(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 408 158**

(51) Int. CI.:

H04L 29/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **04.07.2003 E 03748200 (7)**

(97) Fecha y número de publicación de la concesión europea: **20.03.2013 EP 1574000**

(54) Título: **Procedimiento de reconocimiento y de análisis de protocolos en redes de datos**

(30) Prioridad:

29.07.2002 FR 0209599

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
18.06.2013

(73) Titular/es:

QOSMOS (100.0%)
IMMEUBLE LE CARDINET 5 IMPASSE CHALABRE
75017 PARIS, FR

(72) Inventor/es:

FDIDA, SERGE;
HARMEL, GAUTIER;
HORLAIT, ERIC;
PUJOLLE, GUY y
TOLLET, JÉRÔME

(74) Agente/Representante:

DE ELZABURU MÁRQUEZ, Alberto

DESCRIPCIÓN

Procedimiento de reconocimiento y de análisis de protocolos en redes de datos.

Una clasificación de tráficos que circulan en una red de datos permite decidir acerca de comportamientos que han de adoptarse para cada tráfico en función de su clasificación.

Por ejemplo, en un cortafuegos (firewall en inglés), una implantación de sistema de refuerzo de la seguridad generalmente estriba en un reconocimiento de propiedades de protocolo para impedir determinadas transferencias.

Por ejemplo aún, unos equipos de gestión de la calidad de servicio atribuyen prioridades a unos datos en función de reglas complejas que describen escenarios. Una correspondencia entre estos escenarios y paquetes de datos conducidos en el seno de conexiones utiliza técnicas de clasificación de esas conexiones.

Por ejemplo aún, unos equipos de supervisión de red (monitoring) realizan estadísticas para medir y controlar el estado de la red en un punto particular. Esto precisa de una clasificación y un reconocimiento de los diferentes flujos que transitan por ese punto.

Por ejemplo aún, para facturar servicios es útil una clasificación de diferentes flujos, ya que los costes varían según que esos servicios sean de tipo audio, vídeo, mensajería electrónica o consulta de base de datos. Por otra parte, muchas veces es imprescindible identificar correctamente a los usuarios de esos servicios para garantizar la facturación de los mismos.

Las operaciones de control y de gestión de las redes precisan así de una clasificación de conexiones entre diferentes emisores y receptores que generan flujos de datos digitales en esas redes. Conviene disponer entonces de procedimientos de clasificación eficientes y fiables.

Según el estado de la técnica conocido, se asigna una tarea de observación de paquetes de datos a un nodo de la red, tal como, por ejemplo, un servidor delegado (proxy server en inglés) por el que pasan conexiones que generan esos paquetes de datos.

La solicitud de patente WO 0101272 da a conocer un procedimiento y un aparato para supervisión de tráfico en una red. La aplicación de técnicas de reconocimiento de patrones (pattern matching en inglés) en campos predeterminados de paquetes de datos analizados permite identificar un protocolo que sucede a un protocolo identificado anteriormente en una pila de protocolos de conexión, con la condición de que el protocolo identificado anteriormente permita determinar los campos y los patrones o valores que han de reconocerse en el mismo para identificar el o los protocolos siguientes. El procedimiento y el aparato antedichos permiten la identificación de simplemente los protocolos explícitos. Más aún, el proceso de reconocimiento de patrones llevado a efecto se ejecuta sobre un árbol de protocolos de tipo jerárquico, sin posible vuelta atrás hacia los patrones de protocolo ya analizados.

Entre tales protocolos explícitos, se encuentra el protocolo Ethernet, para el cual la cabecera de los paquetes especifica si el protocolo siguiente en la pila de protocolos es, por ejemplo, el protocolo LLC o el protocolo IP eventualmente con su versión. Igualmente, la cabecera de los paquetes bajo protocolo IP especifica si el protocolo siguiente en la pila de protocolos es, por ejemplo, el protocolo TCP, UDP o ICMP.

Un problema que se plantea es el del reconocimiento de los protocolos implícitos. Un protocolo se llama implícito cuando no es identificable explícitamente de manera inequívoca mediante una cabecera de protocolo que lo precede en la pila de protocolos. Tal es el caso de numerosos protocolos de nivel de aplicación tales como Pointcast o Kazaa cuya utilización en la pila de protocolos de una conexión depende del contexto de la conexión establecida generalmente mediante negociaciones previas, difícilmente compilables con un sondeo en tiempo real al ritmo del flujo de los paquetes circulantes en el seno de la conexión.

