivo de la capa 3 más próximo al destino, de modo que no hay necesidad de mover más este dispositivo, ya que el siguiente salto sería un salto de capa 2. Por lo tanto, la utilidad añade puertos de salida de la ruta candidata a la ruta. Si NHL3 no es igual a la IP de destino, que indica que no está en el segmento final de la capa 2 y se añade el puerto de salida de la ruta candidata a la ruta.

Si no se ha producido ningún enrutamiento durante esta iteración (la indicación dirigida todavía se ajusta a falso), a continuación, la utilidad sondea el dispositivo de enfoque para un puerto de salida para la dirección de la capa 2 NHL2. La etapa de sondeo del dispositivo de enfoque para el puerto de salida NHL2 incluye sondeo específico de la VLAN (red de área local virtual) (como se ha descrito anteriormente). Si no se encuentra el puerto de salida desde la capa 2 FDB (usando una VLAN específica o la VLAN nativa), entonces la utilidad intenta encontrar qué cabezas de interfaz hacia NHL2 desde los registros ARP sondean para ipNetToMediaPhysAddress 71. Es decir, la utilidad intenta aprender qué interfaz de relación de la capa 2 a la capa 3 se aprendió. Una vez que la utilidad ha encontrado un puerto de salida usando la dirección de la capa 2, se añade el puerto de salida al registro de ruta y utiliza el servidor de topología 3 para encontrar el puerto remoto conectado al puerto de salida. Este puerto remoto se registra como el puerto de entrada en el siguiente dispositivo. Si se encuentra un puerto de salida usando cualquiera de las consultas FDB o consultas ARP, la señalización de conmutación se establece en verdadero.

Si, cuando se consulta el servidor de topología, no se encuentra ningún puerto remoto, o el nombre de puerto de salida exige el uso de puertos de capas superiores o inferiores, a continuación, se realiza una comprobación de los puertos de capa inferior o superior como se ha descrito anteriormente. Si se encuentra un puerto de salida, se añade a la ruta, se añade el dispositivo que contiene el puerto a la ruta y el dispositivo de enfoque se establece en el dispositivo remoto.

Esta etapa de "próximo salto" se repite hasta que se alcance un límite prescrito en el número de iteraciones o la ruta llega a su fin (es decir, no se produce la conmutación ni el enrutamiento).

Si el proceso termina en el dispositivo terminal previamente identificado y que el dispositivo es un conmutador de acceso, se añade el puerto de salida de "localizar destino" para el registro de ruta, y se añade el dispositivo de destino al registro de ruta. Si el dispositivo terminal es el propio dispositivo de destino, la utilidad termina.

Las figuras 11A a 11D muestran un diagrama de flujo de la operación de la utilidad ejecutada en el ordenador monitor.

Equilibrador de carga

Como se mencionó anteriormente, si el dispositivo de enfoque es el dispositivo terminal, se añade el dispositivo terminal con el destino al registro de la ruta. Si el terminal tiene un equilibrador de carga, entonces se obtiene la asignación de la IP virtual al grupo de servidores para el equilibrador de carga. Esto permite al servidor asignar el servidor físico al equilibrador de carga para ser identificado. La ruta se retiene hasta la ruta "raíz" (hasta el dispositivo equilibrador de carga). Luego, para cada dirección IP del servidor físico, una utilidad adicional de descubrimiento de ruta se ejecuta desde el equilibrador de carga a la dirección IP del servidor físico, con cada ruta adicional pendiente de manera previa con la ruta "raíz".

Consulta de la tabla de enrutamiento

Uno de los factores que hacen que el algoritmo de la ruta sea particularmente eficaz es la capacidad de generar una consulta a un dispositivo de enrutamiento de manera eficiente, es decir, generar una consulta a la que el dispositivo de enrutamiento puede responder en un corto período de tiempo sin sobrecarga significativa. La figura 5 ilustra la

estructura de una tabla de rutas lineales direccionable a través de SNMP. Para establecer una ruta a un destino particular, ipRouteDest es el índice requerido en la tabla de rutas. Esto se indica mediante 48 en la figura 5. Las entradas de interés en la tabla son ipRoutelfindex 50 que define la interfaz de salida, ipRouteNextHop 52 que define la dirección IP del siguiente salto (siguiente salto IP) y ipRouteType 54 que define el tipo de enrutamiento de entrada (inválido/directo/indirecto). El acceso a la tabla normalmente requiere el conocimiento de la ipRouteMask 56: esto permitiría localizar una dirección IP de red específica. Sin embargo, como puede verse en la figura 5, la propia ipRouteMask está incrustada en la ipRouteEntry y, por lo tanto, no se conoce para establecerse en la consulta. Lo que se requiere es encontrar una coincidencia para:

<IP of interest> y <ipRouteMask.X> == <ipRouteDest.X>

para encontrar la clave ipRouteDest 48 que representa el índice para la tabla.

La figura 27 ilustra el proceso

Como se ha observado por los inventores, solo hay 33 posibilidades para la ipRouteMask (/32.../0), es decir, 255.255.255.255, 255.255.255.254, 255.255.255.252, ... 0.0.0.0. Varios de los mismos producen IDs de red duplicados para la misma dirección IP, debido al número de ceros en la dirección IP. Una lista de las 33 posibles máscaras de red se produce (Z2), y se aplica a la dirección IP (Z3). La figura 6 muestra la aplicación de las 33 máscaras de red a la dirección IP 10.44.1.213 = OA.2C.01.D5 = 0000 1010 0010 1100 0000 0001 1101 0101.

Esto genera 12 valores únicos (con la etiqueta 32, 31, 29, 27, 25, 24, 23, 13, 12, 10, 6, 4). Por lo tanto, ahora solo es necesario hacer 12 consultas SNMP (que se pueden presentar en un solo paquete de consulta) para encontrar la ruta. Después de las etapas Z4 y Z5 para determinar si se permiten las rutas por defecto y eliminar las redes en consecuencia, los 12 resultados se comparan en la tabla de rutas del dispositivo de enfoque y cuando se encuentra una coincidencia de los elementos necesarios ipRoutelfIndex (egresslfindex), ipRouteNextHop y ipRouteType se recuperan (Z12) y se devuelven en una respuesta al ordenador monitor 16.

La interfaz resultante se establece en egressInterface (Z13).

La reducción en el número de consultas necesarias para encontrar la ruta se denomina en este documento "modulación especulativa" y permite la realización de consultas en tiempo real de tabla de rutas de una manera muy eficiente.

Al examinar las tablas de enrutamiento reales, no es raro que la ruta se encuentre para una dirección IP dada que no tiene una interfaz de salida válida y única para proporcionar una dirección del siguiente salto. En estos casos, la dirección del siguiente salto se utiliza para una consulta posterior de la tabla de enrutamiento para tratar de obtener una interfaz de salida para la próxima dirección de salto. Esta reutilización de la dirección del siguiente salto se repite hasta que se obtiene una interfaz de salida. De acuerdo con este enfoque, en una primera etapa, para encontrar sola consulta de rutas, se utiliza modulación especulativa para encontrar una entrada de enrutamiento para la dirección IP especificada (IP_X) como se acaba de señalar. Si el ipRouteType asociado es "directo", IP_X (y $ipRoutelfIndex_X$) se devuelven en una respuesta al ordenador monitor como el siguiente salto. Es decir, está conectado directamente y, por lo tanto, no tiene ningún siguiente salto de la capa 3.

Si el ipRouteType asociado no es directo, ipRouteNextHop y ipRoutelfIndex se devuelve en respuesta al ordenador monitor.

El proceso de descubrimiento de rutas también tiene en cuenta las tablas de enrutamiento entre dominios sin clase IP, que son más difíciles de consultar. En este caso, si la etapa Z10 no se traduce en una dirección IP, el proceso se mueve a la etapa Z14, donde una consulta SNMP (Obtener siguiente) se emite al dispositivo, utilizando IPcidrRouteDest + dirección de red + máscara de red. Si el resultado no es una dirección IP, el proceso vuelve a la etapa Z7 y pasa a través de las etapas Z8, Z9, Z10 de nuevo. Si el resultado es una dirección IP, la dirección de red se extrae del OID devuelto. Se determina entonces si la dirección de red del OID coincide con la dirección de red de la consulta. Si no es así, el proceso vuelve a la etapa Z7. Si no es así, la ruta encontrada se establece para ser verdadera, la clave CIDR se establece en el OID de la consulta devuelta con IPcidrRouteDest eliminado, es decir, el índice en la tabla de rutas CIDR. El proceso pasa a continuación para permitir una consulta SNMP para obtener el siguiente salto, el iflndex de salida y el tipo de ruta.

Como se muestra en la figura P, en el proceso FindRoutelterative, la etapa F1 FindRoute se toma para la dirección IP requerida (IP_x). Si no se encuentra ninguna ruta, se devuelve un error. Si se encuentra una ruta, pero no hay ninguna interfaz de salida, se devuelve ipRouteNextHop. Si no se encuentra la ruta y el ipRoutelfindex es igual a cero, entonces una etapa FindRoutelterative posterior se toma para la dirección IP de ipRouteNextHop, con los mismos cuatro resultados posibles.

Aunque la modulación especulativa es una técnica particularmente buena para la consulta eficaz de grandes conjuntos de datos, su aplicabilidad principal es cuando se consultan datos que se indexan con una clave derivada

para la que ya se conoce una clave parcial. Es por eso que es particularmente útil en el contexto del análisis de la tabla de rutas SNMP y de la tabla de consulta SNMP ARP. Sin embargo, el rápido comportamiento de reenvío por dispositivo de red también puede determinarse utilizando otras técnicas realizando, por ejemplo, acceso CLI y API XML.

Consulta ARP

Ahora se hará referencia a la figura 7 para describir una técnica eficiente para consultar una tabla ARP que utiliza modulación especulativa. La generación de una consulta se describe con más detalle más adelante con referencia a la figura 7. Para el dispositivo que se consulta, se obtiene una lista de índices de interfaz (IfIndices) a partir de la topología de red o caminando IfIndex desde el propio dispositivo. Cada ifIndex para el dispositivo se combina con la dirección NHL 3 para generar un conjunto de claves para incluir en la consulta al dispositivo. Por lo tanto, una consulta que contenga estas claves se formula y se transmite al dispositivo de enfoque. El dispositivo de enfoque produce cero o una respuesta satisfactoria. La figura 7 ilustra un formato de tabla IpNetToMediaEntry que, en principio, permite que la dirección MAC se determine para cualquier dirección IP determinada. Desde una entrada única que no se puede encontrar para una dirección IP específica a menos que se sepa de qué interfaz aprendió la entrada ARP, la modulación especulativa se utiliza mediante la combinación de la dirección IP con todos y cada ifIndex en el dispositivo. Es decir, cada clave de consulta puede crearse mediante la combinación de la dirección IP con un ifIndex. De esta manera el número de consultas SNMP es el número de interfaces del dispositivo, que es típicamente mucho menos que el número de entradas ARP en el dispositivo y, por lo tanto, es mucho más eficiente.

En las claves especulativas, varias claves de consulta pueden estar contenidas en un único mensaje de consulta.

Sigue una descripción de algoritmos alternativos para la identificación de ruta. A continuación, se hará referencia a la figura 12 para describir un proceso de iteración en bucle. La entrada en el bucle principal se indica en la parte superior de la figura 12 mediante la flecha de entrada 4. La flecha de entrada 4 indica el estado al final de los procesos de preparación que se describirán más adelante. El estado de entrada comprende:

<Opciones, dispositivo de enfoque, NHL 3, NHL 2, VLAN>

Estos elementos se establecen por los procesos de preparación que se describirán más adelante. Estos se denominan a continuación como "variables de estado". La variable de estado titulada "opciones" tiene una secuencia ordenada de opciones de bucle. En el presente ejemplo, la secuencia ordenada comprende CSRAcr.

La variable de estado, titulada "VLAN" es el identificador de la red de área local virtual (número) de la VLAN con la cual el paquete hipotético está etiquetado actualmente en este punto en la ruta.

En la etapa L01 del bucle, la primera opción (cabeza de lista) se selecciona y se ejecuta. Estas opciones se describen más adelante. Después de la ejecución de la opción, el proceso vuelve a regresar al punto L02 y se crea un nuevo estado (L03), siguiendo las etapas de procesamiento implementadas por la cabeza de la opción de la lista. Se determina (L04) de si este estado se ha producido antes, y si no, el estado se almacena (L05). Si el estado se ha producido antes, se genera un informe de "bucle descubierto" y el límite del bucle se establece en cero, lo que tendrá por efecto la ruptura del bucle.

En la etapa L07, se disminuye el límite de bucle. Si el límite de bucle es menor que o igual a cero o no hay más opciones disponibles en la secuencia de opciones, o el dispositivo de enfoque es igual al dispositivo terminal, entonces se establece una condición de "terminación es igual a verdadero". En la etapa L08, se realiza una comprobación de la condición de terminación, y si la condición de terminación es verdadera, termina el bucle principal. De lo contrario, vuelve al punto de entrada de bucle principal usando el nuevo estado que fue creado en la etapa L03.

Obsérvese que, en la ejecución de cada opción en una iteración del bucle, la primera etapa en la ejecución de la opción es eliminar esa opción de la secuencia ordenada en la variable de estado de opciones.

Al final de las etapas de procesamiento de la opción, puede ser que esa opción se restablezca de nuevo en la secuencia, o puede haber sido eliminada de forma permanente, dependiendo de la opción y de los resultados de las etapas de procesamiento.

Obsérvese también que, en la ejecución de las etapas de procesamiento de una opción, las otras variables de estado (dispositivo de enfoque, NHL3, NHL3, VLAN) pueden alterarse de forma individual o en total. La alteración de cualesquiera variables de estado resulta en un nuevo estado, que puede constituir un nuevo estado de entrada para una siguiente iteración del bucle.

Ahora se hará referencia a la figura 12 para describir la segunda parte del proceso de bucle principal. En la etapa L09 se determina si se ha alcanzado o no el dispositivo terminal esperado. Si lo ha hecho, en la etapa L10 se determina si el terminal tiene un conmutador de acceso conectado al IP del servidor. Si no lo es, entonces el proceso

termina después de que se indica un descubrimiento de ruta completa con éxito y devuelve una ruta completa. Si el dispositivo terminal es un conmutador de acceso conectado a la IP del servidor, se añade la conexión del puerto de acceso y la IP del servidor a la ruta, y entonces el proceso pasa a completar el descubrimiento de la ruta con éxito y devuelve una ruta completa antes de terminar deteniéndose.

Si en la etapa L09 se determina que el dispositivo terminal esperado no se ha alcanzado, entonces en la etapa L14, se preguntó si NHL3 es la IP del servidor o no. Si no lo es, entonces se determina que el descubrimiento de la ruta completa no ha tenido éxito, y una ruta parcial se devuelve antes de la detención. Si NHL fue la IP del servidor, esto indica que el proceso en el segmento L2 final, por lo que el proceso puede saltar hasta el destino mediante el incremento del contador de segmentos de salto y estableciendo el dispositivo de enfoque en NHL3. La figura 14 ilustra la Opción C. En la primera etapa C1, la opción se elimina de la secuencia en la variable de opciones. En la etapa C2, se realiza una comprobación para una única interfaz de red cuya dirección coincide con el destino (IP del servidor). Si es así, esto indica que el algoritmo ha llegado a la porción de red de conmutación final, y NHL3 se establece en la IP del servidor. En la etapa C3, el dispositivo de enfoque se consulta para encontrar la entrada ARP para la IP del servidor, y NHL2 se establece en el resultado. El proceso vuelve entonces al bucle principal (C4) para permitir que otra opción decida la interfaz de salida. La consulta al dispositivo de enfoque se realiza mediante SNMP a la tabla ARP en el dispositivo de enfoque, o si no se encuentra, se consulta el sistema de gestión de la red ARP en caché. Estas consultas son de acuerdo con técnicas más plenamente descritas más adelante.

En la etapa C2, si la comprobación para la interfaz única para la IP del servidor de destino falla, no existe ninguna interfaz única identificada. Las variables de estado no se actualizan en absoluto. En este caso, todo lo que hemos hecho es la opción "C" evaluada (y descartada) y pareció que es improductiva.

La figura 15 ilustra opción S. Según la etapa S101, la opción S se elimina de la secuencia ordenada. En la etapa S102, se realiza una consulta al sistema de gestión de red o mediante consultas SNMP al dispositivo de enfoque para encontrar si el dispositivo de enfoque aloja NHL3. Si NHL3 está en el dispositivo de enfoque, el proceso vuelve al punto de retorno del bucle principal (S104). Si la dirección de enrutamiento NHL3 no está en el dispositivo de enfoque, se determina si se establece el NHL2 de la dirección de conmutación, y el dispositivo de enfoque se consulta en la tabla de asignación (ARP) para el NHL2 dado el NHL3. Las consultas se realizan como se describe más completamente más adelante. En la etapa S106 se determina si se establece o no el indicio de VLAN indirecto. Indicios de VLAN se discutirán más adelante. Si no, se determina una lista de las VLANs en el dispositivo de enfoque, ya sea desde el dispositivo de enfoque o desde el sistema de gestión de red. Una VLAN se selecciona de la lista y una búsqueda de la entrada de la base de datos de reenvío se realiza utilizando el dispositivo de enfoque, NHL2 y VLAN. La búsqueda de la entrada de base de datos de reenvío que se ilustra en la etapa S109 se muestra en un diagrama de flujo en la figura X. Volviendo a la etapa S106, si el indicio de VLAN se establece a continuación, el proceso pasa directamente a la etapa 110, que es una búsqueda de la entrada FDB como en la etapa S109 usando el indicio de VLAN que la VLAN ha consultado dentro de la FDB. En la etapa S112 (también S111), se determina si hay o no hay una entrada encontrada en la base de datos de reenvío. Si la hay, el proceso pasa a la segunda parte de la opción S que se muestra en la figura 16 (flecha de entrada 5). Si después de la etapa S111, no se encuentra ninguna entrada FDB, se entra en un bucle de VLAN hasta que se determina si existe una entrada FDB o que el proceso debe volver al bucle principal. El punto de entrada a la segunda parte de la opción S se muestra en la flecha 5 en la parte inferior de la figura 15. Esto también se muestra en la parte superior de la figura 16. Como se describe, anteriormente, si se encuentra una entrada de base de datos de reenvío, esto indica el puerto de salida para la ruta S115. Esto se puede utilizar para obtener el siguiente dispositivo conectado desde la topología de la red, como se muestra en la etapa S117, y se describe más completamente más adelante. La etapa S116 es la etapa de obtener un indicio de VLAN que se muestra en la figura 2 y se describirá más adelante.

La etapa S117 se muestra en la figura 22, que es un diagrama de flujo que ilustra el proceso para la obtención del puerto conectado y, por lo tanto, el dispositivo conectado posterior del sistema de gestión de red basado en el puerto de salida regresa desde la base de datos de reenvío: Si en la etapa S118 se encuentra el puerto conectado, el puerto conectado y dispositivo conectado se añaden a la ruta identificada (etapa S119) y en la etapa S120 el dispositivo de enfoque se cambia al dispositivo conectado. En la etapa S121, las opciones de bucle se restablecen a CSRAcr. A continuación, el procesador vuelve al bucle principal en la etapa S122. Volviendo a la etapa S118, si no se encuentra el puerto conectado, el proceso salta adelante a NHL3, incrementando un contador de segmento de salto 89 (figura 8) y cambiando el dispositivo de enfoque a NHL3. El contador de segmento de salto se implementa en el ordenador de administración de hardware, firmware o software y permite a los segmentos de la ruta que se distingan por ser saltados cuando está claro que el siguiente dispositivo conectado no puede determinarse fácilmente a partir de las etapas de proceso anteriores. En la etapa S124, se determina si el dispositivo de enfoque no es el destino (IP del servidor). Si no es así, las opciones de bucle se restablecen a CSRAcr en la etapa S125. Si el dispositivo de enfoque es el servidor de destino, en la etapa 126 se determina si el destino está en un conmutador de acceso conocido, y si es así, NHL 3 se establece en la dirección del conmutador de acceso al servidor. Después de configurar las opciones de bucle en la etapa S125, en la etapa S128 el dispositivo de enfoque realiza una consulta para NHL2 utilizando la dirección NHL3 que fue creada en la etapa S127, mediante la consulta de la tabla ARP del dispositivo de enfoque o un NMS. Debe tenerse en cuenta que la opción S incluye la etapa de eliminarlo de las opciones disponibles en la etapa S101, y luego restablecerse de nuevo en las opciones disponibles en la etapa S121 y la etapa S125 en base a los resultados de las etapas de procesamiento.