Determinados protocolos conocidos tales como los protocolos HTTP, Telnet, FTP se hallan a día de hoy en el límite de los protocolos explícitos e implícitos. Estos protocolos se pueden considerar explícitos cuando un número de puerto reservado que figura en una cabecera de protocolo TCP da un indicador de destino que permite identificar de manera inequívoca el protocolo que es transportado, por ejemplo un número 80 correspondiente al protocolo HTTP, un número 23 correspondiente al protocolo Telnet, un número 21 correspondiente al protocolo FTP. Una estación cliente utiliza por ejemplo, bajo TCP, el número de puerto 80 para establecer una conexión de petición HTTP con una estación servidora, asignando un número de puerto dinámico a una conexión par que permite a la estación servidora responder a la estación cliente. Se destaca ya en este punto que el carácter explícito del protocolo HTTP en la conexión par para conducir las respuestas de la estación servidora a la estación cliente se ve mermado por la asignación dinámica de número de puerto, relacionado con el contexto de la conexión de petición. Además, a día de hoy no hay nada que impida que una estación cliente negocie previamente con la estación servidora un número de puerto distinto del número 80 para la conexión de petición HTTP. En tal caso, el protocolo HTTP es más implícito que explícito. Esto no deja de ser cierto para otros protocolos. Por otro lado, una conexión de petición bajo el protocolo FTP origina de manera conocida otras conexiones dinámicas para la transferencia efectiva de los archivos, utilizándose la conexión de petición y su conexión par para las transferencias de control. En el seno de la o las

conexiones dinámicas originadas, los números de puerto no permiten reconocer explícitamente el protocolo FTP. Una aplicación de filtros al campo del número de puerto bajo TCP no permite identificar de manera inequívoca el protocolo transportado.

Otro problema que se plantea es el del reconocimiento de protocolos cuya puesta en práctica varía tanto por la arquitectura de su utilización como por la incesante creación de nuevos protocolos.

Es conocida, por ejemplo, una arquitectura convencional de utilización del protocolo Telnet mediante apilamiento de la sucesión ordenada de protocolos Ethernet, IP, TCP, Telnet. Son posibles otras arquitecturas mediante apilamiento de la sucesión ordenada de protocolos Ethernet, IP, TCP, HTTP, Telnet o incluso Ethernet, IP, IP, TCP, HTTP, Telnet para gestionar la itinerancia.

Los sistemas del estado conocido de la técnica apenas se adaptan a modificaciones de arquitectura de protocolos mediante modificación de nexos de dependencia entre protocolos existentes o nuevos cuando estos sistemas se basan en reconocimientos de patrones en unos campos determinados por esos nexos de dependencia para identificar protocolos utilizados. Este inconveniente es especialmente notable en los sistemas de soporte físico para los cuales cualquier confrontación con conexiones establecidas según una arquitectura de protocolos no prevista precisa de una reconstrucción so pena de ineficacia.

La invención tiene por objeto un procedimiento para clasificar por medio de un sistema informático tal y como se define en la reivindicación 1. Formas particulares de realización de este procedimiento están definidas en las reivindicaciones 2 a 9.

Los mecanismos autoidentificadores asociados a los nombres de protocolos utilizables y las listas de protocolos utilizables asociadas cada una de ellas a un protocolo utilizable ofrecen una modularidad que permite adaptar el procedimiento a cualquier modificación de arquitectura. Cuando existe una nueva arquitectura que crea un nuevo nexo de dependencia posible entre protocolo padre y protocolo hijo, basta con añadir el nombre de protocolo hijo en la lista asociada al nombre de protocolo padre. Cuando existe un nuevo protocolo utilizable, basta con añadir en el sistema un nuevo mecanismo autoidentificador asociado al nombre del nuevo protocolo utilizable. Se puede llevar a cabo entonces una clasificación de conexión mediante construcción progresiva de una sucesión ordenada de protocolos utilizados en cada una de cuyas etapas el último protocolo constituye un protocolo padre.

La clasificación se termina cuando ya no es posible encontrar un protocolo hijo utilizado para un protocolo padre que es el último de los protocolos de la sucesión ordenada.

Ventajosamente, para buscar un nombre de protocolo hijo cuya naturaleza es implícita, el núcleo informático somete la información conducida a cada mecanismo autoidentificador asociado a un nombre de la lista de nombres de protocolos hijos hasta que uno de los mecanismos autoidentificadores declare reconocer información determinante o hasta que ningún mecanismo autoidentificador pueda declarar reconocer información determinante.

Disponiendo de las reglas de lenguaje del protocolo hijo con el que está asociado, el mecanismo autoidentificador puede comprobar fácilmente si la información sometida pertenece a ese lenguaje.

También ventajosamente, para buscar un nombre de protocolo hijo cuya naturaleza es explícita, el núcleo informático somete la información conducida al mecanismo autoidentificador asociado al último nombre de dicha sucesión ordenada, al objeto de que ese mecanismo autoidentificador encuentre el nombre del protocolo hijo entre la información determinante del protocolo padre.

Disponiendo de las reglas de lenguaje del protocolo padre con el que está asociado, el mecanismo autoidentificador puede encontrar fácilmente, entre la información sometida, aquella que especifica al protocolo hijo.

Cuando la sucesión ordenada se completa hasta el nivel de aplicación, la conexión con la que ésta va asociada se clasifica por tipo de nivel de aplicación.

Ello permite, por ejemplo, a un cortafuegos bloquear cualquier conexión de tipo transferencia de archivo o a un administrador de red medir un volumen de conexiones de tipo navegación por la Red (WWW por World Wide Web en inglés).