Ahora se hará referencia a la figura 17 para describir las opciones R y r. Cada una de esas opciones comienza con la eliminación de esa opción de la secuencia ordenada en la opción variable de la etapa R1, r1. En la etapa R2, r2, un proceso es instigado a buscar la ruta a la IP de destino (IP del servidor) utilizando un proceso de búsqueda de rutas iterativo que se ilustra en la figura P. En la opción R, el proceso opera donde no hay ruta por defecto permitida (ruta por defecto permitida es igual a falso). En la opción R, el proceso permite una ruta por defecto (la ruta por defecto permitida es igual a verdadero). Si no se encuentra ninguna ruta, etapa R3, el procesador vuelve al bucle principal. Si se encuentra una ruta, a continuación, se envía una consulta al dispositivo de enfoque mediante un NHL3 candidato, que se ha determinado a partir de la tabla de enrutamiento desde la que se encontró la ruta. Esta consulta es la tabla ARP del dispositivo para determinar un NHL2 candidato que corresponde al NHL3 candidato. Si no se encuentra ningún NHL2 candidato, el proceso se mueve al punto de entrada 7 para la segunda parte de la opción R/r. Si un NHL2 candidato se encuentra a partir de la consulta ARP, a continuación, se realiza una comprobación en la etapa R8 para determinar si el NHL2 candidato es el mismo que el NHL2 que se registra como la variable de estado en el estado de la entrada para el proceso R/r. Si son iguales, entonces el proceso pasa al punto de entrada 6. Si no lo son, siguiendo la etapa R8, si el NHL2 candidato no es el mismo que el estado de entrada NHL2, a continuación, NHL3 se establece en siguiente salto de IP de la ruta candidata, y NHL2 se establece en el NHL2 candidato. En la etapa R10 se determina si las consultas de la tabla de enrutamiento en R2 y r2 también proporcionan un puerto de salida. Si es así, el proceso prosigue al punto de entrada 6. Si no es así, las opciones se restablecen a CSRAcr en la etapa R11. La figura 18 ilustra el punto de entrada 6 en la parte superior de la figura. En la siguiente etapa R12, se determina si la tabla de enrutamiento proporciona un puerto de salida. Si no, el proceso vuelve al bucle principal. En la etapa R13, el proceso de obtener un indicio de VLAN se realiza de acuerdo con la figura 21 y se describirá posteriormente.

A continuación, el proceso de obtener el puerto conectado se realiza como se ilustra en la figura 22. En la etapa R15, se determina si se encuentra o no el puerto conectado. Si lo es, el puerto de salida, el puerto conectado, y el dispositivo conectado se añaden a la ruta. El dispositivo de enfoque se cambia al dispositivo conectado y las opciones de bucle se restablecen en CSRAcr. Si no se encuentra un puerto conectado, no se hace nada en esta etapa. El proceso pasa a la etapa R20, donde se determina si NHL3 se ha actualizado. Si no es así, el proceso vuelve al bucle principal. Si es así, el dispositivo de enfoque anterior se consulta para el candidato NHL2, dado el candidato NHL3 de la tabla de enrutamiento. Si, después de esta etapa, la NHL2 se ha resuelto, el proceso vuelve al bucle principal. Si no es así, el nuevo dispositivo de enfoque se consulta para el candidato NHL 2 dado el candidato NHL3.

Se hará referencia a la opción c con referencia a la figura 19. En la primera etapa c1, c se elimina de las opciones en la variable de estado. En la etapa C2 se realiza una comprobación para una única interfaz, cuya dirección de red coincide con la dirección de red NHL3. Si no se encuentra una interfaz única, el proceso vuelve al bucle principal. Si se encuentra una interfaz única, y es un nombre de interfaz de salida, se realiza el proceso de indicio de VLAN que se describirá con referencia a la figura 21. Después de la etapa C4, se obtiene el puerto conectado empleando el proceso de obtener el puerto conectado que se muestra en la figura 22.

En la etapa c7, se determina si se encuentra un puerto emparejado. Si es así, el puerto de salida, el puerto emparejado y el dispositivo emparejado se añaden a la ruta y el dispositivo de enfoque se fija en el dispositivo emparejado. Las opciones disponibles se restablecen a CSRAcr en c8. Si ningún puerto emparejado se encuentra en C7 paso, el proceso vuelve al bucle principal.

Ahora se hará referencia a la figura 20 para describir la Opción A. En la etapa A1, la opción A se retira de la secuencia de opción en la variable de estado. En la etapa A2, se encuentra la entrada ARP SNMP para asignación de NHL3 a NHL2. Si no se encuentra ninguna asignación, el proceso vuelve al bucle principal.

Si se encuentra una asignación, el proceso utiliza SNMP para encontrar el iflndex de la interfaz desde la que se aprendió la relación. En la etapa A5, se determina si una interfaz única se ha encontrado o no. Si no es así, el proceso vuelve al bucle principal. Si es así, el proceso pasa a la etapa A6, donde se determina si hay un nombre de interfaz de salida disponible. Si lo hay, el proceso de indicio de VLAN es instigado como se describirá con referencia a la figura 21. Luego, en A8, se instiga el proceso de obtener el puerto conectado, como se ilustra en la figura 22.

Si como resultado del proceso de obtener el puerto conectado se encuentra un puerto emparejado, se añade el puerto de salida a la ruta, el puerto emparejado y el dispositivo emparejado se añaden a la ruta y el dispositivo de enfoque se fija en el dispositivo emparejado. Además, las opciones disponibles se restablecen a CSRAcr. Si no se encuentra ningún puerto emparejado, el proceso vuelve al bucle principal.

Ahora se hará referencia a la figura 21 para explicar el proceso de indicio de VLAN. Este proceso se utilizó en la Opción A en la etapa A7, en la Opción C en la etapa c5, en la Opción R/r en la etapa R13 y en la Opción S en la etapa S116. Además, se utiliza en uno de los procesos de preparación que aún no ha sido abordado. El proceso comienza en la etapa VL1 con un nombre de puerto de acceso. En la etapa VL2 se determina si el nombre está en la forma de "VL + un número", y si el número se extrae y se almacena como un indicio de VLAN en la etapa VL3.

Si no es así, entonces en el sistema de gestión de la red se solicita una lista de todas las VLANs en la interfaz. La

etapa VL5 comprueba cualesquiera VLANs en conflicto. Si no hay ninguna, entonces la única VLAN se almacena como un indicio de VLAN en la etapa VL6. Si hay VLANs en conflicto, cualquier indicio de VLAN que ya ha sido almacenado se deja sin cambios.

Los indicios de VLAN abordan un problema que puede surgir en determinadas redes que utilizan una técnica de conmutación llamada STP (protocolo de árbol de extensión) que se utiliza para evitar tener bucles lógicos en porciones conmutadas (capa 2) de la red (para evitar el bucle de tráfico alrededor de manera infinita). El protocolo se utiliza para decidir a dónde un dispositivo de conmutación debe enviar un paquete dado.

Es decir, si el dispositivo se está conmutando, el conmutador mirará al encabezado de la capa 2 (MAC/Ethernet) y mirará a la dirección de la capa 2 de destino (NHL2) y luego consultará una base de datos interna (la FDB - base de datos de reenvío) para determinar desde cuál de sus puertos se debe enviar el paquete. Muchas empresas utilizan una extensión de STP llamada PVSTP (protocolo de árbol de expansión por VLAN), por lo que cada paquete está marcado con un identificador de VLAN también. Entonces el conmutador mantiene FDBs separadas - una por cada VLAN.

Esto se hace en parte por eficiencia y en parte para permitir topologías virtuales más complejas. Así, es perfectamente posible (y no es raro) que dos paquetes con la misma capa 2 de destino abandonen por diferentes puertos, ya que están etiquetados como que están en diferentes VLANs, a pesar de que su destino sea el mismo dispositivo/puerto.

La consecuencia de esto es que el proceso no puede simplemente arrastrar todos los FDBs por VLAN hasta que se encuentra una coincidencia. Es importante saber a priori qué VLAN del paquete se etiqueta como un miembro.

Este etiquetado de la VLAN puede producirse en diferentes lugares de la red, por ejemplo, en el puerto de acceso de origen - es decir, cuando el dispositivo de origen está conectado físicamente, o en otro lugar en la red - no es raro que una etiqueta de VLAN sea sustituida por otra (esto se llama enrutamiento entre VLAN).

Por ejemplo, dado un paquete que llega al dispositivo D de la red (en la ruta A->B->C->D) D solo puede ser consultado por la interfaz de salida correcta si se conoce la VLAN del paquete desde A estuviera en el punto cuando alcanza D. Podría ser, por ejemplo, que A coloque el paquete en la VLAN 100, B pase (usando la VLAN 100), entonces C cambie de 100 a 200 y luego D lo conmute usando la VLAN 200.

Por esta razón, es necesario 'llevar' un indicio de VLAN en toda la red desde nuestro dispositivo de origen a nuestro dispositivo de destino como parte del paquete hipotético que rastreamos. Así, en su caso el indicio de VLAN se usa, anula, restablece o actualiza.

Como ya se ha mencionado, la figura 23 ilustra el proceso de findRouteIterative que se utiliza en las opciones R y r. El proceso consiste en un bucle de ruta de hallazgo que comienza en la etapa F1 y termina en una verificación del límite de la ruta F2. El proceso findRouteIterative determina entonces si se ha encontrado una ruta y permite un índice de salida que se encuentra perteneciente a la ruta.

Antes de embarcarse en el bucle principal, hay tres procesos de preparación que se implementan para establecer el estado de entrada para la primera iteración del bucle. Un primer proceso de preparación se muestra en la figura 24, que establece como punto inicial el conmutador de acceso o dispositivo de red identificado por el dispositivo de origen (indicado en la figura 24 como el lado del cliente). Del mismo modo, un conmutador de acceso o dispositivo de red se almacena como un punto de detención, en base al dispositivo de destino, que se refiere en la figura 24 como el lado del servidor. En la Figura 24, la IP del cliente y la IP del servidor son el origen y destino respectivamente.

La figura 25 es un proceso de preparación para el establecimiento de las direcciones NHL3 y NHL2 de estado de entrada inicial.

La figura 26 ilustra un tercer proceso de preparación que establece el dispositivo de enfoque para el estado de la entrada inicial.

Obsérvese que el primer proceso de preparación de la figura 24 conduce al segundo proceso de preparación de la figura 25, y el segundo proceso de preparación de la figura 25 conduce al tercer proceso de preparación de la figura 26. El tercer proceso de preparación lleva al punto de entrada principal 4 del bucle principal que se muestra en la figura A.

Tecnologías/protocolos adicionales

El algoritmo de identificación de ruta de acceso cuando que se utiliza anteriormente proporciona una forma eficaz de identificar una ruta particular que es probable que tome un paquete o mensaje en particular a través de la red dispositivos interconectados que operan de acuerdo con los protocolos de red conocidos generalmente. Se

presentan situaciones en que, por una razón u otra, el algoritmo de identificación de ruta se encuentra con un reto particular. Algunos de estos retos se describen a continuación.

En algunos casos, la utilidad que se ejecuta en el algoritmo tiene que atravesar un segmento de red conmutado de etiqueta de múltiples protocolos (MPLS). Esto se logra mediante la búsqueda de la asignación de etiquetas inicial (en el punto donde el tráfico entra en el segmento MPLS) y el seguimiento a través de la red MPLS por saltos usan detalles de salto de la etiqueta que aparece, se empuja y se reenvía hasta que el tráfico tiene su etiqueta final aparecida y abandona el segmento MPLS.

Otro reto es atravesar límites NAT que pueden realizarse mediante tablas NAT de sondeo del dispositivo NAT. Esto puede requerir el sondeo especulativo en tiempo real para NAT dinámica, pero podría ser posible utilizar el sondeo de fondo para NAT estática.

Para los protocolos de túnel como IPSEC/GRE/SSL, etc., la utilidad comprueba una ruta directa desde un extremo del túnel al otro (típicamente con un salto de la capa 3 desconocido que representa todos los nodos entre medio). La utilidad también comprueba la información topológica específica del protocolo y comprueba las tablas/interfaces de enrutamiento para la presencia de saltos criptográficos/de túnel.

Otro reto es la virtualización. Es importante que el algoritmo identifique puertos de salida físicos, de manera que un dispositivo físico conectado al puerto de salida se pueda acceder desde la topología. Muchas redes operan en varias capas diferentes de virtualización. Los conmutadores virtuales se pueden consultar utilizando APIs adicionales y para garantizar que el servidor de topología tiene información oportuna sobre la ubicación del servidor final, podría ser necesario que el servidor de topología se integre con plataformas de gestión de virtualización para obtener actualizaciones sobre la reubicación de la máquina virtual para permitir un sondeo proactivo de ubicación del servidor final en los conmutadores virtuales afectados.

La utilidad negocia enrutamiento virtualizado y tablas de reenvío (VRF) mediante la consulta de la IP de reenvío apropiada (tabla de enrutamiento) requerida para un identificador VRF específico. En SNMP, por ejemplo, esto se puede hacer usando cadenas de comunidad contextualizada de VRF.

REIVINDICACIONES

1. Un método implementado por ordenador de consulta de una tabla de reenvío de tráfico en un dispositivo en una red de ordenadores, teniendo la tabla de reenvío de tráfico entradas que son accesibles mediante una clave, comprendiendo el método:

identificar una dirección de reenvío para su uso en la consulta de la tabla de reenvío de tráfico, donde la dirección de reenvío constituye solo parte de una clave y donde una parte restante de la clave es un índice incrustado en la tabla de reenvío de tráfico;
combinar la dirección de reenvío de una pluralidad de índices integrados de la tabla de reenvío de tráfico para generar un conjunto de claves para consultar la tabla de reenvío de tráfico; y
generar un mensaje de consulta para la tabla de reenvío utilizando al menos uno de dicho conjunto de claves.

2. Un método de acuerdo con la reivindicación 1, donde la etapa de combinación de la dirección de reenvío con cada índice incrustado comprende combinar lógicamente una secuencia de bits que representa la dirección de reenvío con una secuencia de bits que representa el índice incrustado.

3. Un método de acuerdo con la reivindicación 1 o 2, donde la(s) clave(s) que se utiliza(n) en el mensaje de consulta generada(s) por el dispositivo de reenvío se seleccionan de entre el conjunto de claves mediante la selección de solo las claves únicas desde el conjunto de claves.

4. Un método de acuerdo con cualquier reivindicación anterior, donde la tabla de reenvío es una tabla de enrutamiento (92) y cada índice en la tabla de enrutamiento es una máscara de red (56).

5. Un método de acuerdo con la reivindicación 4, donde la tabla de reenvío es para un dispositivo de enrutamiento de capa 3 (12).

6. Un método de acuerdo con cualquiera de las reivindicaciones 1 a 3, donde el índice incrustado es un índice de interfaz de una tabla ARP (91) en un dispositivo de reenvío.

7. Un método de acuerdo con cualquier reivindicación anterior, donde el mensaje de consulta contiene una pluralidad de claves de consulta para consultar la tabla en paralelo.

8. Un método de acuerdo con la reivindicación 6, donde cada clave también comprende un identificador de puerto de salida del dispositivo.

9. Un ordenador (16) configurado para implementar un método de consulta de una tabla de reenvío de tráfico, comprendiendo el ordenador:

un procesador (80); y
una memoria (82), conteniendo la memoria un programa de ordenador que, cuando se ejecuta mediante el procesador (80), realiza las etapas de una cualquiera de las reivindicaciones 1 a 8.

10. Un producto de programa de ordenador en forma de un programa de ordenador que, cuando se ejecuta mediante el ordenador (16), implementa el método de una cualquiera de las reivindicaciones 1 a 8.