De manera particular, el núcleo informático elabora una firma actual para cada conexión detectada, sometiendo la totalidad o parte de la información conducida a al menos un mecanismo autoidentificador asociado a uno de los nombres de pequeño rango dentro de dicha sucesión ordenada, al objeto de que ese mecanismo autoidentificador encuentre, entre la información determinante, indicadores de origen y de destino, incorporados en dicha firma actual por el núcleo informático.

En la sucesión ordenada que constituye una imagen de pila de protocolos, los rangos pequeños corresponden generalmente a las capas de nivel físico, red y transporte. La firma permite distinguir cada conexión. Tal distinción entre conexiones de igual tipo de aplicación permite, por ejemplo a un cortafuegos, bloquear cualquier conexión de tipo transferencia de archivo desde o con destino a un equipo informático particular. Ello permite, todavía por

ejemplo, a un administrador de red medir un volumen de conexiones de tipo navegación por la Red iniciado por un equipo cliente o con destino a un equipo servidor particular.

De manera más particular, el núcleo informático cataloga cada primera estructura de datos en una primera tabla estableciendo una primera correspondencia asociativa entre cada primera estructura de datos y la firma actual elaborada para la conexión asociada.

El efecto técnico suplementario del que provee la firma es entonces el de permitir al núcleo informático remitirse a una misma conexión durante la construcción de la estructura de datos cuando la información conducida es dispar sin ser agrupada por un mecanismo subsidiario.

De hecho, conexiones distintas pueden pertenecer a un mismo flujo semántico. Es el caso, por ejemplo, de las conexiones pares y/o de las conexiones dinámicas.

De manera particularmente ventajosa, el núcleo informático establece en dicha primera tabla una segunda correspondencia asociativa entre cada firma actual y una firma par cuyos indicadores de origen son los indicadores de destino de la firma actual y cuyos indicadores de destino son los indicadores de origen de la firma actual.

También de manera particularmente ventajosa:

- el núcleo informático repasa los nombres de protocolos utilizados de la sucesión ordenada en la estructura de datos que construye para detectar cada nombre de protocolo con conexión dinámica,

- para cada nombre de protocolo con conexión dinámica detectado, el núcleo informático somete la información conducida al mecanismo autoidentificador asociado al nombre detectado con objeto de determinar si existe una conexión dinámica ulterior y, si existe una conexión dinámica ulterior, de asociarle una segunda estructura de datos establecida para contener una sucesión ordenada de potenciales nombres de protocolos que comienza con el nombre de protocolo llamado de base.

La segunda correspondencia asociativa y/o la segunda estructura de datos permiten dar indicación de las conexiones de un mismo flujo semántico incluso antes de detectarlas y seguidamente completar la clasificación de varias conexiones de un mismo flujo semántico a partir de la información conducida en una de las conexiones detectadas de ese mismo flujo semántico.

De manera más particularmente ventajosa, el núcleo informático cataloga cada segunda estructura de datos en una segunda tabla, estableciendo una correspondencia asociativa entre cada segunda estructura de datos y una firma potencial elaborada por el mecanismo autoidentificador asociado al nombre detectado.

De manera aún más particularmente ventajosa, el núcleo informático construye además la primera estructura de datos:

- buscando las sucesiones ordenadas de potenciales nombres de protocolos en las cuales viene incluida la sucesión ordenada de nombres de protocolos utilizados y

- cuando existe una sucesión ordenada de potenciales nombres de protocolos cuya firma potencial corresponde a la firma actual, completando la primera estructura de datos por medio de la segunda estructura de datos.

Con objeto de ser ejecutado en tiempo real para permitir una mejor reactividad de otros sistemas que utilizan las clasificaciones producidas y/o de reducir un volumen necesario de recursos de memoria del sistema informático que ejecuta el procedimiento, el procedimiento para clasificar conexiones se ve mejorado porque:

- el núcleo informático recolecta, dentro de paquetes de datos que pasan por el sistema informático en el seno de conexiones que han de detectarse, la información conducida útil para elaborar una firma al objeto de elaborar la firma actual cada vez que la información conducida útil es suficiente,

- el núcleo informático utiliza la firma actual así elaborada en tiempo real para detectar una conexión, al objeto de buscar, dentro de dicha primera tabla, la primera estructura de datos que corresponde a la firma actual, de asociar una nueva primera estructura de datos a la conexión detectada cuando no existe ninguna primera estructura de datos que corresponde a la firma actual y de empezar a construir, o seguir construyendo, la primera estructura de datos cuando existe una primera estructura de datos que corresponde a la firma actual, recolectando, dentro de los paquetes de datos, la información conducida útil para construir una primera estructura de datos.

En caso de fragmentación de paquetes de gran tamaño en paquetes de menor tamaño, puede suceder que la información útil recolectada dentro de un paquete de datos no es suficiente para elaborar una firma.

De acuerdo con una mejora suplementaria del procedimiento, el núcleo informático cataloga la información útil dentro de una segunda tabla, estableciendo una correspondencia asociativa entre la información útil que comprende entonces nexos de pertenencia a una misma conexión, hasta que la información útil sea suficiente para elaborar la firma actual.