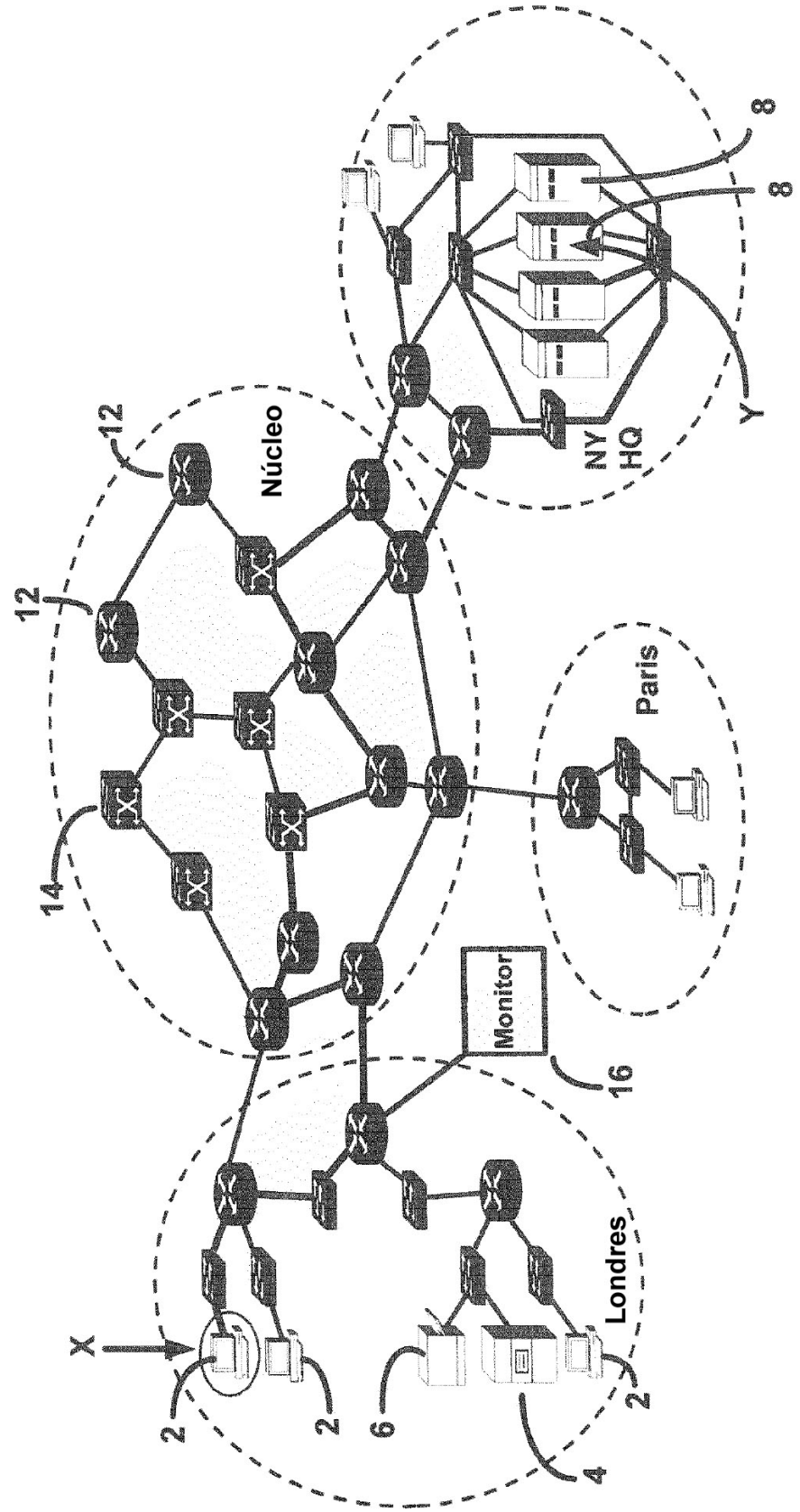


Fig. 1

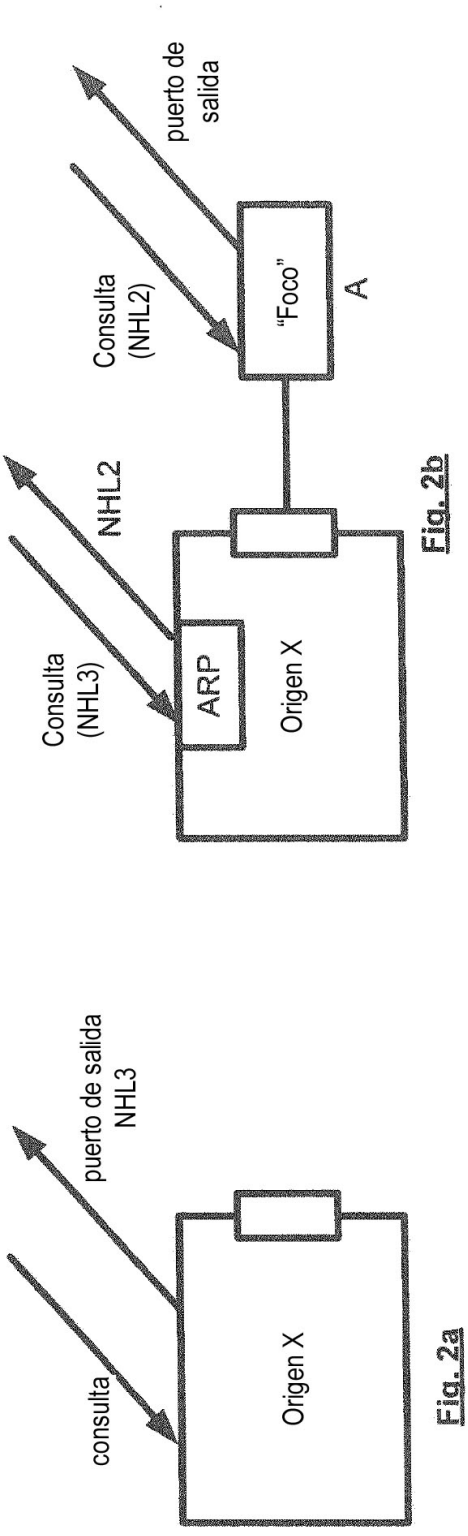


Fig. 2a

Fig. 2b

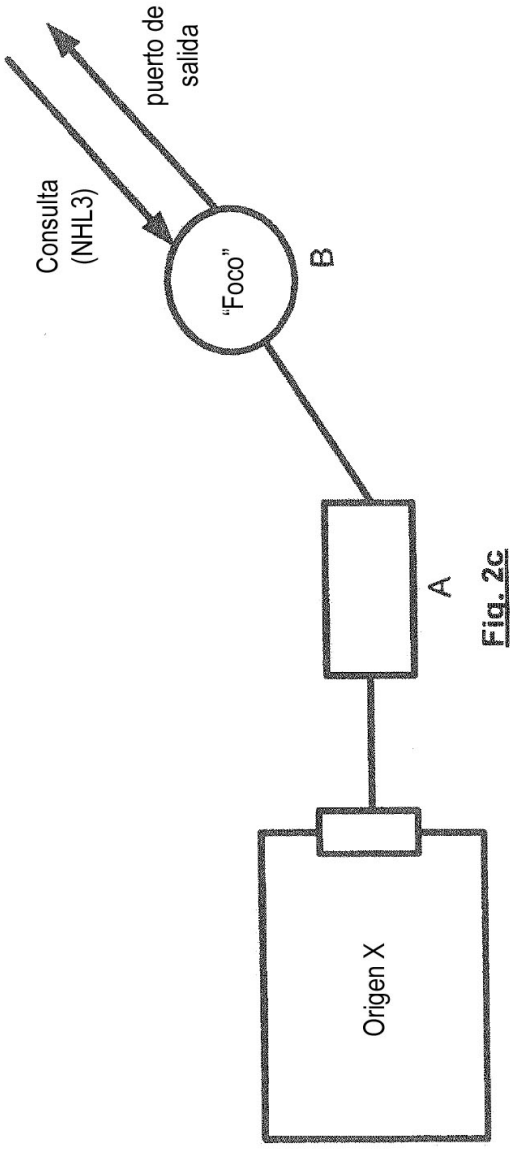


Fig. 2c

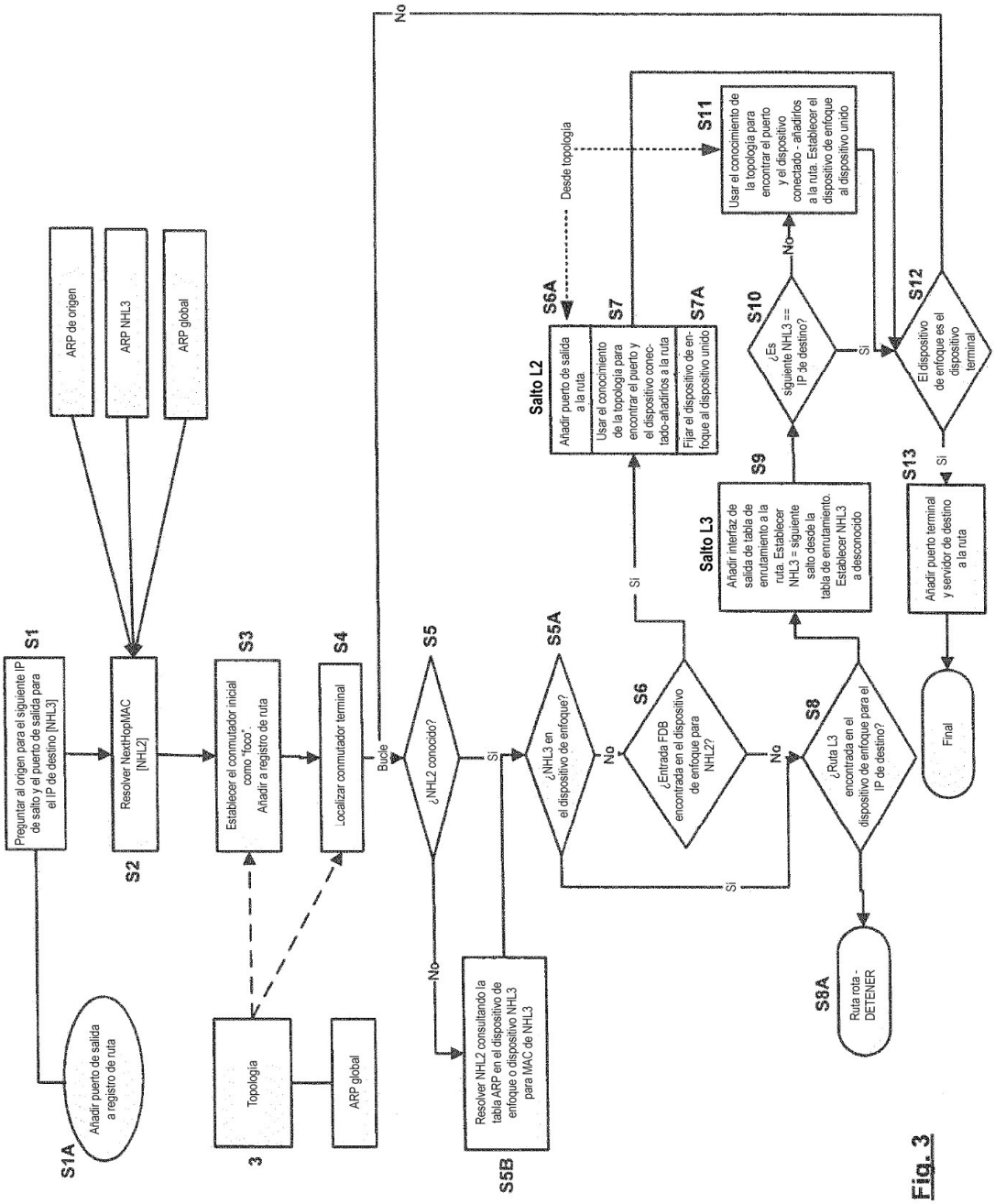

S1
Preguntar al origen para el siguiente IP de salto y el puerto de salida para el IP de destino [NHL3]
S1A
Añadir puerto de salida a registro de ruta
S2
Resolver NextHopMAC [NHL2]
ARP de origen
ARP NHL3
ARP global
S3
Establecer el conmutador inicial como 'foco'. Añadir a registro de ruta
3
Topología
ARP global
S4
Localizar conmutador terminal
Bucle
S5
¿NHL2 conocido?
No
S5B
Resolver NHL2 consultando la tabla ARP en el dispositivo de enfoque o dispositivo NHL3 para MAC de NHL3
Si
S5A
¿NHL3 en el dispositivo de enfoque?
No
S6
¿Entrada FDB encontrada en el dispositivo de enfoque para NHL2?
No
Si
Si
S8
¿Ruta L3 encontrada en el dispositivo de enfoque para el IP de destino?
No
S8A
Ruta rota - DETENER
Si
Salto L3
S9
Añadir interfaz de salida de tabla de enrutamiento a la ruta. Establecer NHL3 = siguiente salto desde la tabla de enrutamiento. Establecer NHL3 a desconocido
S10
¿Es siguiente NHL3 == IP de destino?
No
Si
S11
Usar el conocimiento de la topología para encontrar el puerto y el dispositivo conectado - añadirlos a la ruta. Establecer el dispositivo de enfoque al dispositivo unido
Desde topología
S12
El dispositivo de enfoque es el dispositivo terminal
Si
S13
Añadir puerto terminal y servidor de destino a la ruta
Final
Salto L2
S6A
Añadir puerto de salida a la ruta.
S7
Usar el conocimiento de la topología para encontrar el puerto y el dispositivo conectado-añadirlos a la ruta
S7A
Fijar el dispositivo de enfoque al dispositivo unido
No
Si

Fig. 3

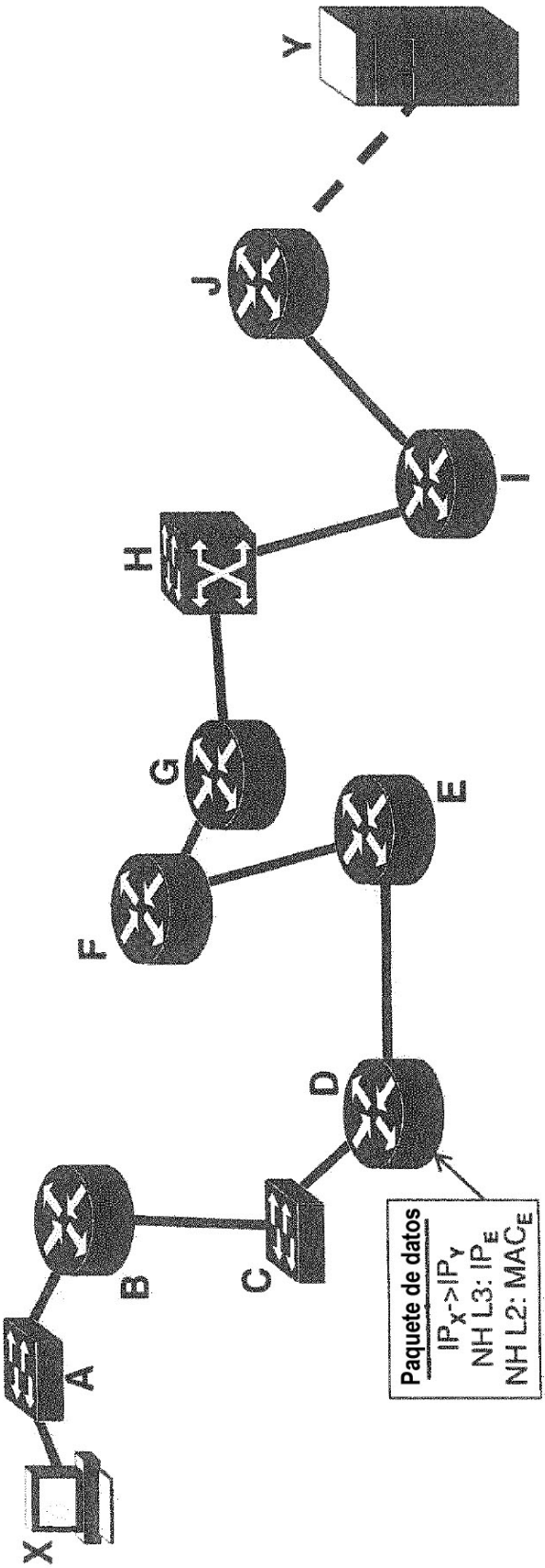


Fig. 4

```
ipRouteEntry TIPO DE OBJETO
-- 1.3.6.1.2.1.4.21.1
-- iso(1) org(3) dod(6) internet(1) mgmt(2) mib-2(1) ip(4) ipRouteTable(21) ipRouteEntry(1)
        SINTAXIS                    IpRouteEntry
        ACCESO                      no accesible
        ÍNDICE { ipRouteDest } ---------- 48
        DESCRIPCIÓN
        "Una ruta a un destino particular."
::= { ipRouteTable 1 }

IpRouteEntry
::=
  SECUENCIA {
        ipRouteDest                 IpAddress,
50 ---- ipRouteIfIndex              ENTERO .
        ipRouteMetric1              ENTERO .
        ipRouteMetric2              ENTERO .
        ipRouteMetric3              ENTERO .
        ipRouteMetric4              ENTERO .
52 ---- ipRouteNextHop              IpAddress,
54 ---- ipRouteType                 ENTERO .
        ipRouteProto                ENTERO .
        ipRouteAge                  ENTERO .
56 ---- ipRouteMask                 IpAddress,
        ipRouteMetric5              ENTERO .
        ipRouteInfo                 IDENTIFICADOR DE OBJETOS
        }
```

Fig. 5

Dirección IP 10.44.1.213 = 0A.2C.01.D5 = 0000 1010 0010 1100 0000 0001 1101 0101

/32:	0000	1010	0010	1100	0000	0001	1101	0101	/15:	0000	1010	0010	1100	0000	0000	0000	0000
/31:	0000	1010	0010	1100	0000	0001	1101	0100	/14:	0000	1010	0010	1100	0000	0000	0000	0000
/30:	0000	1010	0010	1100	0000	0001	1101	0100	/13:	0000	1010	0010	1000	0000	0000	0000	0000
/29:	0000	1010	0010	1100	0000	0001	1101	0000	/12:	0000	1010	0010	0000	0000	0000	0000	0000
/28:	0000	1010	0010	1100	0000	0001	1101	0000	/11:	0000	1010	0010	0000	0000	0000	0000	0000
/27:	0000	1010	0010	1100	0000	0001	1100	0000	/10:	0000	1010	0000	0000	0000	0000	0000	0000
/26:	0000	1010	0010	1100	0000	0001	1100	0000	/09:	0000	1010	0000	0000	0000	0000	0000	0000
/25:	0000	1010	0010	1100	0000	0001	1000	0000	/08:	0000	1010	0000	0000	0000	0000	0000	0000
/24:	0000	1010	0010	1100	0000	0001	0000	0000	/07:	0000	1010	0000	0000	0000	0000	0000	0000
/23:	0000	1010	0010	1100	0000	0000	0000	0000	/06:	0000	1000	0000	0000	0000	0000	0000	0000
/22:	0000	1010	0010	1100	0000	0000	0000	0000	/05:	0000	1000	0000	0000	0000	0000	0000	0000
/21:	0000	1010	0010	1100	0000	0000	0000	0000	/04:	0000	0000	0000	0000	0000	0000	0000	0000
/20:	0000	1010	0010	1100	0000	0000	0000	0000	/03:	0000	0000	0000	0000	0000	0000	0000	0000
/19:	0000	1010	0010	1100	0000	0000	0000	0000	/02:	0000	0000	0000	0000	0000	0000	0000	0000
/18:	0000	1010	0010	1100	0000	0000	0000	0000	/01:	0000	0000	0000	0000	0000	0000	0000	0000
/17:	0000	1010	0010	1100	0000	0000	0000	0000	/00:	0000	0000	0000	0000	0000	0000	0000	0000
/16:	0000	1010	0010	1100	0000	0000	0000	0000									

Fig. 6

ES 2 620 383 T3

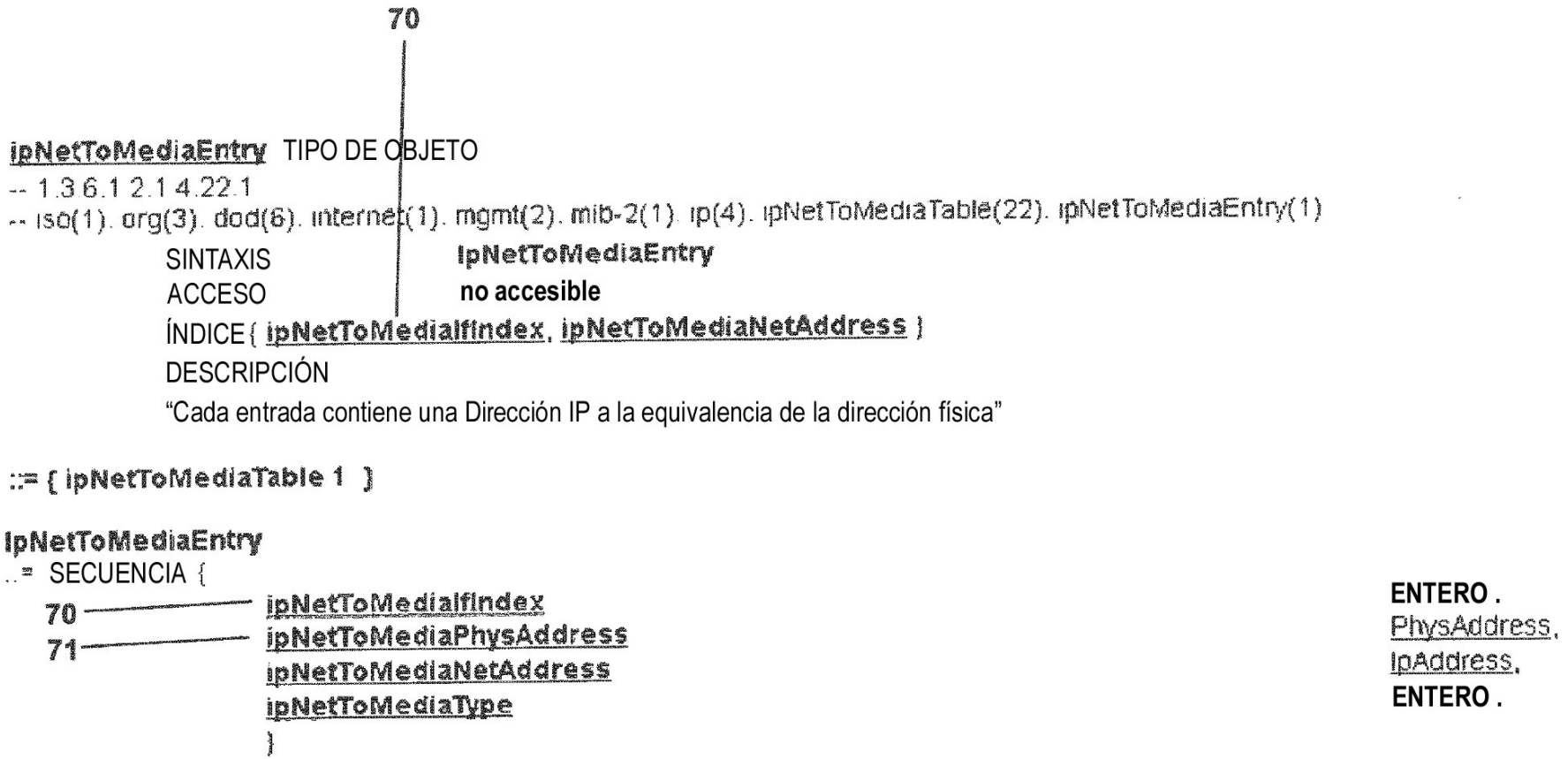

70
ipNetToMediaEntry TIPO DE OBJETO
-- 1.3.6.1.2.1.4.22.1
-- iso(1). org(3). dod(6). internet(1). mgmt(2). mib-2(1). ip(4). ipNetToMediaTable(22). ipNetToMediaEntry(1)
SINTAXIS ipNetToMediaEntry
ACCESO no accesible
ÍNDICE { ipNetToMediaIfIndex, ipNetToMediaNetAddress }
DESCRIPCIÓN
"Cada entrada contiene una Dirección IP a la equivalencia de la dirección física"
::= { ipNetToMediaTable 1 }
IpNetToMediaEntry
::= SECUENCIA {
70 ipNetToMediaIfIndex
71 ipNetToMediaPhysAddress
ipNetToMediaNetAddress
ipNetToMediaType
}
ENTERO .
PhysAddress,
IpAddress,
ENTERO .

Fig. 7

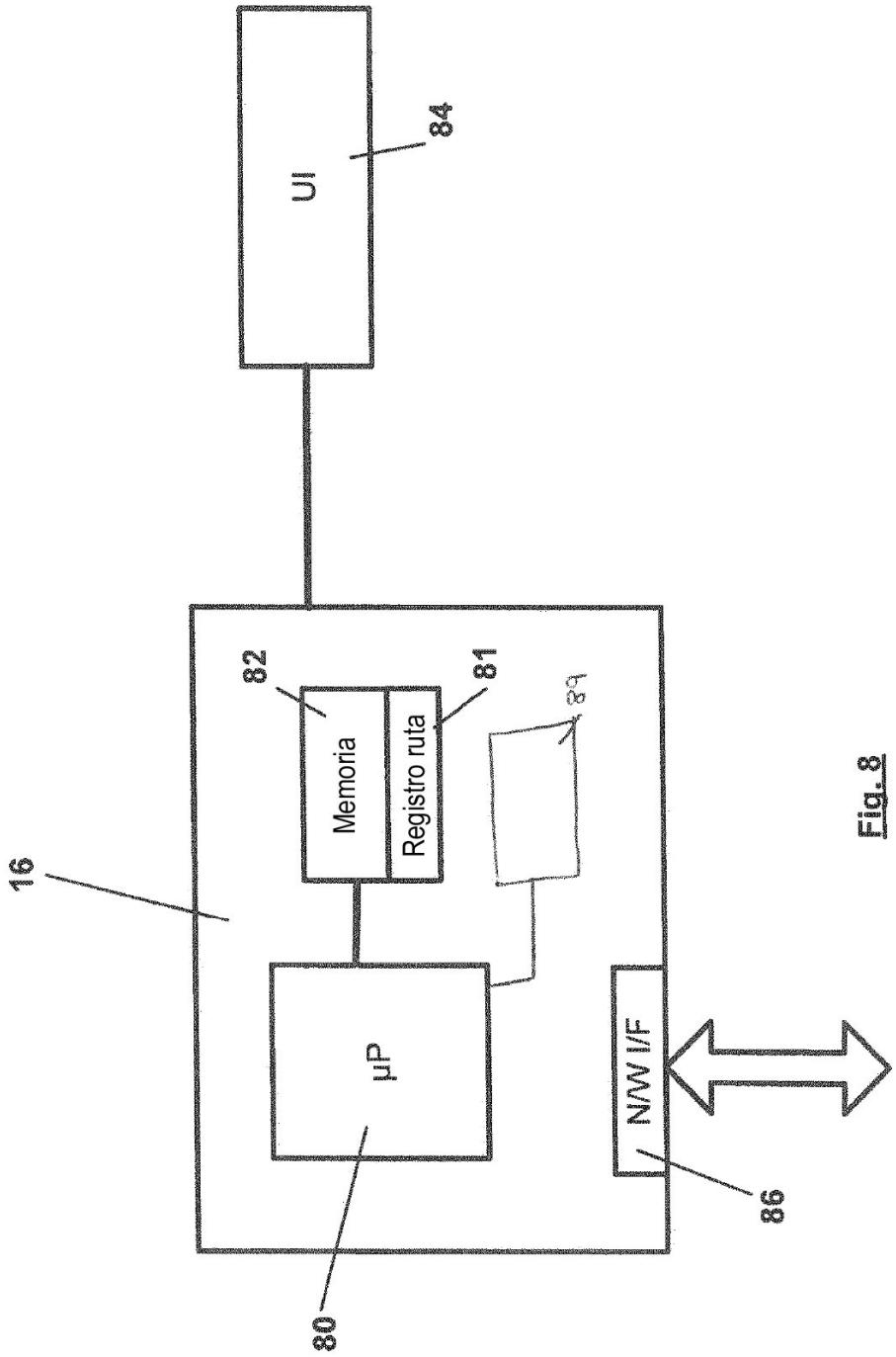


Fig. 8

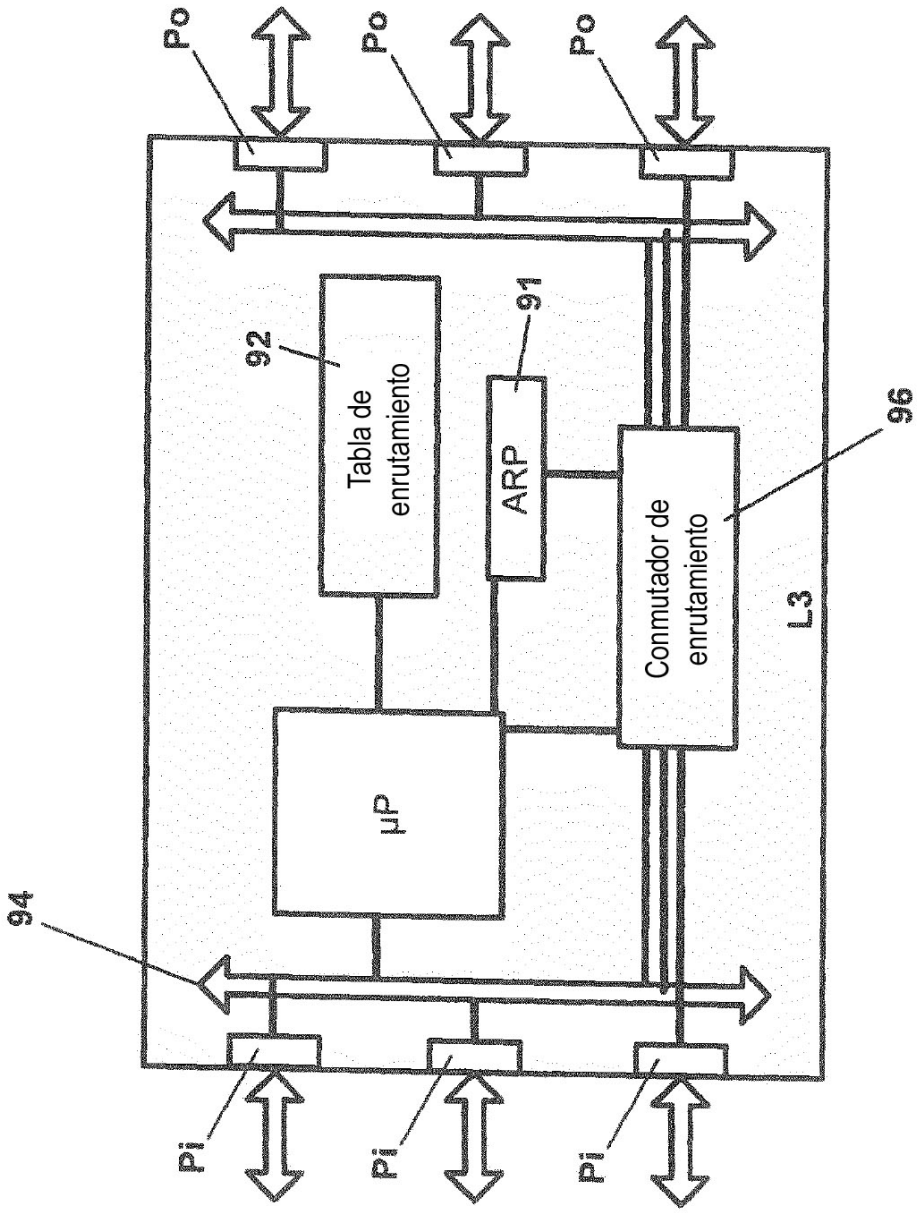

Po
Po
Po
92
Tabla de enrutamiento
91
ARP
96
Conmutador de enrutamiento
L3
μP
94
Pi
Pi
Pi

Fig. 9

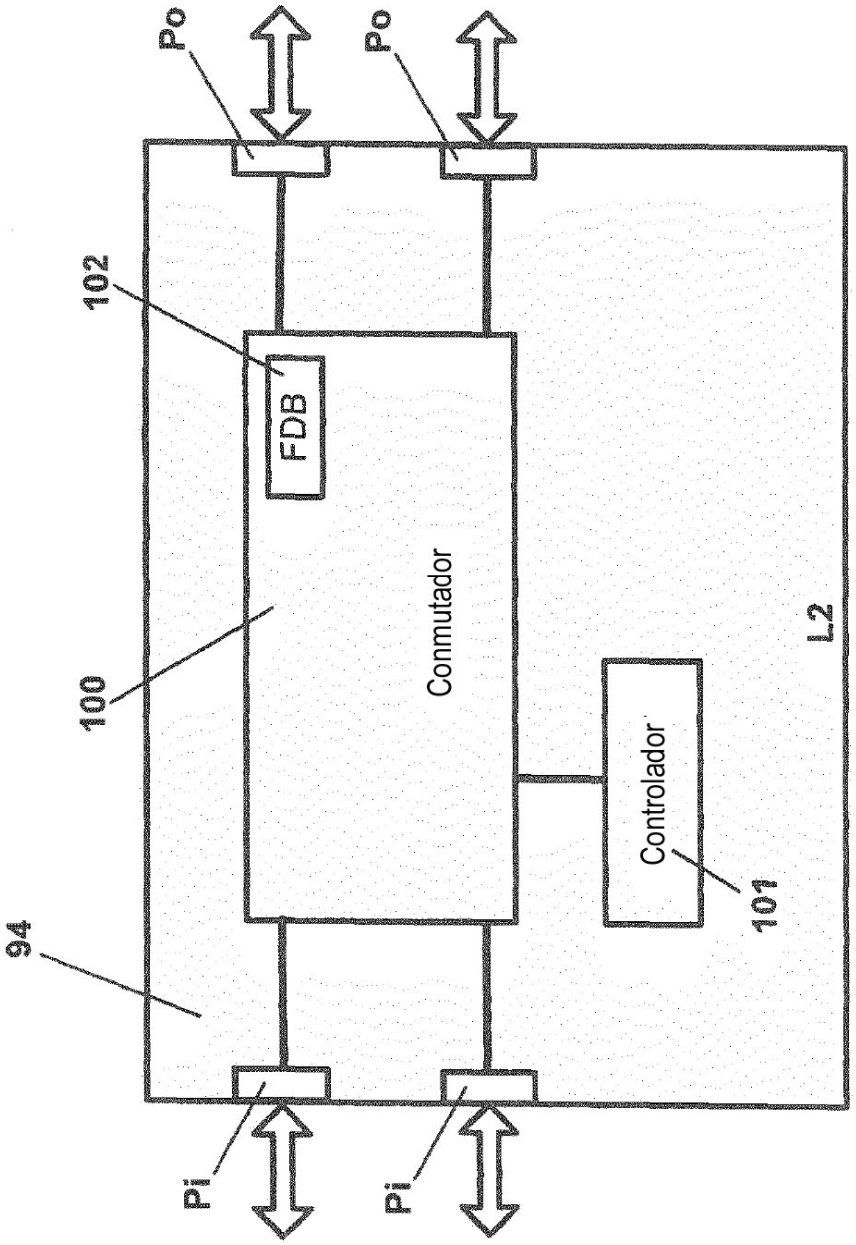


Fig. 10

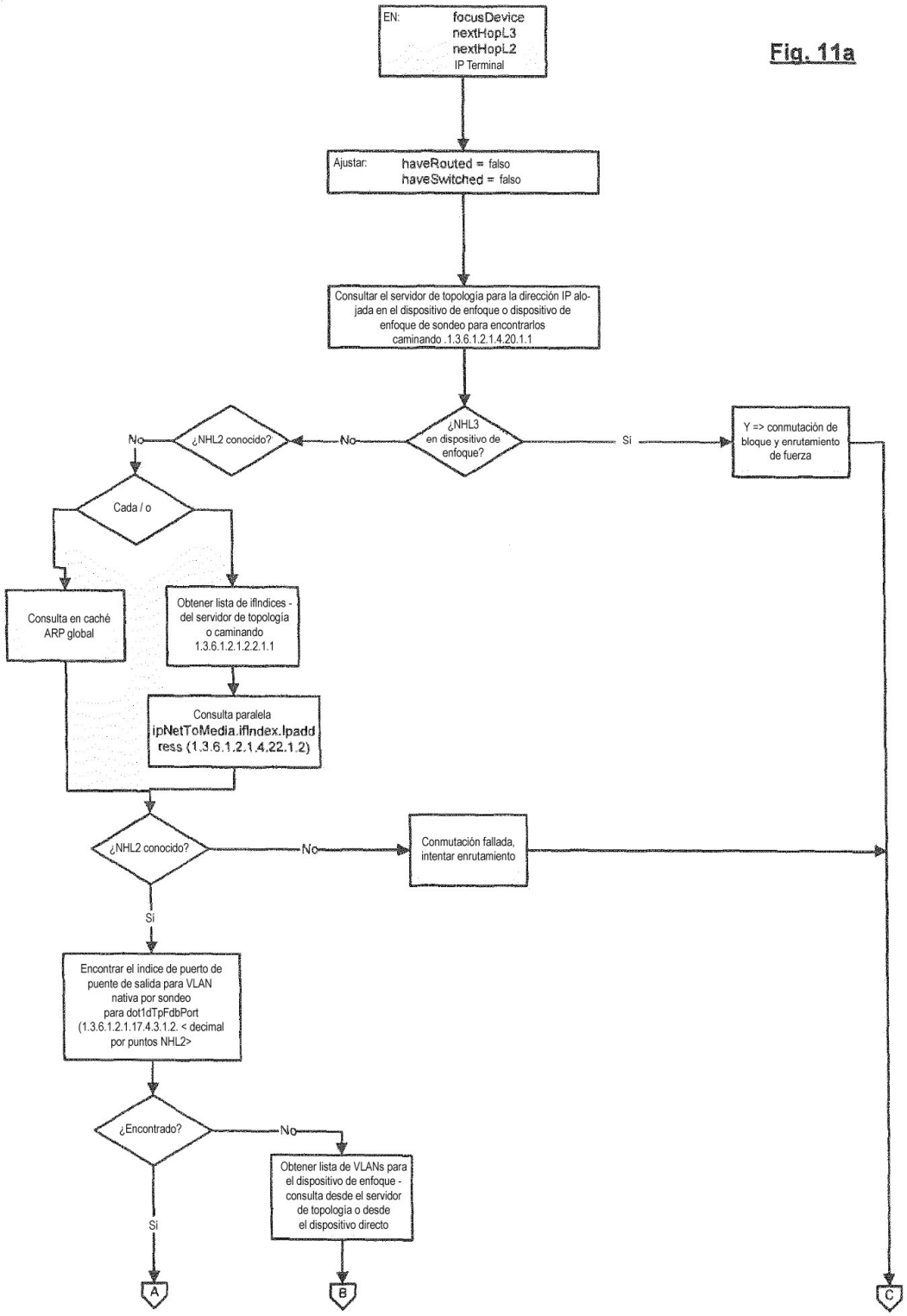

EN:
focusDevice
nextHopL3
nextHopL2
IP Terminal
Ajustar:
haveRouted = falso
haveSwitched = falso
Consultar el servidor de topología para la dirección IP alojada en el dispositivo de enfoque o dispositivo de enfoque de sondeo para encontrarlos caminando .1.3.6.1.2.1.4.20.1.1
¿NHL3 en dispositivo de enfoque?
No
Sí
Y => conmutación de bloque y enrutamiento de fuerza
¿NHL2 conocido?
No
Cada / o
Consulta en caché ARP global
Obtener lista de ifIndices - del servidor de topología o caminando 1.3.6.1.2.1.2.2.1.1
Consulta paralela ipNetToMedia.ifIndex.Ipaddress (1.3.6.1.2.1.4.22.1.2)
¿NHL2 conocido?
No
Conmutación fallada, intentar enrutamiento
Sí
Encontrar el índice de puerto de puente de salida para VLAN nativa por sondeo para dot1dTpFdbPort (1.3.6.1.2.1.17.4.3.1.2. < decimal por puntos NHL2>
¿Encontrado?
No
Obtener lista de VLANs para el dispositivo de enfoque - consulta desde el servidor de topología o desde el dispositivo directo
Sí
A
B
C

Fig. 11a

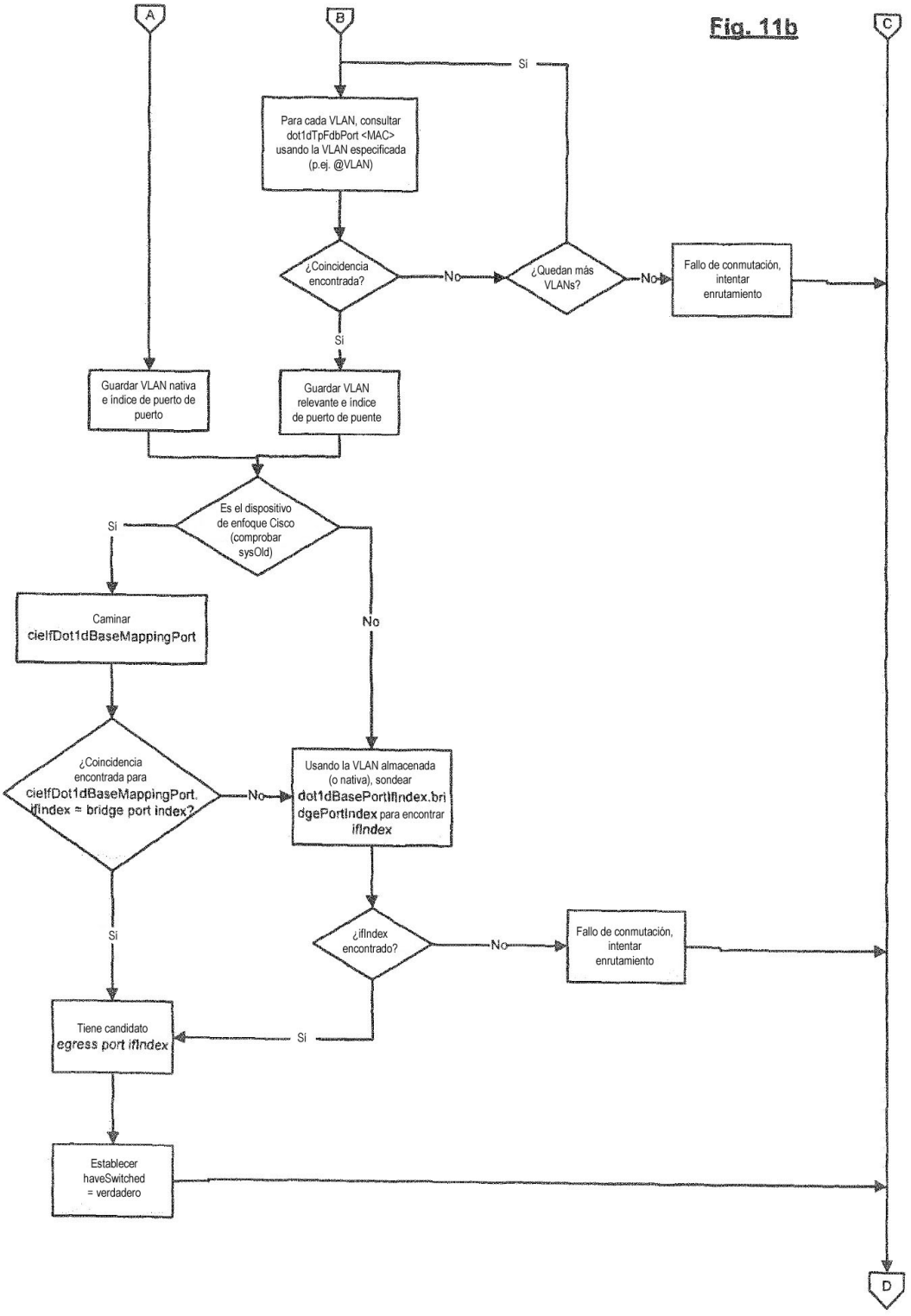
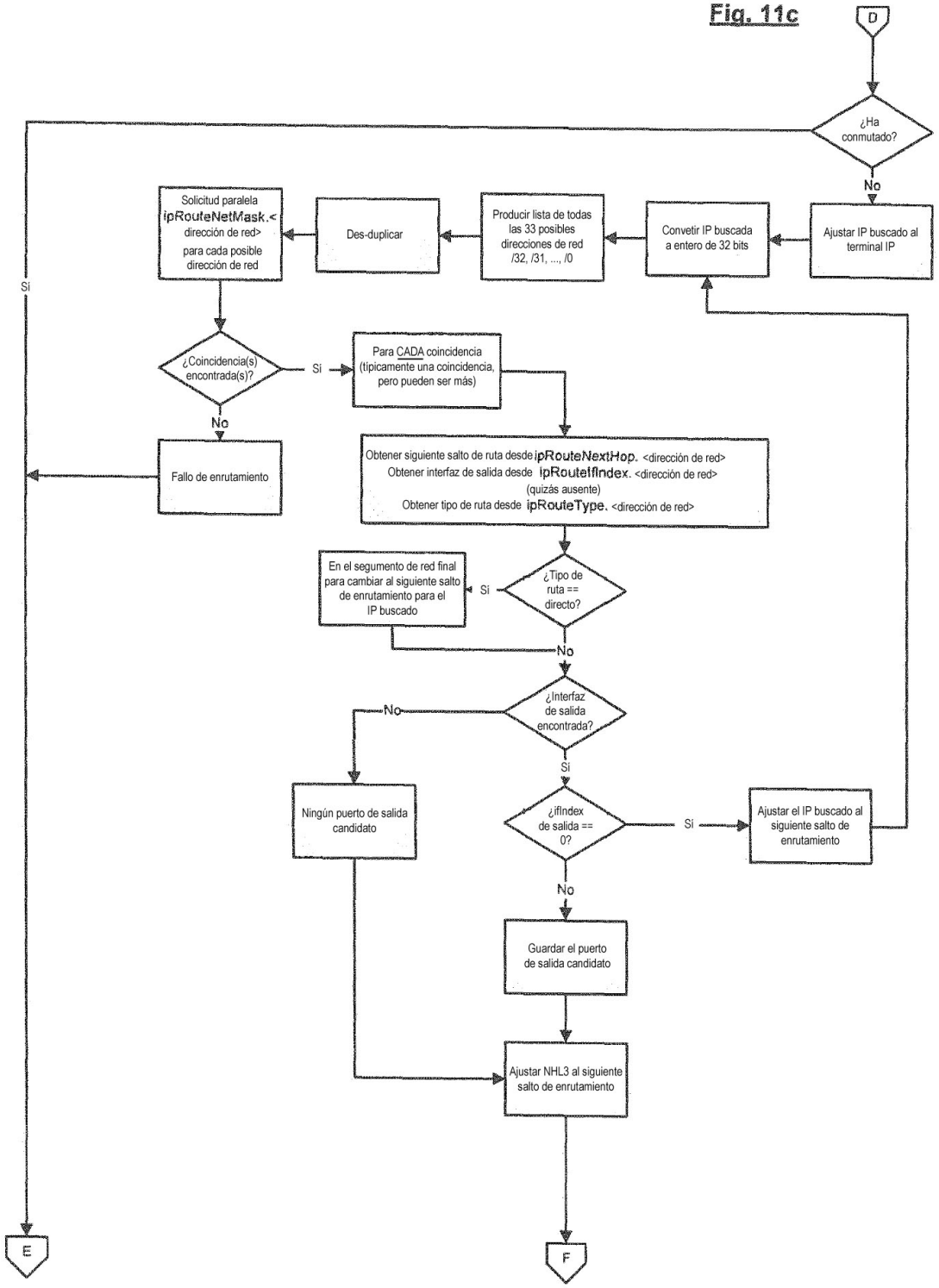

A
B
C
D
Sí
Para cada VLAN, consultar dot1dTpFdbPort <MAC> usando la VLAN especificada (p.ej. @VLAN)
¿Coincidencia encontrada?
No
¿Quedan más VLANs?
No
Fallo de conmutación, intentar enrutamiento
Sí
Guardar VLAN nativa e índice de puerto de puerto
Guardar VLAN relevante e índice de puerto de puente
Es el dispositivo de enfoque Cisco (comprobar sysOld)
Sí
No
Caminar cieIfDot1dBaseMappingPort
¿Coincidencia encontrada para cieIfDot1dBaseMappingPort. ifIndex = bridge port index?
No
Usando la VLAN almacenada (o nativa), sondear dot1dBasePortIfIndex.bridgePortIndex para encontrar ifIndex
Sí
¿ifIndex encontrado?
No
Fallo de conmutación, intentar enrutamiento
Tiene candidato egress port ifIndex
Sí
Establecer haveSwitched = verdadero

Fig. 11b

Fig. 11c


D
¿Ha conmutado?
No
Si
Ajustar IP buscado al terminal IP
Convetir IP buscada a entero de 32 bits
Producir lista de todas las 33 posibles direcciones de red /32, /31, ..., /0
Des-duplicar
Solicitud paralela ipRouteNetMask.<dirección de red> para cada posible dirección de red
¿Coincidencia(s) encontrada(s)?
Si
Para CADA coincidencia (tipicamente una coincidencia, pero pueden ser más)
No
Fallo de enrutamiento
Obtener siguiente salto de ruta desde ipRouteNextHop.<dirección de red>
Obtener interfaz de salida desde ipRouteIfIndex.<dirección de red>
(quizás ausente)
Obtener tipo de ruta desde ipRouteType.<dirección de red>
¿Tipo de ruta == directo?
Si
En el segumento de red final para cambiar al siguiente salto de enrutamiento para el IP buscado
No
¿Interfaz de salida encontrada?
No
Si
Ningún puerto de salida candidato
¿ifIndex de salida == 0?
Si
Ajustar el IP buscado al siguiente salto de enrutamiento
No
Guardar el puerto de salida candidato
Ajustar NHL3 al siguiente salto de enrutamiento
E
F