La invención se comprenderá más fácilmente con la lectura de la descripción del ejemplo de una puesta en práctica preferida que sigue, con referencia a los dibujos que se acompañan, en los que:

la figura 1 muestra una posible representación en memoria de un grafo de protocolos, y

las figuras 2 y 3 son organigramas del procedimiento conforme a la invención.

Haciendo referencia a la figura 1, una tabla 101 contiene en una columna 102 un nombre de protocolo utilizable en cada fila señalada con 001 a 021, sin que el número de filas sea limitativo. En la fila correspondiente, cada nombre de protocolo utilizable de la columna 102, llamado entonces protocolo padre, lleva asociada una lista. En las columnas 105 a 110, sin que el número de columnas sea limitativo, cada lista queda establecida en el presente caso mediante punteros hacia filas de la tabla cuyo nombre que figura en la columna 102 es llamado entonces protocolo hijo.

Por ejemplo, la fila señalada con 001 contiene el nombre Base que identifica un protocolo llamado de base el cual, como excepción, no es realmente un protocolo utilizable por las conexiones sino, antes bien, un protocolo raíz utilizable para arrancar un recorrido de grafo de protocolos constituido a partir de una concatenación de grado en grado de todas las listas de protocolos hijos. La lista de nombres de protocolos hijos, asociada al nombre Base, contiene en el presente caso los nombres de protocolos Ethernet, ATM, PPP, que son protocolos de capa física, los primeros en ser efectivamente utilizables para conexiones. Así, el puntero en la columna 105 apunta a la fila 002, el puntero en la columna 106 apunta a la fila 003, el puntero en la columna 107 apunta a la fila 004.

En la fila señalada con 002, la lista de nombres de protocolos hijos, asociada al nombre Ethernet, contiene los nombres de protocolos LLC, Ipv6, Ipv4 a cada uno de los cuales apuntan respectivamente, en la fila 005, 007, 008, los punteros respectivamente situados en la columna 105, 106, 107.

En la fila señalada con 005, la lista de nombres de protocolos hijos, asociada al nombre LLC, contiene los nombres de protocolos STP, Ipv4 a cada uno de los cuales apuntan respectivamente, en la fila 006, 008, los punteros respectivamente situados en la columna 105, 106.

En la fila señalada con 006, la lista de nombres de protocolos hijos, asociada al nombre STP, está vacía. El nombre de protocolo STP se halla en una hoja del grafo de protocolos, es decir, no existe un protocolo hijo para este protocolo padre.

En la fila señalada con 007, la lista de nombres de protocolos hijos, asociada al nombre Ipv6, contiene los nombres de protocolos Ipv6, Ipv4, TCP, UDP e ICMP, a cada uno de los cuales apuntan respectivamente, en la fila 007, 008, 009, 010, 011, los punteros respectivamente situados en la columna 105, 106, 107, 108, 109. Se destaca que el nombre Ipv6 es a un tiempo nombre de protocolo padre y nombre de protocolo hijo. Ello materializa una posibilidad de poner una capa de red por encima de una capa de red de idéntico protocolo para gestionar la itinerancia de manera conocida creando un túnel de red. También se puede poner una capa de red de protocolo diferente por encima de una capa de red para gestionar diferencias de compatibilidad entre redes, por ejemplo una red compatible Ipv4 por la cual hacer pasar conexiones Ipv6.

La columna 103 contiene, para cada nombre de protocolo, un indicador que indica si el protocolo es explícito o implícito, es decir, si el protocolo en cuanto protocolo hijo está especificado o no por el protocolo padre. Como excepción, el indicador en la fila 103 indica que el protocolo designado Base es de tipo raíz, ya que no tiene protocolo padre.

La columna 104 contiene, para cada nombre de protocolo, otro indicador que indica si el protocolo es susceptible de generar conexiones dinámicas. Es el caso, por ejemplo, del protocolo FTP en la fila 012 o del protocolo HTTP en la fila 013.

Cada nombre de protocolo utilizable que figura en la columna 102 lleva asociado un mecanismo autoidentificador. Cada mecanismo autoidentificador se establece para ser activado por un núcleo informático por medio del nombre de protocolo al que está asociado. Cada mecanismo autoidentificador comprende unos filtros que, de manera preferida, se establecen en forma de reglas acordes con el lenguaje del protocolo con cuyo nombre está asociado el mecanismo autoidentificador. Por medio de un motor de inferencia, el mecanismo autoidentificador se establece para buscar dentro de un paquete de datos que a él somete el núcleo informático aquellos que cumplen unas reglas para responder a una solicitud del núcleo informático. De manera conocida, una regla comprende una parte premisa y una parte acción que genera una respuesta cuando se activa la regla y se valida la premisa. El motor de inferencia del mecanismo autoidentificador se establece para validar progresivamente una premisa de regla según el núcleo informático va sometiendo sucesivamente paquetes de datos, al objeto de que la parte acción pueda generar, con cada sometimiento, una respuesta de tipo negativa, probable o positiva, respectivamente según la invalidación o la validación parcial o total de la parte premisa.

Haciendo referencia a la figura 2, el procedimiento se activa en una etapa 1000 para cada paquete de datos que el sistema informático desvía de la red al nivel físico. En el nivel físico, todas las conexiones que han de detectarse pasan físicamente por un acoplador del sistema informático, por ejemplo de tipo Ethernet (normas IEEE 802), de tipo

ATM (Asynchronous Transfer Mode), de tipo PPP (Point to Point Protocol) o de cualquier otro tipo de nivel físico. En este nivel, el sistema informático dispone de una cadena de bits que constituye físicamente el paquete.

En una etapa 1001, el núcleo informático extrae una firma del paquete sometiendo el contenido del paquete a un mecanismo autoidentificador asociado al protocolo utilizado por el acoplador físico que recibe el paquete. El núcleo informático solicita al mecanismo autoidentificador que le devuelva un indicador de origen, un indicador de destino y un nombre de protocolo transportado. El mecanismo autoidentificador dispone de filtros establecidos para reconocer, dentro del paquete, los indicadores de origen y de destino, por ejemplo direcciones MAC en el caso de acoplador físico de tipo Ethernet, identificadores de circuito virtual (VCI por Virtual Circuit Identifier en inglés) y de camino virtual (VPI por Virtual Path Identifier en inglés) en el caso de acoplador físico de tipo ATM, números de teléfono llamante y llamado en el caso de acoplador físico de tipo PPP. Nótese que en el nivel físico, el indicador de destino es aquel que corresponde a la dirección física del acoplador del sistema informático. Al ser un protocolo de capa física generalmente de naturaleza explícita, el mecanismo autoidentificador también dispone de filtros establecidos para reconocer el protocolo transportado.

El núcleo informático, al recibir del mecanismo autoidentificador los indicadores de origen y de destino, genera una firma de nivel físico que contiene los identificadores de origen y de destino.

Por medio del nombre de protocolo transportado que es devuelto por el mecanismo autoidentificador de nivel físico, el núcleo informático somete los datos del paquete al mecanismo autoidentificador asociado y le solicita que le devuelva un indicador de origen, un indicador de destino y, de ser posible, un nombre de protocolo transportado. El mecanismo autoidentificador dispone de filtros establecidos para reconocer dentro del paquete los indicadores de origen y de destino, por ejemplo direcciones IP en el caso de protocolo IP. Al ser un protocolo de capa de red generalmente de naturaleza explícita, el mecanismo autoidentificador también dispone de filtros establecidos para reconocer el protocolo transportado.

El núcleo informático, al recibir del mecanismo autoidentificador los indicadores de origen y de destino, genera una firma de nivel de red que contiene los identificadores de origen y de destino.

Por medio del nombre de protocolo transportado que es devuelto por el mecanismo autoidentificador de nivel de red, el núcleo informático somete los datos del paquete al mecanismo autoidentificador asociado para generar, al igual que anteriormente, una firma de nivel correspondiente y repetir las operaciones anteriormente descritas hasta que un mecanismo autoidentificador señalice que no puede dar un nombre de protocolo transportado.

El núcleo informático concatena las firmas generadas para cada nivel con objeto de obtener una firma global que es específica de una conexión en cuyo seno es conducido el paquete.

Determinados protocolos ponen en práctica una fragmentación, es decir, dividen un paquete en varios paquetes de tamaño compatible con los protocolos de capa inferior. Es el caso, por ejemplo, cuando un protocolo de nivel más alto maneja paquetes de tamaño superior al tamaño máximo de los datos que pueden ser contenidos en una trama física (MTU por Maximum Transfer Unit en inglés). Recordemos que en los fragmentos de un mismo paquete, la cabecera original no se halla reproducida íntegramente en todos los fragmentos. Así, cuando un paquete que se presenta no constituye de hecho más que un fragmento de paquete, no necesariamente contiene toda la información que permite generar la firma específica de una conexión. Por ejemplo, en el caso del protocolo UDP por encima de IP, un fragmento UDP puede presentarse sin que contenga la cabecera UDP, en particular los puertos de origen y destino que permiten un cálculo de firma.

Cada mecanismo autoidentificador asociado a un nombre de protocolo susceptible de poner en práctica una fragmentación dispone de al menos un filtro establecido para reconocer una indicación de tamaño de paquete completo superior a un tamaño de paquete actual y/o para reconocer un identificador de restitución de paquete completo. El identificador de restitución generalmente es un número de orden asignado por el origen a cada fragmento de paquete. El mecanismo autoidentificador señaliza entonces la fragmentación al núcleo informático y el identificador de restitución. El núcleo informático utiliza una tabla asociativa que hace corresponder, con cada firma calculable hasta el nivel de fragmentación, la firma global para el paquete completo. Así, cuando se presenta un paquete o fragmento de paquete que cumple las condiciones de la tabla asociativa, el núcleo informático asocia directamente con ese paquete o fragmento de paquete la firma global correspondiente.