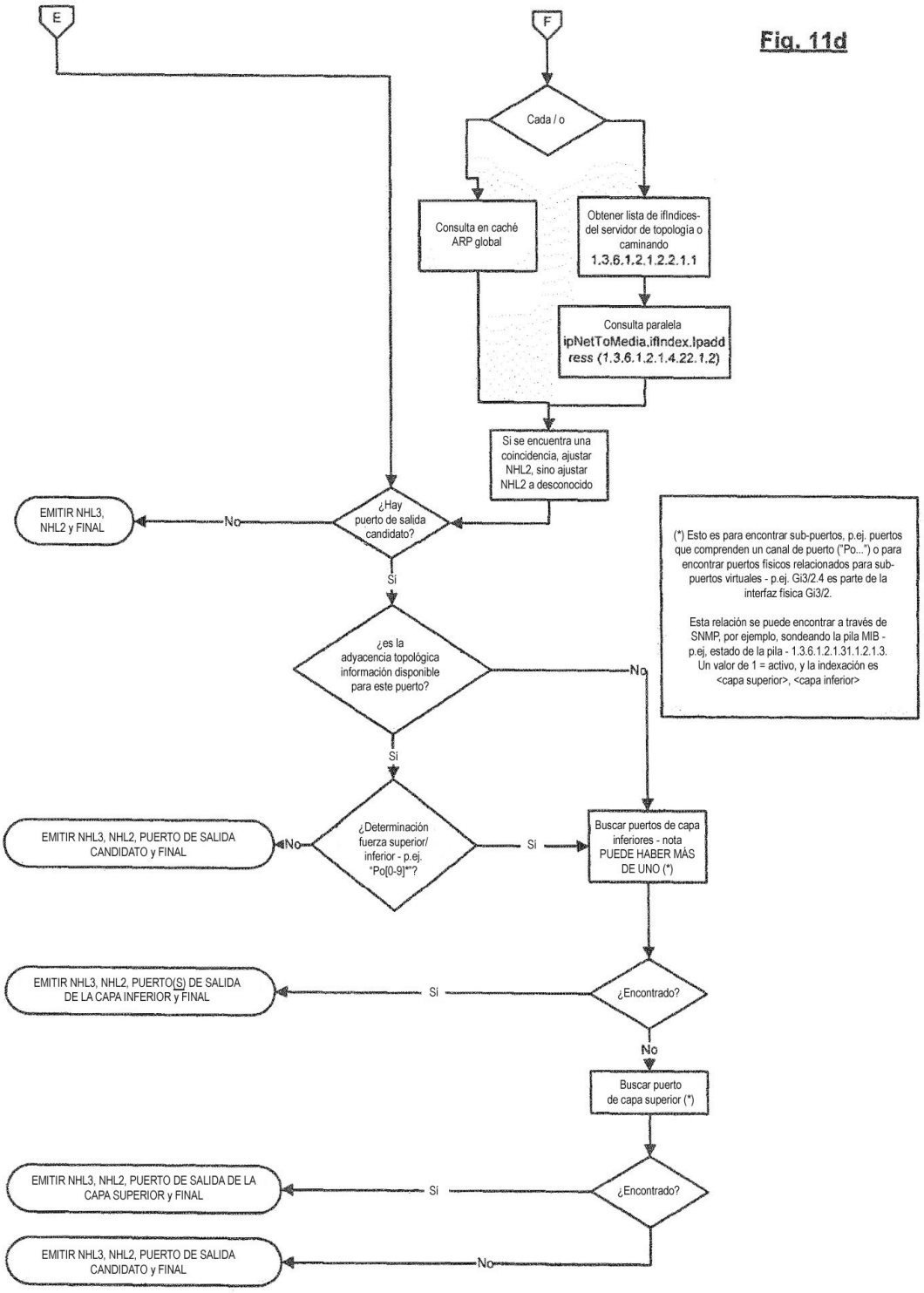
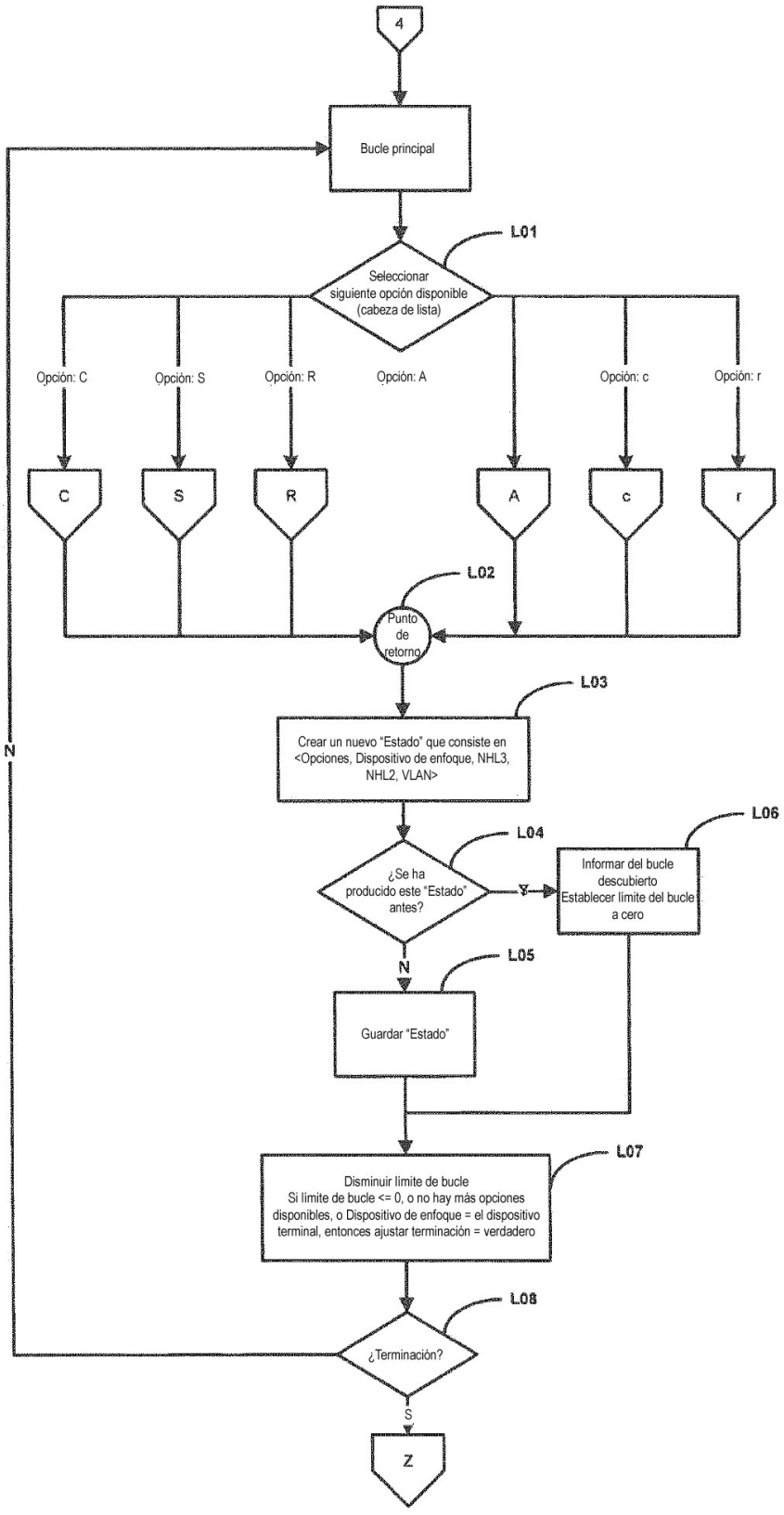
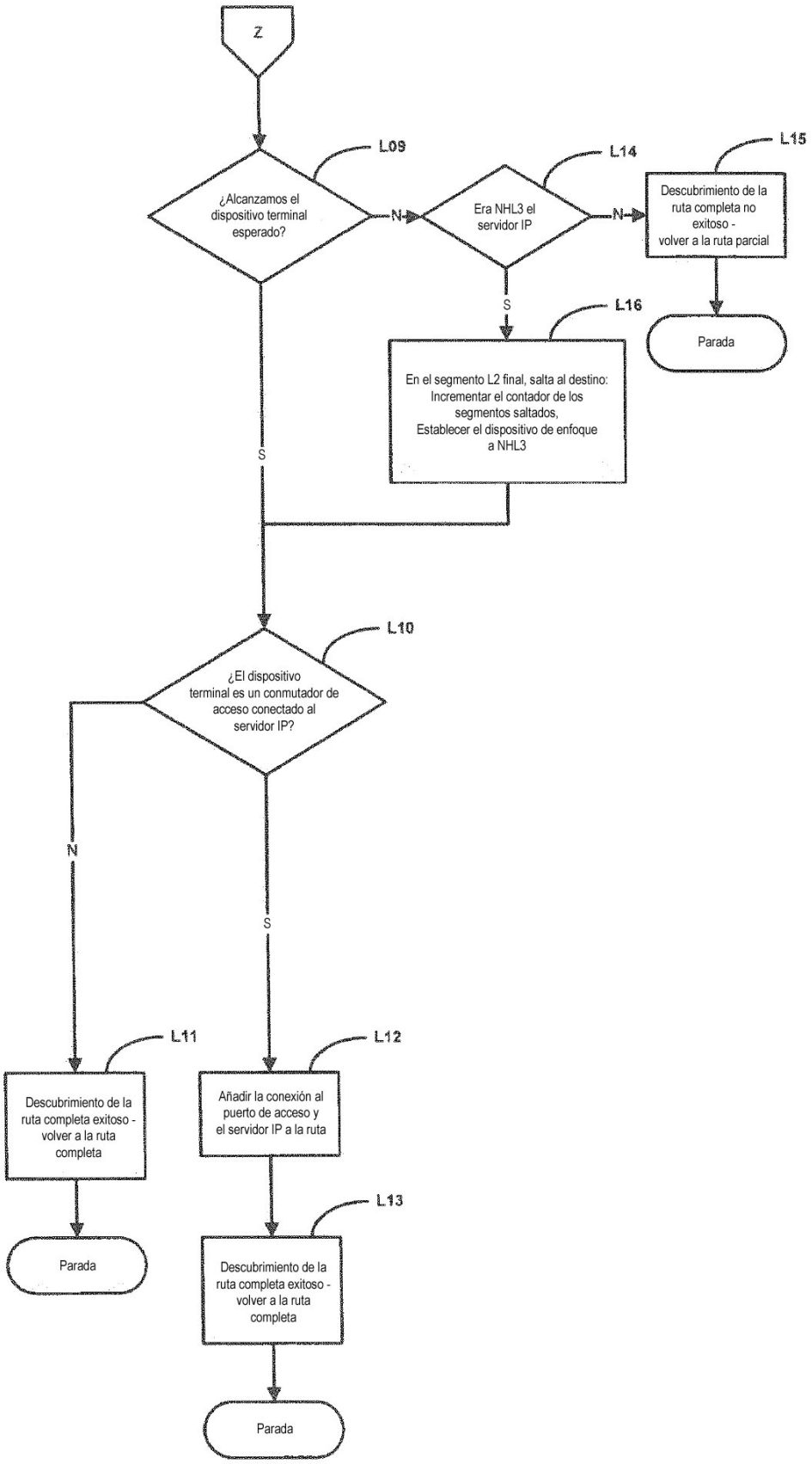
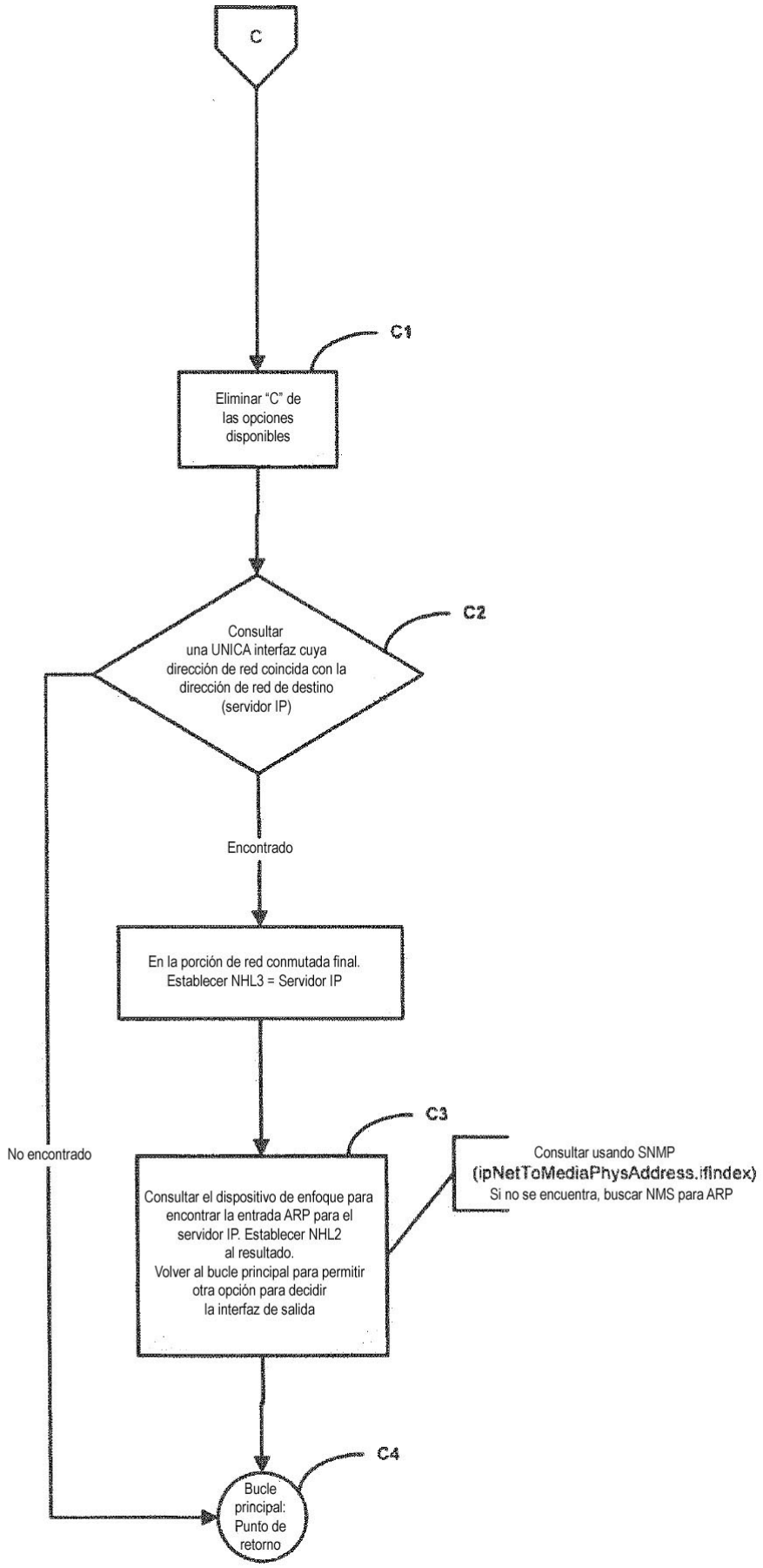
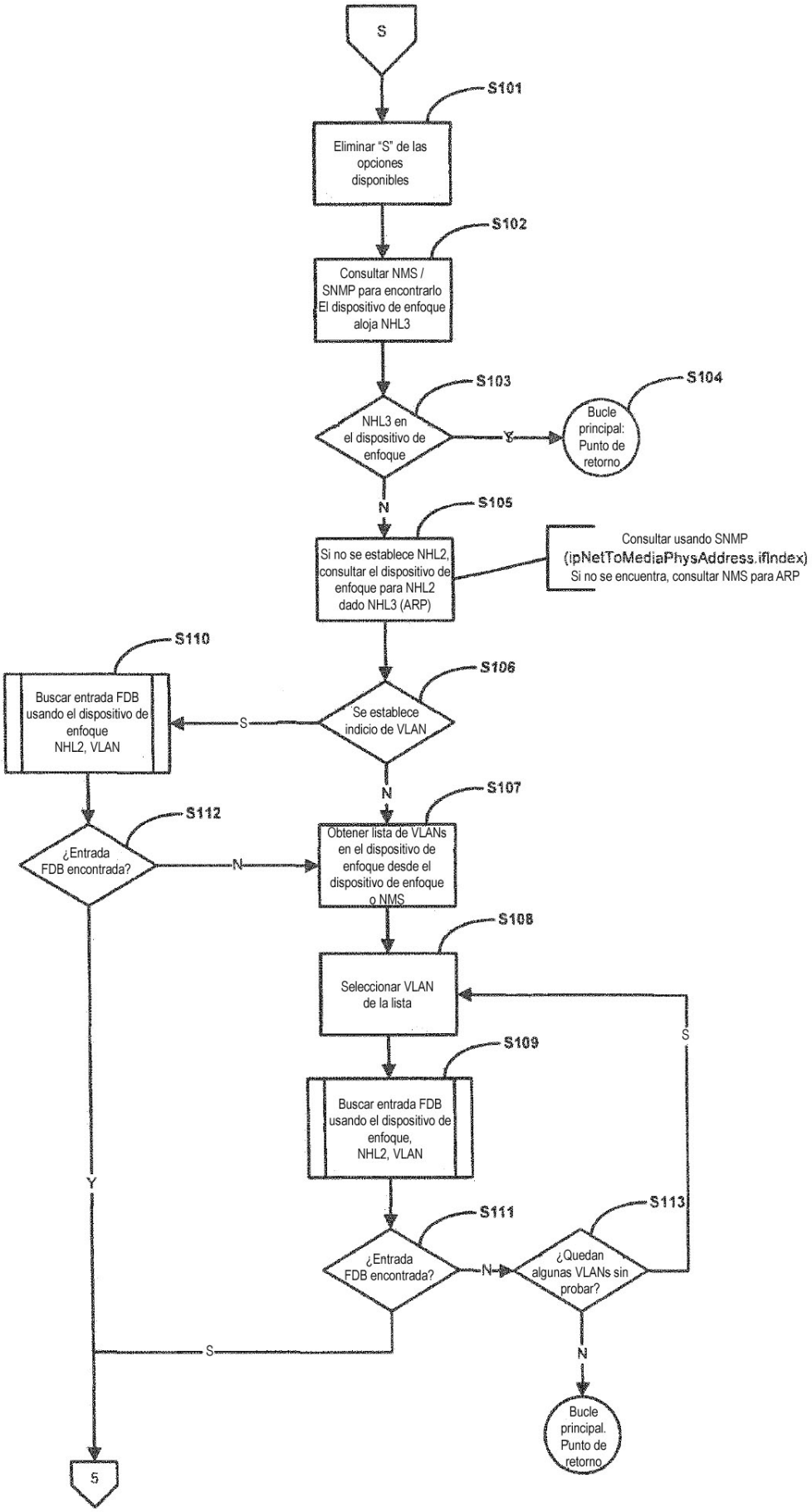

E
F
Cada / o
Consulta en caché ARP global
Obtener lista de ifIndices- del servidor de topología o caminando 1.3.6.1.2.1.2.2.1.1
Consulta paralela ipNetToMedia.ifIndex.Ipaddress (1.3.6.1.2.1.4.22.1.2)
Si se encuentra una coincidencia, ajustar NHL2, sino ajustar NHL2 a desconocido
¿Hay puerto de salida candidato?
No
EMITIR NHL3, NHL2 y FINAL
Sí
¿es la adyacencia topológica información disponible para este puerto?
No
Sí
¿Determinación fuerza superior/ inferior - p.ej. "Po[0-9]*"?
No
EMITIR NHL3, NHL2, PUERTO DE SALIDA CANDIDATO y FINAL
Sí
Buscar puertos de capa inferiores - nota PUEDE HABER MÁS DE UNO (*)
¿Encontrado?
Sí
EMITIR NHL3, NHL2, PUERTO(S) DE SALIDA DE LA CAPA INFERIOR y FINAL
No
Buscar puerto de capa superior (*)
¿Encontrado?
Sí
EMITIR NHL3, NHL2, PUERTO DE SALIDA DE LA CAPA SUPERIOR y FINAL
No
EMITIR NHL3, NHL2, PUERTO DE SALIDA CANDIDATO y FINAL
(*) Esto es para encontrar sub-puertos, p.ej. puertos que comprenden un canal de puerto ("Po...") o para encontrar puertos físicos relacionados para sub-puertos virtuales - p.ej. Gi3/2.4 es parte de la interfaz física Gi3/2.
Esta relación se puede encontrar a través de SNMP, por ejemplo, sondeando la pila MIB - p.ej, estado de la pila - 1.3.6.1.2.1.31.1.2.1.3. Un valor de 1 = activo, y la indexación es <capa superior>, <capa inferior>