El núcleo informático utiliza a continuación, en esta puesta en práctica del procedimiento, la firma extraída en la etapa 1001 para construir una primera estructura de datos para la conexión detectada mediante la presentación de paquete en la etapa 1000.

En una etapa 1002, el núcleo informático busca en una tabla asociativa 1 de las conexiones actuales si existe una correspondencia con la firma global extraída en la etapa 1001. Con objeto de facilitar la búsqueda, el núcleo informático aplica a la firma global una función resumen (hash) para calcular una clave de obtención de resumen que referencia de manera unívoca una fila de la tabla asociativa 1 de manera conocida por medio de una tabla hash.

Si el núcleo informático encuentra una correspondencia en la tabla asociativa 1, activa una etapa 1005, en caso contrario, activa una etapa 1003. Así, el núcleo informático verifica, en la etapa 1002, si el paquete cuya firma se extrae pertenece a una conexión existente ya catalogada en la tabla asociativa de las conexiones actuales.

En la etapa 1003, el núcleo informático crea una fila en la tabla asociativa 1 estableciendo una correspondencia de la conexión detectada con la firma global extraída en la etapa 1001. Para la conexión detectada, el núcleo informático inicializa, en la fila creada, la primera estructura de datos con una sucesión ordenada de nombres de protocolos utilizados que empieza por el protocolo designado Base. La primera estructura de datos también comprende una cadena de variables binarias asignadas cada una de ellas a una columna de tabla 101 a partir de la columna 105. Para inicializar esta cadena de variables binarias, el núcleo informático lee cada celda de la tabla 101 que se encuentra en la intersección de una columna a partir de la columna 105 y de la fila de la tabla 101 que contiene, en la columna 102, el último nombre de protocolo de la sucesión ordenada. Para una celda no vacía, el núcleo informático posiciona a 1 la variable binaria de rango correspondiente para indicar una probabilidad no nula de utilización de nombre de protocolo referenciado por esa celda. Para una celda vacía, el núcleo informático posiciona a 0 la variable binaria de rango correspondiente para indicar una probabilidad nula de utilización de nombre de protocolo referenciado por esa celda. Simultáneamente, el núcleo informático calcula una firma de conexión par al objeto de que cada indicador de origen de la firma de conexión par sea el indicador de destino de nivel correspondiente que figura en la firma global y al objeto de que cada indicador de destino de la firma de conexión par sea el indicador de origen de nivel correspondiente que figura en la firma global. El núcleo informático busca a continuación, en la tabla asociativa 1, si existe una correspondencia con la firma de conexión par.

El núcleo informático, si encuentra en la tabla asociativa 1 una correspondencia con la firma de conexión par, establece una referencia cruzada entre la fila que contiene una correspondencia con la firma global actual y la fila que contiene una correspondencia con la firma de conexión par. Si el núcleo informático no encuentra en la tabla asociativa 1 una correspondencia con la firma de conexión par, crea una fila en la tabla asociativa 1 estableciendo una correspondencia de conexión par con la firma de conexión par y luego el núcleo informático establece una referencia cruzada entre la fila que contiene la correspondencia con la firma global actual y la fila que contiene la correspondencia con la firma de conexión par.

A continuación de la etapa 1003, el núcleo informático activa una etapa 1004.

En la etapa 1005, el núcleo informático mira, en la fila de la tabla asociativa 1 que contiene una correspondencia de conexión detectada con la firma global, si la conexión está marcada con un estado clasificado.

Si la conexión detectada está marcada con un estado clasificado, el núcleo informático activa una etapa 1006, en caso contrario, el núcleo informático activa la etapa 1004.

En la etapa 1004, el núcleo informático sigue construyendo la primera estructura de datos ejecutando las etapas que ahora se describen con referencia a la figura 3.

En una etapa 2000, el núcleo informático apunta, por una parte, a la fila de la tabla asociativa 1 que ha sido encontrada en la etapa 1002 o creada en la etapa 1003 y, por otra, al paquete recibido en la etapa 1000. El núcleo informático lee, como nombre de protocolo padre, el nombre de protocolo utilizado que se encuentra al final de la sucesión ordenada de protocolos utilizados de la fila a la que se apunta.

El núcleo informático busca en la primera estructura de datos si la cadena de variables binarias contiene una variable binaria de valor no nulo. Si todas las variables binarias son nulas, el núcleo informático activa una etapa 2010. Si existe al menos una variable binaria no nula, el núcleo informático activa una etapa 2001.

En la etapa 2001, el núcleo informático mira, en la columna 103 de la tabla 101, si el protocolo padre es de tipo implícito. Si el protocolo padre es de tipo implícito, el núcleo informático activa una etapa 2004. En caso contrario, es decir, si el protocolo padre es de tipo explícito, el núcleo informático activa una etapa 2002.

En la etapa 2002, el núcleo informático transmite el contenido del paquete al mecanismo autoidentificador asociado al nombre de protocolo padre y solicita al mecanismo autoidentificador que le devuelva el nombre de protocolo hijo. El mecanismo autoidentificador del protocolo padre aplica a los datos contenidos en el paquete sometido las reglas de reconocimiento del protocolo hijo y devuelve al núcleo informático una respuesta que contiene el nombre de protocolo hijo, si los datos son suficientes para validar la integridad de las premisas, o devuelve una respuesta de espera de datos complementarios si los datos son insuficientes para validar la integridad de las premisas.