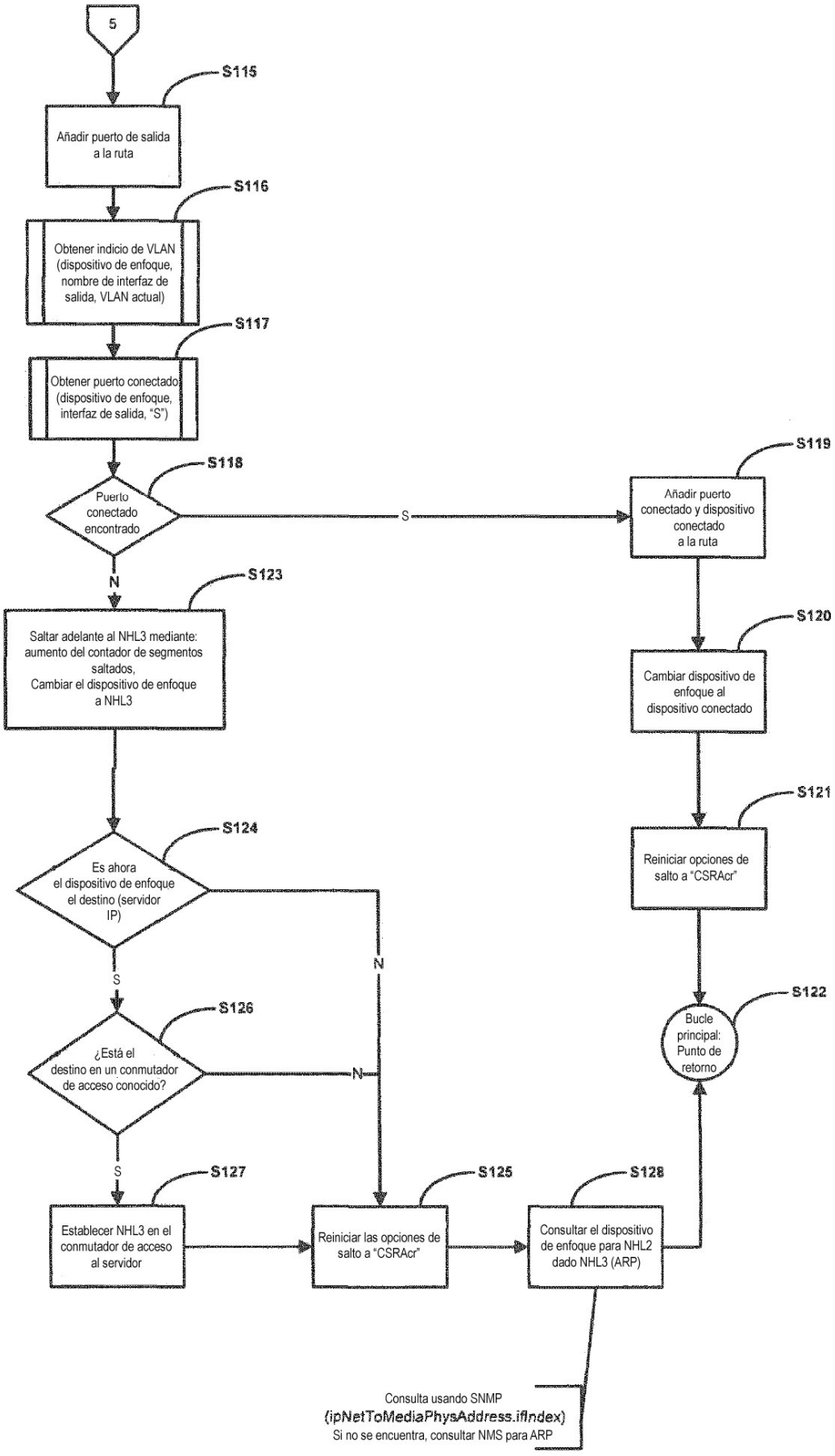
Fig. 11d

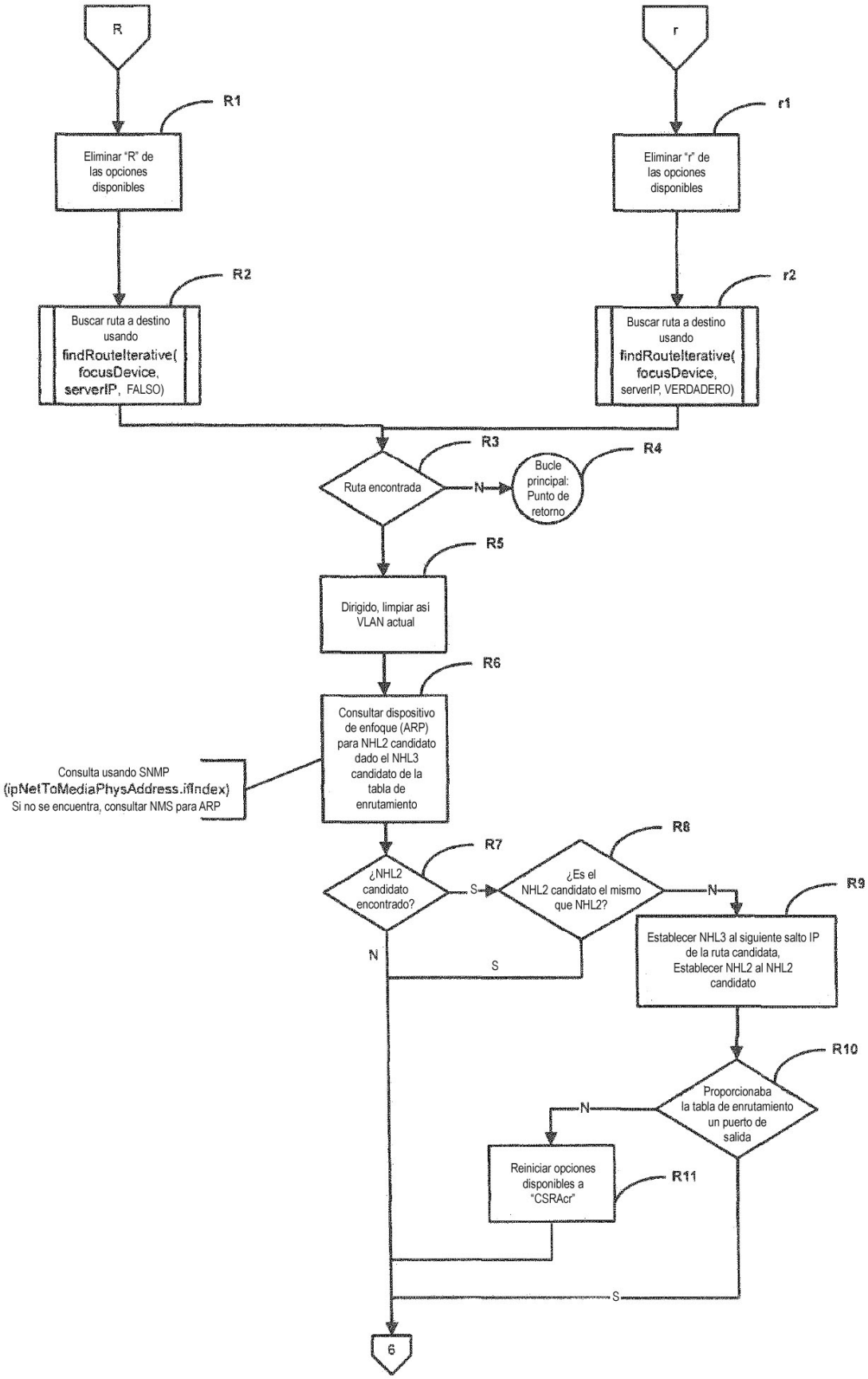

4
Bucle principal
L01
Seleccionar siguiente opción disponible (cabeza de lista)
Opción: C
Opción: S
Opción: R
Opción: A
Opción: c
Opción: r
C
S
R
A
c
r
L02
Punto de retorno
L03
Crear un nuevo "Estado" que consiste en <Opciones, Dispositivo de enfoque, NHL3, NHL2, VLAN>
L04
¿Se ha producido este "Estado" antes?
Y
L06
Informar del bucle descubierto Establecer límite del bucle a cero
N
L05
Guardar "Estado"
L07
Disminuir límite de bucle
Si límite de bucle <= 0, o no hay más opciones disponibles, o Dispositivo de enfoque = el dispositivo terminal, entonces ajustar terminación = verdadero
L08
¿Terminación?
N
S
Z

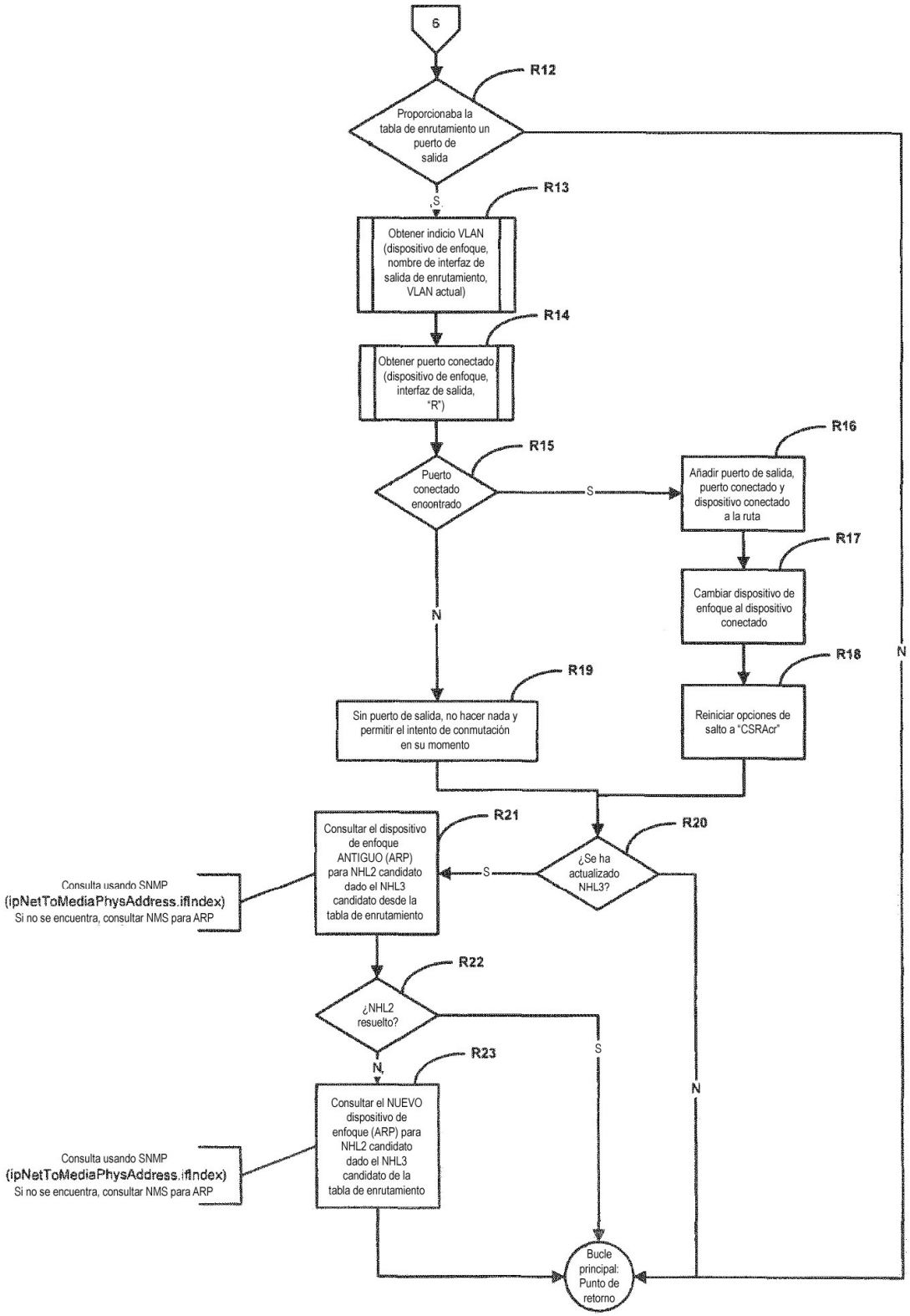

Z
L09
¿Alcanzamos el dispositivo terminal esperado?
N
L14
Era NHL3 el servidor IP
N
L15
Descubrimiento de la ruta completa no exitoso - volver a la ruta parcial
Parada
S
L16
En el segmento L2 final, salta al destino: Incrementar el contador de los segmentos saltados, Establecer el dispositivo de enfoque a NHL3
S
L10
¿El dispositivo terminal es un conmutador de acceso conectado al servidor IP?
N
S
L11
Descubrimiento de la ruta completa exitoso - volver a la ruta completa
Parada
L12
Añadir la conexión al puerto de acceso y el servidor IP a la ruta
L13
Descubrimiento de la ruta completa exitoso - volver a la ruta completa
Parada

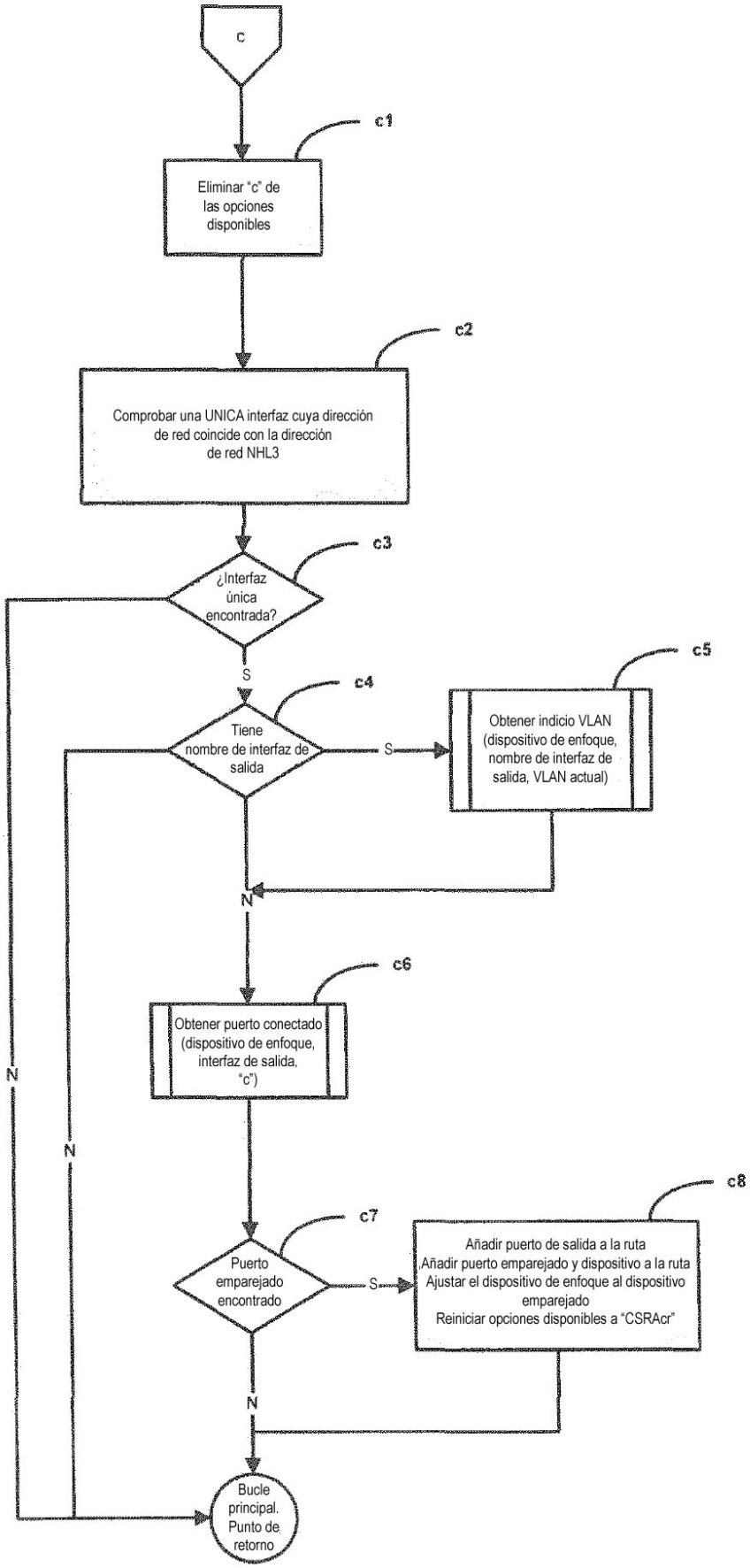

C
C1
Eliminar "C" de las opciones disponibles
C2
Consultar una UNICA interfaz cuya dirección de red coincida con la dirección de red de destino (servidor IP)
Encontrado
No encontrado
En la porción de red conmutada final. Establecer NHL3 = Servidor IP
C3
Consultar el dispositivo de enfoque para encontrar la entrada ARP para el servidor IP. Establecer NHL2 al resultado. Volver al bucle principal para permitir otra opción para decidir la interfaz de salida
Consultar usando SNMP (ipNetToMediaPhysAddress.ifIndex) Si no se encuentra, buscar NMS para ARP
C4
Bucle principal: Punto de retorno

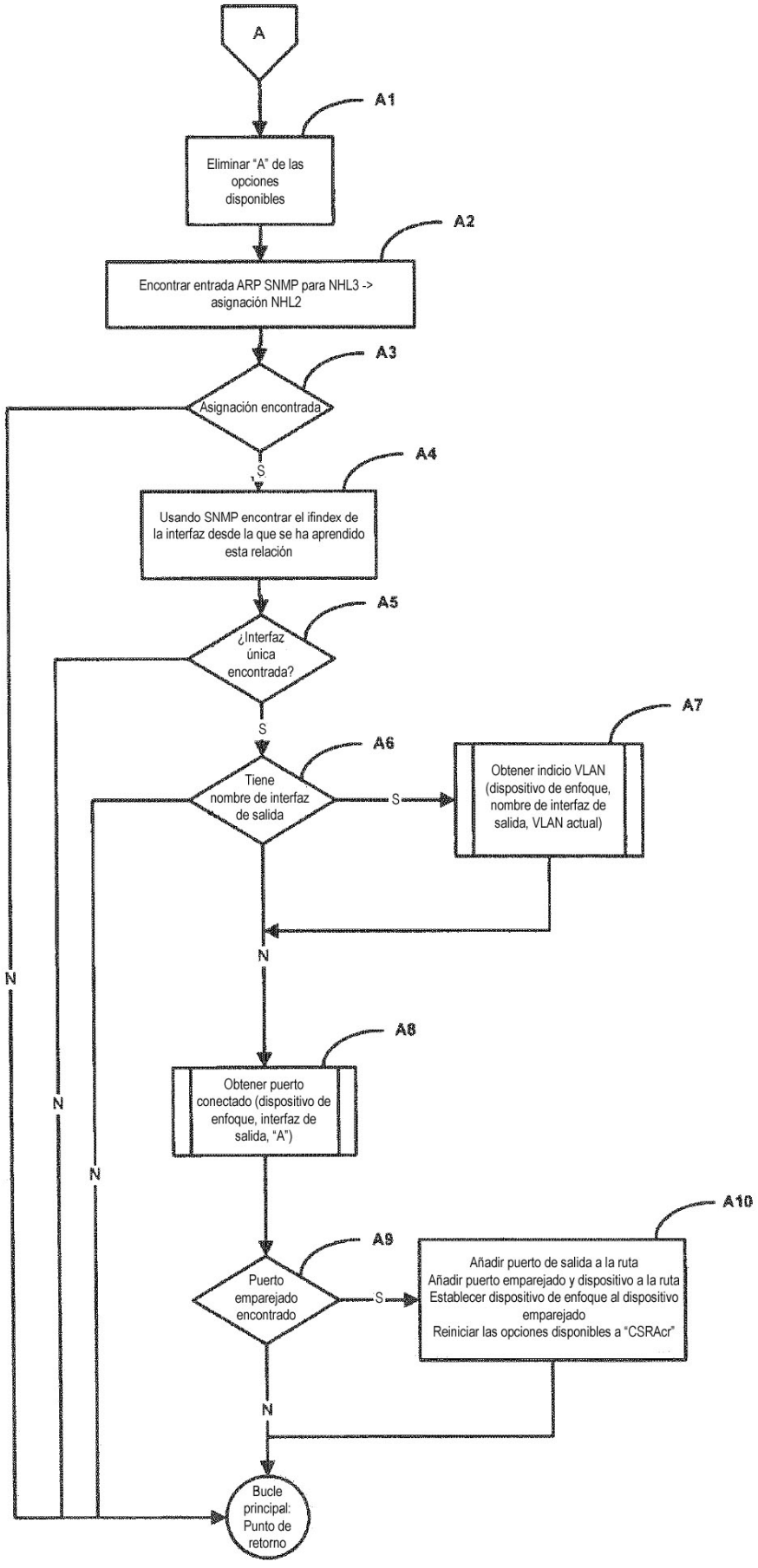

S
S101
Eliminar "S" de las opciones disponibles
S102
Consultar NMS / SNMP para encontrarlo El dispositivo de enfoque aloja NHL3
S103
NHL3 en el dispositivo de enfoque
Y
S104
Bucle principal: Punto de retorno
N
S105
Si no se establece NHL2, consultar el dispositivo de enfoque para NHL2 dado NHL3 (ARP)
Consultar usando SNMP (ipNetToMediaPhysAddress.ifIndex) Si no se encuentra, consultar NMS para ARP
S110
Buscar entrada FDB usando el dispositivo de enfoque NHL2, VLAN
S106
Se establece indicio de VLAN
S
N
S107
Obtener lista de VLANs en el dispositivo de enfoque desde el dispositivo de enfoque o NMS
S112
¿Entrada FDB encontrada?
N
S108
Seleccionar VLAN de la lista
S
S109
Buscar entrada FDB usando el dispositivo de enfoque, NHL2, VLAN
Y
S111
¿Entrada FDB encontrada?
N
S113
¿Quedan algunas VLANs sin probar?
S
N
Bucle principal. Punto de retorno
S

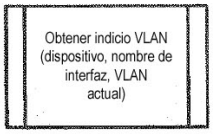

5
S115
Añadir puerto de salida a la ruta
S116
Obtener indicio de VLAN (dispositivo de enfoque, nombre de interfaz de salida, VLAN actual)
S117
Obtener puerto conectado (dispositivo de enfoque, interfaz de salida, "S")
S118
Puerto conectado encontrado
S
N
S119
Añadir puerto conectado y dispositivo conectado a la ruta
S120
Cambiar dispositivo de enfoque al dispositivo conectado
S121
Reiniciar opciones de salto a "CSRAcr"
S122
Bucle principal: Punto de retorno
S123
Saltar adelante al NHL3 mediante: aumento del contador de segmentos saltados, Cambiar el dispositivo de enfoque a NHL3
S124
Es ahora el dispositivo de enfoque el destino (servidor IP)
S
N
S126
¿Está el destino en un conmutador de acceso conocido?
S
N
S127
Establecer NHL3 en el conmutador de acceso al servidor
S125
Reiniciar las opciones de salto a "CSRAcr"
S128
Consultar el dispositivo de enfoque para NHL2 dado NHL3 (ARP)
Consulta usando SNMP
(ipNetToMediaPhysAddress.ifIndex)
Si no se encuentra, consultar NMS para ARP

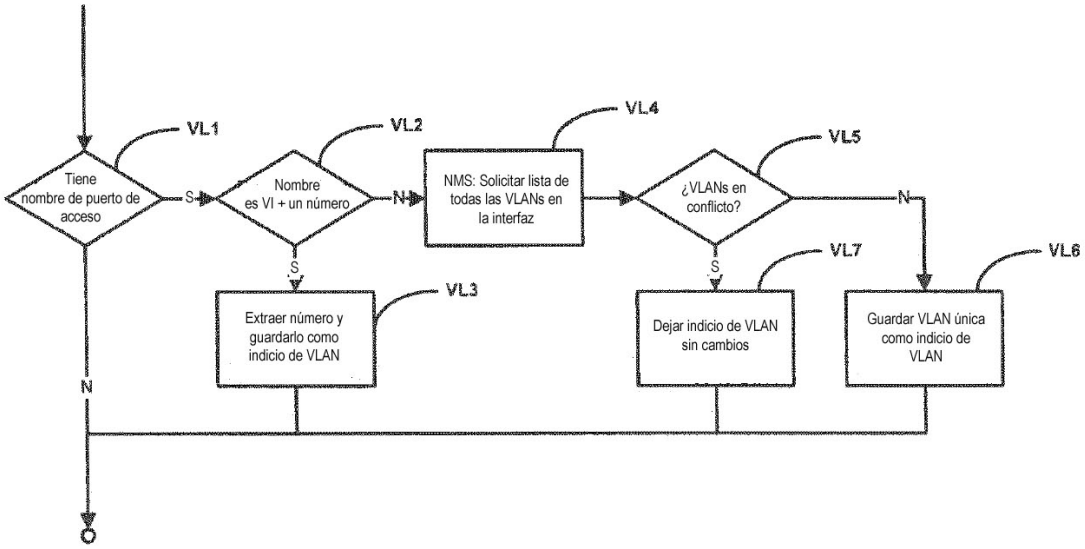

R
R1
Eliminar "R" de las opciones disponibles
R2
Buscar ruta a destino usando findRouteIterative(focusDevice, serverIP, FALSO)
r
r1
Eliminar "r" de las opciones disponibles
r2
Buscar ruta a destino usando findRouteIterative(focusDevice, serverIP, VERDADERO)
R3
Ruta encontrada
N
R4
Bucle principal: Punto de retorno
R5
Dirigido, limpiar así VLAN actual
R6
Consultar dispositivo de enfoque (ARP) para NHL2 candidato dado el NHL3 candidato de la tabla de enrutamiento
Consulta usando SNMP (ipNetToMediaPhysAddress.ifIndex)
Si no se encuentra, consultar NMS para ARP
R7
¿NHL2 candidato encontrado?
S
R8
¿Es el NHL2 candidato el mismo que NHL2?
N
R9
Establecer NHL3 al siguiente salto IP de la ruta candidata, Establecer NHL2 al NHL2 candidato
N
S
R10
Proporcionaba la tabla de enrutamiento un puerto de salida
N
Reiniciar opciones disponibles a "CSRAcr"
R11
S
6

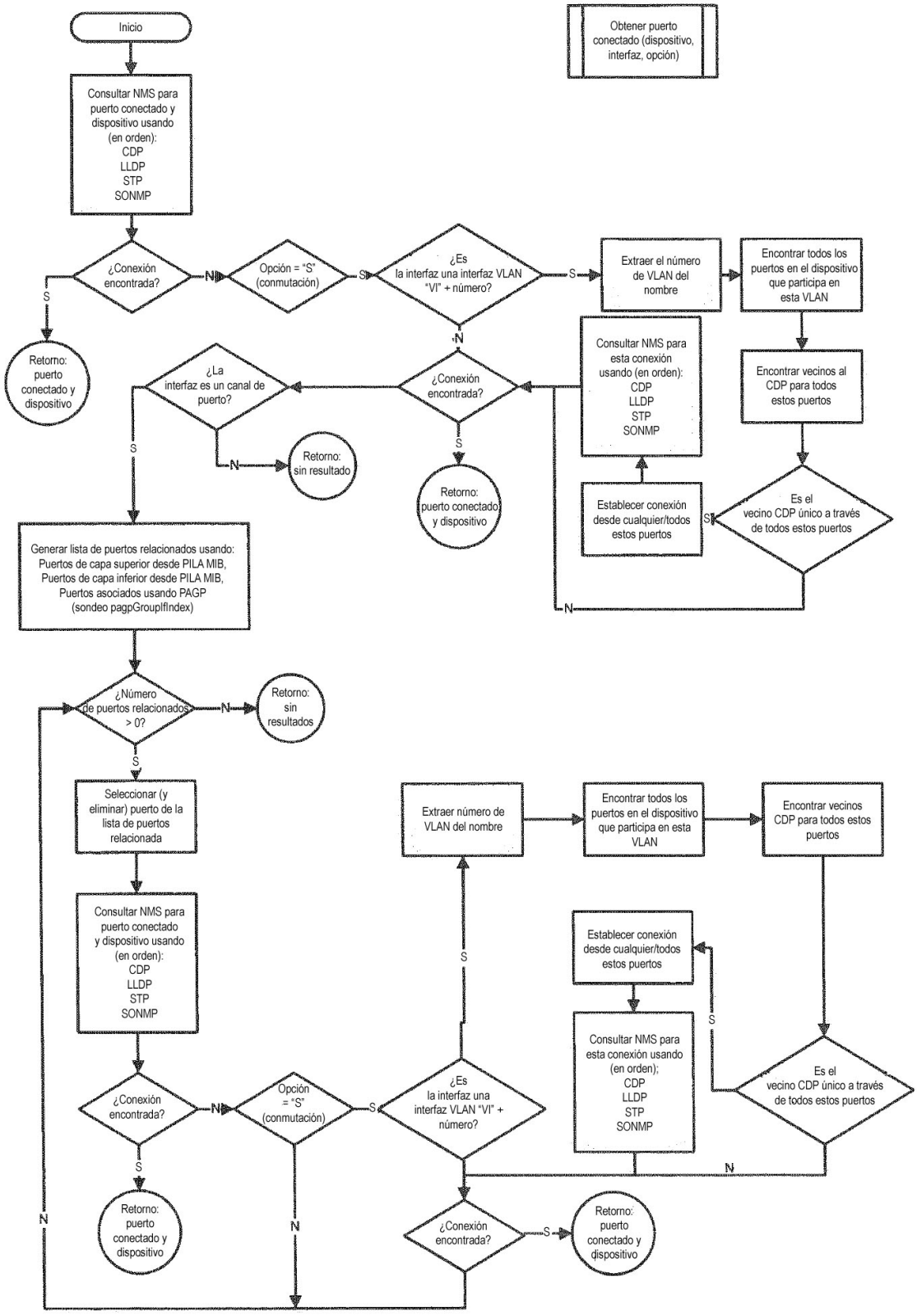

5
R12
Proporcionaba la tabla de enrutamiento un puerto de salida
S
N
R13
Obtener indicio VLAN (dispositivo de enfoque, nombre de interfaz de salida de enrutamiento, VLAN actual)
R14
Obtener puerto conectado (dispositivo de enfoque, interfaz de salida, "R")
R15
Puerto conectado encontrado
S
N
R16
Añadir puerto de salida, puerto conectado y dispositivo conectado a la ruta
R17
Cambiar dispositivo de enfoque al dispositivo conectado
R18
Reiniciar opciones de salto a "CSRAcr"
R19
Sin puerto de salida, no hacer nada y permitir el intento de conmutación en su momento
R20
¿Se ha actualizado NHL3?
S
N
R21
Consultar el dispositivo de enfoque ANTIGUO (ARP) para NHL2 candidato dado el NHL3 candidato desde la tabla de enrutamiento
Consulta usando SNMP (ipNetToMediaPhysAddress.ifIndex) Si no se encuentra, consultar NMS para ARP
R22
¿NHL2 resuelto?
S
N
R23
Consultar el NUEVO dispositivo de enfoque (ARP) para NHL2 candidato dado el NHL3 candidato de la tabla de enrutamiento
Consulta usando SNMP (ipNetToMediaPhysAddress.ifIndex) Si no se encuentra, consultar NMS para ARP
Bucle principal: Punto de retorno

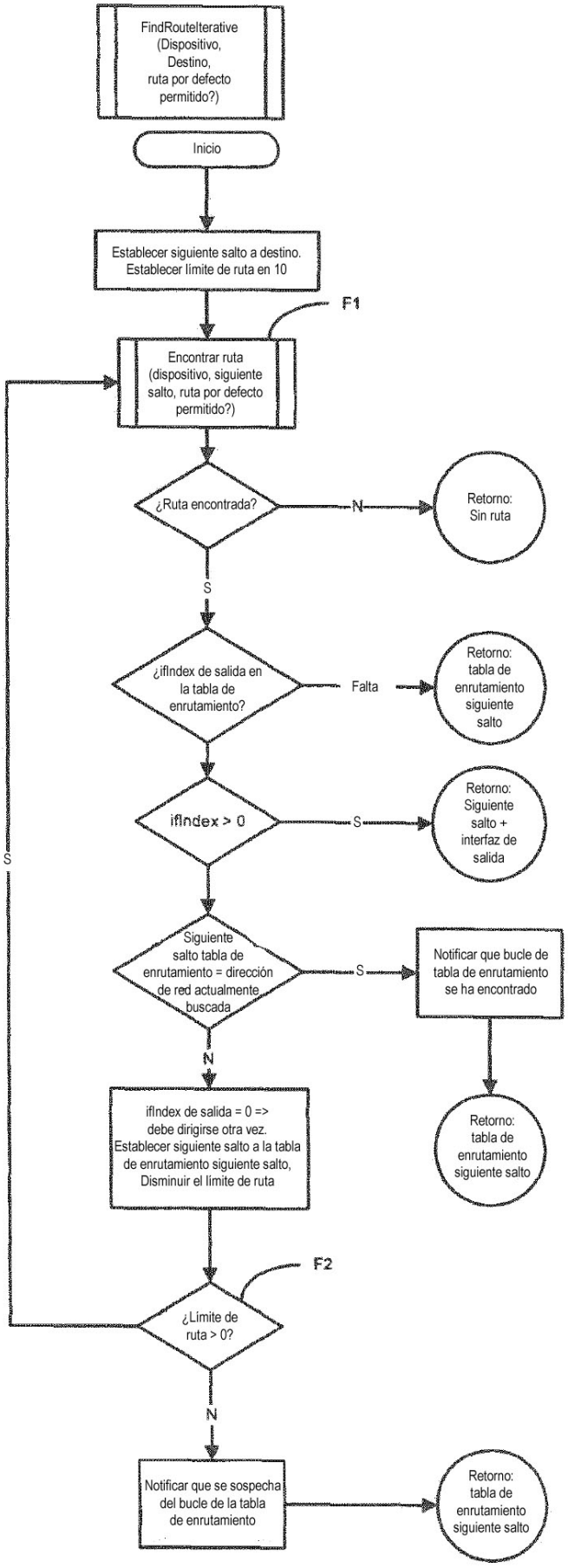

c
c1
Eliminar "c" de las opciones disponibles
c2
Comprobar una UNICA interfaz cuya dirección de red coincide con la dirección de red NHL3
c3
¿Interfaz única encontrada?
S
c4
Tiene nombre de interfaz de salida
S
c5
Obtener indicio VLAN (dispositivo de enfoque, nombre de interfaz de salida, VLAN actual)
N
c6
Obtener puerto conectado (dispositivo de enfoque, interfaz de salida, "c")
N
N
c7
Puerto emparejado encontrado
S
c8
Añadir puerto de salida a la ruta
Añadir puerto emparejado y dispositivo a la ruta
Ajustar el dispositivo de enfoque al dispositivo emparejado
Reiniciar opciones disponibles a "CSRAcr"
N
Bucle principal. Punto de retorno

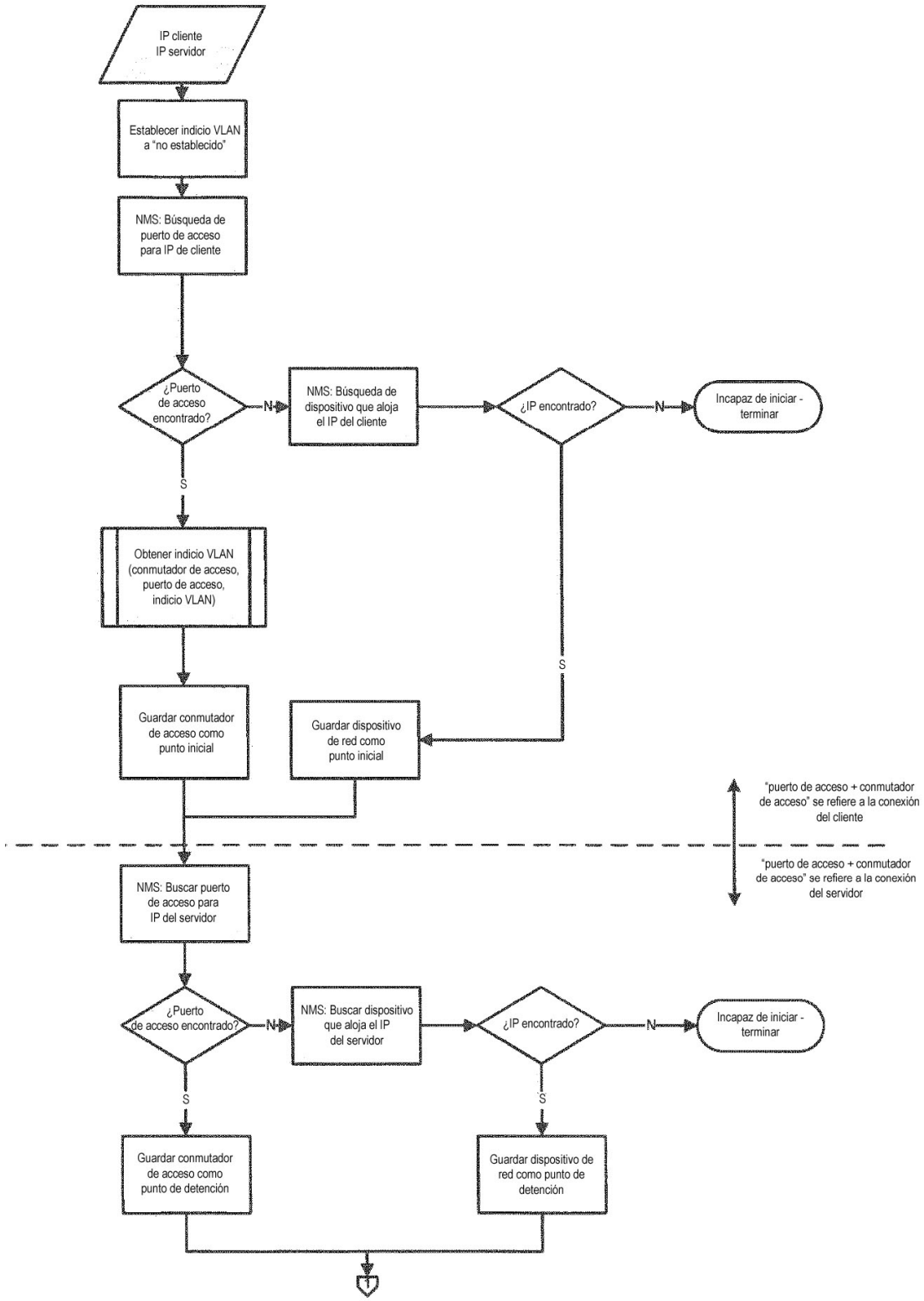

A
A1
Eliminar "A" de las opciones disponibles
A2
Encontrar entrada ARP SNMP para NHL3 -> asignación NHL2
A3
Asignación encontrada
S
A4
Usando SNMP encontrar el ifindex de la interfaz desde la que se ha aprendido esta relación
A5
¿Interfaz única encontrada?
S
A6
Tiene nombre de interfaz de salida
S
A7
Obtener indicio VLAN (dispositivo de enfoque, nombre de interfaz de salida, VLAN actual)
N
N
N
N
A8
Obtener puerto conectado (dispositivo de enfoque, interfaz de salida, "A")
A9
Puerto emparejado encontrado
S
A10
Añadir puerto de salida a la ruta
Añadir puerto emparejado y dispositivo a la ruta
Establecer dispositivo de enfoque al dispositivo emparejado
Reiniciar las opciones disponibles a "CSRAcr"
N
Bucle principal: Punto de retorno

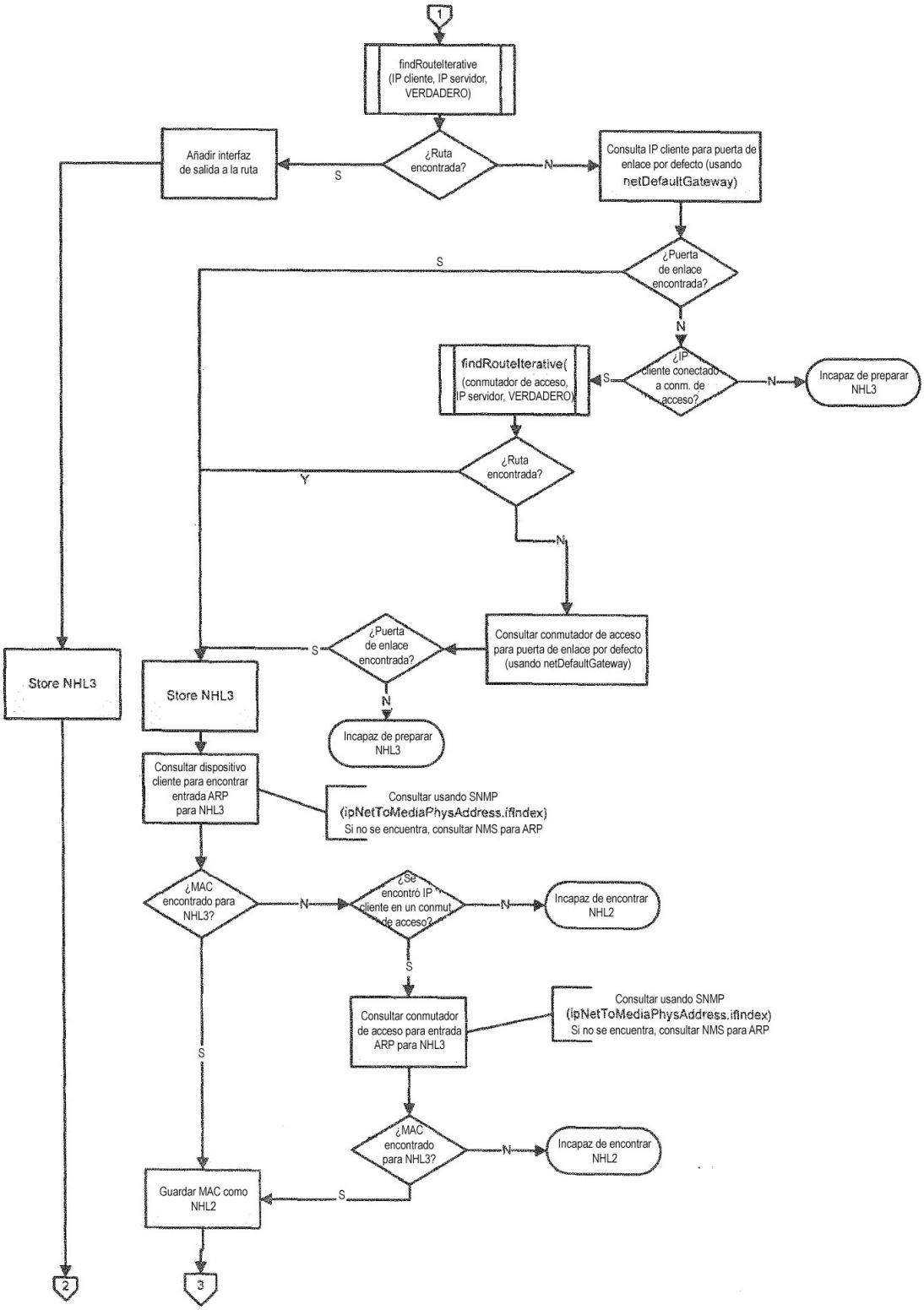

Obtener indicio VLAN (dispositivo, nombre de interfaz, VLAN actual)