Con la recepción de la respuesta del mecanismo autoidentificador, el núcleo informático termina la etapa 1004 si la respuesta es de tipo espera de datos complementarios. Si la respuesta contiene el nombre de protocolo hijo, el núcleo informático activa una etapa 2003.

En la etapa 2003, el núcleo informático añade el nombre de protocolo hijo a la sucesión ordenada de nombres de protocolos utilizados y reactiva la etapa 2000 para la cual el último nombre de protocolo hijo añadido en la etapa 2003 constituye entonces un nombre de protocolo padre. Según se ha descrito anteriormente, la primera estructura de datos comprende una cadena de variables binarias asignadas cada una de ellas a una columna de tabla 101 a

partir de la columna 105. Antes de reactivar la etapa 2000, el núcleo informático lee cada celda de la tabla 101 que se encuentra en la intersección de una columna a partir de la columna 105 y de la fila de la tabla 101 que contiene, en la columna 102, el último nombre de protocolo de la sucesión ordenada. Para una celda no vacía, el núcleo informático posiciona a 1 la variable binaria de rango correspondiente para indicar una probabilidad no nula de utilización de nombre de protocolo referenciado por esa celda. Para una celda vacía, el núcleo informático posiciona a 0 la variable binaria de rango correspondiente para indicar una probabilidad nula de utilización de nombre de protocolo referenciado por esa celda.

Según se ha descrito anteriormente, la etapa 2004 es activada si el protocolo padre es de tipo implícito, es decir, no identifica de manera inequívoca el protocolo transportado. En la etapa 2004, el núcleo informático busca en una segunda tabla asociativa 2 si existe una correspondencia entre la firma actual y una segunda sucesión ordenada de nombres de protocolos que contiene la primera sucesión ordenada de nombres de protocolos que es la catalogada en la primera tabla asociativa 1. Si el núcleo informático encuentra tal segunda sucesión ordenada en la tabla 2, activa una etapa 2005. En caso contrario, el núcleo informático activa una etapa 2006.

En la etapa 2005, el núcleo informático recupera, en la segunda sucesión ordenada, el o los nombres de protocolo que suceden al nombre de protocolo padre conservando el orden de los mismos. El núcleo informático elimina a continuación la segunda sucesión ordenada de la tabla asociativa 2 y luego activa la etapa 2003.

En la etapa 2003, el núcleo informático añade el nombre de protocolo hijo o los nombres de protocolo hijos a la primera sucesión ordenada de nombres de protocolos utilizados conservando su orden. Antes de reactivar la etapa 2000, el núcleo informático posiciona las variables de la cadena de variables binarias para el último nombre de protocolo añadido a la sucesión ordenada de nombres de protocolos utilizados.

En la etapa 2006, el núcleo informático lee sucesivamente una variable no nula de la cadena de variables binarias de la primera estructura de datos, empezando por la primera variable binaria no nula. El núcleo informático deduce, de la celda de la tabla 101 con un rango correspondiente a la variable binaria no nula, un nombre de protocolo hijo llamado probable. El núcleo informático somete el paquete de datos al mecanismo autoidentificador asociado al nombre de protocolo hijo probable, preguntando al mecanismo autoidentificador si el paquete que a él es sometido le permite identificar de manera inequívoca que el protocolo hijo es un protocolo utilizado. Con la recepción de la respuesta del mecanismo autoidentificador, el núcleo informático activa una etapa 2007.

En la etapa 2007, el núcleo informático activa la etapa 2003 si mecanismo autoidentificador responde que el protocolo hijo es utilizado, el núcleo informático activa una etapa 2008 si mecanismo autoidentificador responde que el protocolo hijo no es utilizado, el núcleo informático activa una etapa 2012 si mecanismo autoidentificador responde que el protocolo hijo posiblemente es utilizado.

En la etapa 2008, el núcleo informático posiciona a 0 la variable binaria actual para indicar que el nombre de protocolo hijo no es el de un protocolo utilizado. El núcleo informático activa a continuación la etapa 2012.

En la etapa 2012, el núcleo informático mira en la cadena de variables binarias si existe una variable binaria no nula después de la que ha llevado a la activación precedente de la etapa 2006. Si existe una variable binaria no nula, el núcleo informático activa la etapa 2006 para la nueva variable binaria no nula detectada en la etapa 2012. Si no existe una variable binaria no nula, el núcleo informático activa una etapa 2009.

En la etapa 2009, el núcleo informático mira si la cadena de variables binarias contiene tan sólo valores nulos. Si la cadena de variables binarias contiene tan sólo valores nulos, el núcleo informático activa la etapa 2010. En caso contrario, es decir, si la cadena de variables binarias contiene al menos un valor no nulo, el núcleo informático activa una etapa 2011.