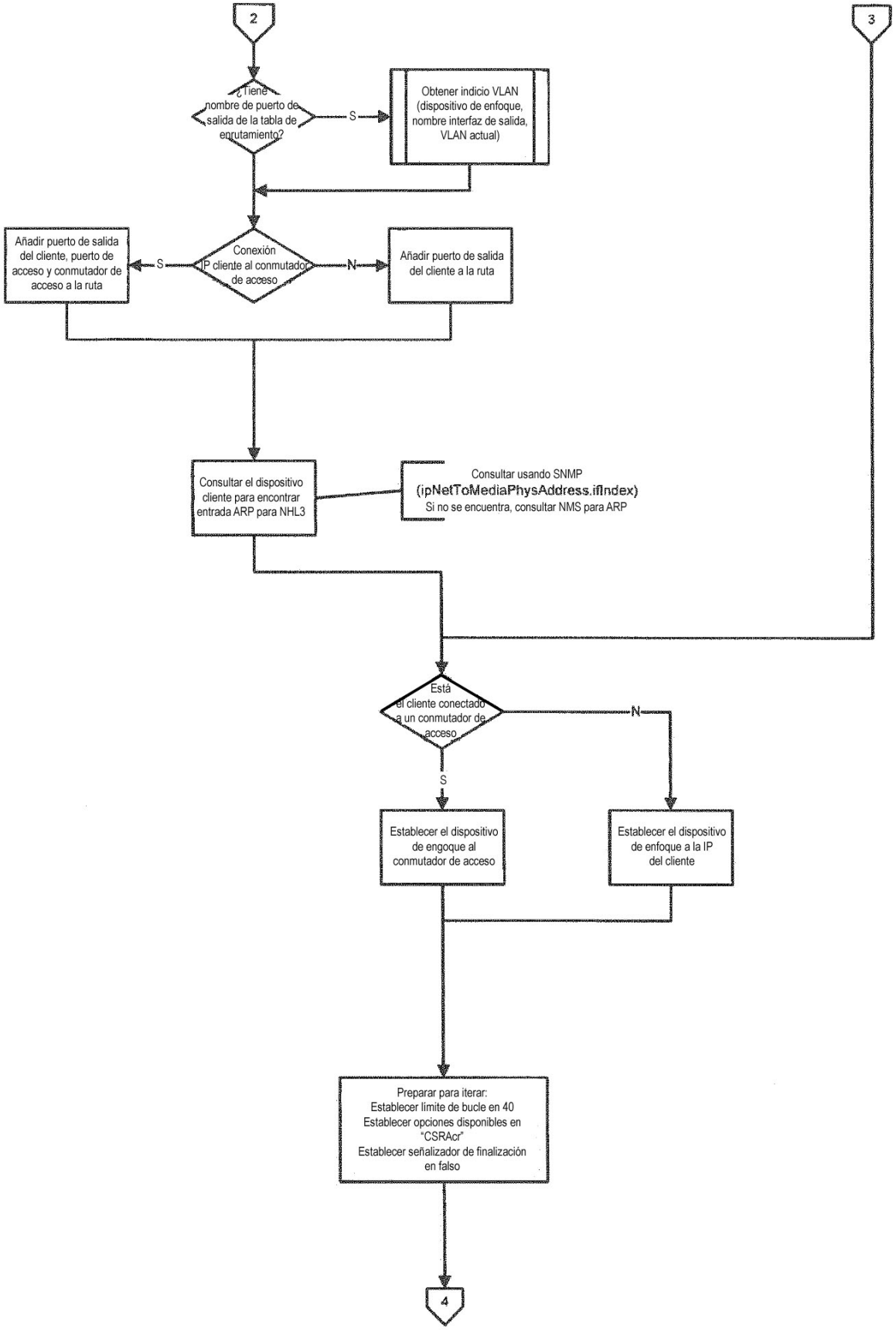

VL1
Tiene nombre de puerto de acceso
S
VL2
Nombre es VI + un número
N
VL4
NMS: Solicitar lista de todas las VLANs en la interfaz
VL5
¿VLANs en conflicto?
N
S
S
VL3
Extraer número y guardarlo como indicio de VLAN
VL7
Dejar indicio de VLAN sin cambios
VL6
Guardar VLAN única como indicio de VLAN
N

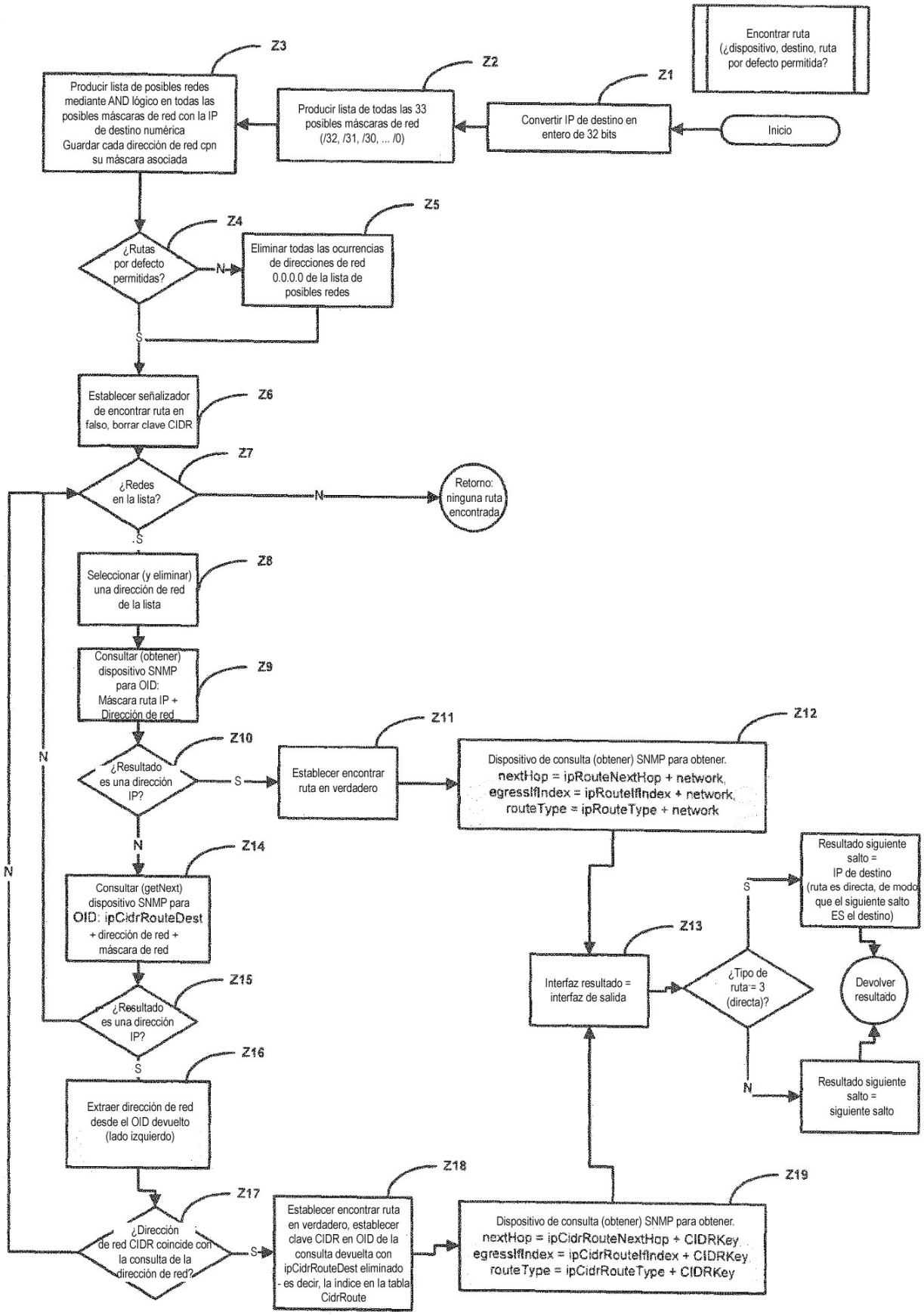

Inicio
Obtener puerto conectado (dispositivo, interfaz, opción)
Consultar NMS para puerto conectado y dispositivo usando (en orden): CDP LLDP STP SONMP
¿Conexión encontrada?
N
Opción = "S" (conmutación)
S
¿Es la interfaz una interfaz VLAN "VI" + número?
S
Extraer el número de VLAN del nombre
Encontrar todos los puertos en el dispositivo que participa en esta VLAN
Encontrar vecinos al CDP para todos estos puertos
Es el vecino CDP único a través de todos estos puertos
S
N
Establecer conexión desde cualquier/todos estos puertos
Consultar NMS para esta conexión usando (en orden): CDP LLDP STP SONMP
¿Conexión encontrada?
S
Retorno: puerto conectado y dispositivo
¿La interfaz es un canal de puerto?
N
Retorno: sin resultado
S
Retorno: puerto conectado y dispositivo
Generar lista de puertos relacionados usando: Puertos de capa superior desde PILA MIB, Puertos de capa inferior desde PILA MIB, Puertos asociados usando PAGP (sondeo pagpGroupIfIndex)
¿Número de puertos relacionados > 0?
N
Retorno: sin resultados
S
Seleccionar (y eliminar) puerto de la lista de puertos relacionada
Consultar NMS para puerto conectado y dispositivo usando (en orden): CDP LLDP STP SONMP
¿Conexión encontrada?
N
Opción = "S" (conmutación)
S
¿Es la interfaz una interfaz VLAN "VI" + número?
S
Extraer número de VLAN del nombre
Encontrar todos los puertos en el dispositivo que participa en esta VLAN
Encontrar vecinos CDP para todos estos puertos
Es el vecino CDP único a través de todos estos puertos
S
N
Establecer conexión desde cualquier/todos estos puertos
Consultar NMS para esta conexión usando (en orden); CDP LLDP STP SONMP
¿Conexión encontrada?
S
Retorno: puerto conectado y dispositivo
S
Retorno: puerto conectado y dispositivo
N
N

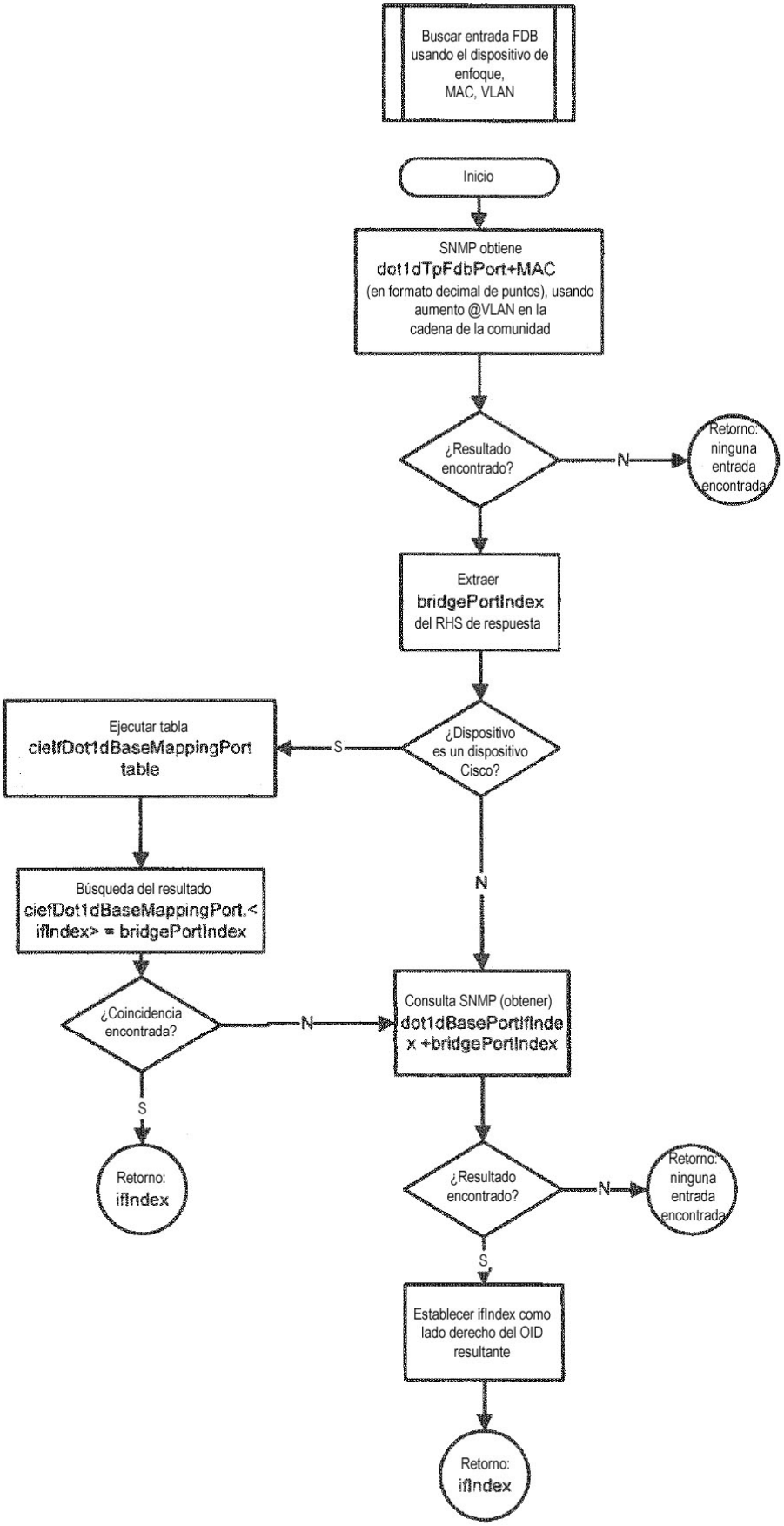

FindRouteIterative (Dispositivo, Destino, ruta por defecto permitido?)
Inicio
Establecer siguiente salto a destino. Establecer límite de ruta en 10
F1
Encontrar ruta (dispositivo, siguiente salto, ruta por defecto permitido?)
¿Ruta encontrada?
N
Retorno: Sin ruta
S
¿ifIndex de salida en la tabla de enrutamiento?
Falta
Retorno: tabla de enrutamiento siguiente salto
ifIndex > 0
S
Retorno: Siguiente salto + interfaz de salida
Siguiente salto tabla de enrutamiento = dirección de red actualmente buscada
S
Notificar que bucle de tabla de enrutamiento se ha encontrado
Retorno: tabla de enrutamiento siguiente salto
N
ifIndex de salida = 0 => debe dirigirse otra vez. Establecer siguiente salto a la tabla de enrutamiento siguiente salto, Disminuir el límite de ruta
F2
¿Límite de ruta > 0?
S
N
Notificar que se sospecha del bucle de la tabla de enrutamiento
Retorno: tabla de enrutamiento siguiente salto


IP cliente
IP servidor
Establecer indicio VLAN a "no establecido"
NMS: Búsqueda de puerto de acceso para IP de cliente
¿Puerto de acceso encontrado?
N
NMS: Búsqueda de dispositivo que aloja el IP del cliente
¿IP encontrado?
N
Incapaz de iniciar - terminar
S
Obtener indicio VLAN (conmutador de acceso, puerto de acceso, indicio VLAN)
S
Guardar conmutador de acceso como punto inicial
Guardar dispositivo de red como punto inicial
"puerto de acceso + conmutador de acceso" se refiere a la conexión del cliente
"puerto de acceso + conmutador de acceso" se refiere a la conexión del servidor
NMS: Buscar puerto de acceso para IP del servidor
¿Puerto de acceso encontrado?
N
NMS: Buscar dispositivo que aloja el IP del servidor
¿IP encontrado?
N
Incapaz de iniciar - terminar
S
S
Guardar conmutador de acceso como punto de detención
Guardar dispositivo de red como punto de detención
1


1
findRouteIterative (IP cliente, IP servidor, VERDADERO)
¿Ruta encontrada?
S
N
Añadir interfaz de salida a la ruta
Consulta IP cliente para puerta de enlace por defecto (usando netDefaultGateway)
¿Puerta de enlace encontrada?
S
N
findRouteIterative((conmutador de acceso, IP servidor, VERDADERO)
¿IP cliente conectado a conm. de acceso?
S
N
Incapaz de preparar NHL3
¿Ruta encontrada?
Y
N
Consultar conmutador de acceso para puerta de enlace por defecto (usando netDefaultGateway)
¿Puerta de enlace encontrada?
S
N
Incapaz de preparar NHL3
Store NHL3
Store NHL3
Consultar dispositivo cliente para encontrar entrada ARP para NHL3
Consultar usando SNMP (ipNetToMediaPhysAddress.ifIndex) Si no se encuentra, consultar NMS para ARP
¿MAC encontrado para NHL3?
N
¿Se encontró IP cliente en un conmut. de acceso?
N
Incapaz de encontrar NHL2
S
Consultar conmutador de acceso para entrada ARP para NHL3
Consultar usando SNMP (ipNetToMediaPhysAddress.ifIndex) Si no se encuentra, consultar NMS para ARP
S
¿MAC encontrado para NHL3?
N
Incapaz de encontrar NHL2
Guardar MAC como NHL2
S
2
3


2
3
¿Tiene nombre de puerto de salida de la tabla de enrutamiento?
S
Obtener indicio VLAN (dispositivo de enfoque, nombre interfaz de salida, VLAN actual)
Añadir puerto de salida del cliente, puerto de acceso y conmutador de acceso a la ruta
S
Conexión IP cliente al conmutador de acceso
N
Añadir puerto de salida del cliente a la ruta
Consultar el dispositivo cliente para encontrar entrada ARP para NHL3
Consultar usando SNMP
(ipNetToMediaPhysAddress.ifIndex)
Si no se encuentra, consultar NMS para ARP
Está el cliente conectado a un conmutador de acceso
N
S
Establecer el dispositivo de engoque al conmutador de acceso
Establecer el dispositivo de enfoque a la IP del cliente
Preparar para iterar:
Establecer límite de bucle en 40
Establecer opciones disponibles en "CSRAcr"
Establecer señalizador de finalización en falso
4


Encontrar ruta (¿dispositivo, destino, ruta por defecto permitida?
Inicio
Z1
Convertir IP de destino en entero de 32 bits
Z2
Producir lista de todas las 33 posibles máscaras de red (/32, /31, /30, ... /0)
Z3
Producir lista de posibles redes mediante AND lógico en todas las posibles máscaras de red con la IP de destino numérica Guardar cada dirección de red cpn su máscara asociada
Z4
¿Rutas por defecto permitidas?
N
Z5
Eliminar todas las ocurrencias de direcciones de red 0.0.0.0 de la lista de posibles redes
S
Establecer señalizador de encontrar ruta en falso, borrar clave CIDR
Z6
Z7
¿Redes en la lista?
N
Retorno: ninguna ruta encontrada
S
Z8
Seleccionar (y eliminar) una dirección de red de la lista
Z9
Consultar (obtener) dispositivo SNMP para OID: Máscara ruta IP + Dirección de red
Z10
¿Resultado es una dirección IP?
S
Z11
Establecer encontrar ruta en verdadero
Z12
Dispositivo de consulta (obtener) SNMP para obtener. nextHop = ipRouteNextHop + network, egressIfIndex = ipRouteIfIndex + network, routeType = ipRouteType + network
N
Z14
Consultar (getNext) dispositivo SNMP para OID: ipCidrRouteDest + dirección de red + máscara de red
Z15
¿Resultado es una dirección IP?
S
Z16
Extraer dirección de red desde el OID devuelto (lado izquierdo)
Z17
¿Dirección de red CIDR coincide con la consulta de la dirección de red?
S
Z18
Establecer encontrar ruta en verdadero, establecer clave CIDR en OID de la consulta devuelta con ipCidrRouteDest eliminado - es decir, la índice en la tabla CidrRoute
Z19
Dispositivo de consulta (obtener) SNMP para obtener. nextHop = ipCidrRouteNextHop + CIDRKey egressIfIndex = ipCidrRouteIfIndex + CIDRKey routeType = ipCidrRouteType + CIDRKey
Z13
Interfaz resultado = interfaz de salida
¿Tipo de ruta = 3 (directa)?
S
Resultado siguiente salto = IP de destino (ruta es directa, de modo que el siguiente salto ES el destino)
Devolver resultado
N
Resultado siguiente salto = siguiente salto


Buscar entrada FDB usando el dispositivo de enfoque, MAC, VLAN
Inicio
SNMP obtiene dot1dTpFdbPort+MAC (en formato decimal de puntos), usando aumento @VLAN en la cadena de la comunidad
¿Resultado encontrado?
N
Retorno: ninguna entrada encontrada
Extraer bridgePortIndex del RHS de respuesta
¿Dispositivo es un dispositivo Cisco?
S
Ejecutar tabla cieIfDot1dBaseMappingPort table
N
Búsqueda del resultado cieIfDot1dBaseMappingPort.<ifIndex> = bridgePortIndex
¿Coincidencia encontrada?
N
Consulta SNMP (obtener) dot1dBasePortIfIndex +bridgePortIndex
S
Retorno: ifIndex
¿Resultado encontrado?
N
Retorno: ninguna entrada encontrada
S
Establecer ifIndex como lado derecho del OID resultante
Retorno: ifIndex