En la etapa 2010, el núcleo informático marca un estado de conexión clasificado en la primera estructura de datos y activa seguidamente la etapa 2011.

En la etapa 2011, el núcleo informático actualiza la conexión par completando la primera estructura de datos asociada a la conexión par con la sucesión ordenada de nombres de protocolos utilizados y la cadena de variables binarias contenidas en la primera estructura de datos asociada a la conexión actual. El núcleo informático termina entonces la etapa 1004.

Después de la etapa 1004, el núcleo informático activa la etapa 1006 que ahora se describe con referencia a la figura 2.

En la etapa 1006, el núcleo informático somete sucesivamente a consideración un nombre de protocolo de la sucesión ordenada de nombres de protocolos utilizados empezando por el nombre de protocolo que sigue al nombre Base. El núcleo informático mira en la columna 104 si el nombre de protocolo considerado corresponde a un protocolo generador de conexión dinámica. Si el nombre de protocolo considerado no corresponde a un protocolo generador de conexión dinámica, el núcleo informático activa una etapa 1010. Si el nombre de protocolo considerado corresponde a un protocolo generador de conexión dinámica, el núcleo informático activa una etapa 1008.

En la etapa 1008, el núcleo informático somete el paquete de datos al mecanismo autoidentificador asociado al nombre de protocolo considerado para preguntar al mecanismo autoidentificador si detecta que en el paquete de datos sometido se genera una conexión dinámica y, si es así, devolver los nombres de protocolos previstos y los identificadores origen y destino de la conexión dinámica que se genera. Si el mecanismo autoidentificador devuelve nombres de protocolos previstos e identificadores origen y destino para una conexión dinámica, el núcleo informático activa una etapa 1009. En caso contrario, es decir, si el mecanismo autoidentificador responde que no detecta ninguna generación de conexión dinámica, el núcleo informático activa la etapa 1010.

En la etapa 1009, el núcleo informático añade a la segunda tabla asociativa 2 una segunda estructura de datos, asociada a una futura conexión utilizando la primera estructura de datos asociada a la conexión actual y utilizando la respuesta devuelta por el mecanismo autoidentificador en la etapa 1008.

En la etapa 1010, el núcleo informático mira si existe un nombre de protocolo siguiente en la sucesión ordenada de nombres de protocolos utilizados. Si existe un nombre de protocolo siguiente en la sucesión ordenada de nombres de protocolos utilizados, el núcleo informático reactiva la etapa 1006 para ese nombre de protocolo siguiente. En caso contrario, es decir, si no existe un nombre de protocolo siguiente en la sucesión ordenada de nombres de protocolos utilizados, el núcleo informático expulsa el paquete recibido en la etapa 1000 para que ese paquete prosiga su camino hasta su siguiente destino.

El procedimiento que se acaba de describir es reiterado para cada recepción de paquete en la etapa 1000.

REIVINDICACIONES

1. Procedimiento para clasificar, por medio de un sistema informático, conexiones detectadas entre emisores y receptores en redes de comunicación que utilizan una pila de protocolos designados cada uno de ellos por un nombre de protocolo utilizable, llamándose explícito un protocolo de la pila cuando es identificable de manera inequívoca mediante una cabecera de protocolo que lo precede en la pila de protocolos y llamándose implícito un protocolo en el caso contrario, comprendiendo el procedimiento, en el sistema informático:

- asociar, con cada nombre de protocolo, implícito o explícito, utilizable, un mecanismo autoidentificador de protocolo establecido para reconocer, de entre información conducida en una conexión detectada, información determinante del protocolo designado por ese nombre,

- asociar, con cada nombre de protocolo utilizable llamado protocolo padre, una lista vacía o no vacía de nombres de protocolos utilizables llamados protocolos hijos,

- con cada conexión detectada, asociar (1002, 1003), en un núcleo informático, una primera estructura de datos establecida para contener una sucesión ordenada de nombres de protocolos utilizados que se inicializa con un nombre de protocolo llamado de base,

- para cada conexión detectada, construir (1004), en el núcleo informático, la primera estructura de datos buscando (2000) en la lista de nombres de protocolos hijos asociada al último nombre de dicha sucesión ordenada, discriminando (2001), entre nombre de protocolo hijo de naturaleza implícita y nombre de protocolo hijo de naturaleza explícita, un nombre de protocolo hijo para el cual el mecanismo autoidentificador asociado reconoce (2002, 2007) información determinante de entre la información conducida y añadiendo (2003) seguidamente, al final de la sucesión ordenada, el nombre de protocolo hijo cuando éste es encontrado y recomenzando a buscar mientras sea posible encontrar (2009) en la lista de nombres de protocolos hijos, asociada al último nombre de dicha sucesión ordenada, un nombre de protocolo hijo para el cual el mecanismo autoidentificador asociado reconoce información determinante de entre la información conducida,

- declarar (2010) como clasificada la conexión detectada en el núcleo informático cuando ya no es posible encontrar en la lista de nombres de protocolos hijos, asociada al último nombre de dicha sucesión ordenada, un nombre de protocolo hijo para el cual el mecanismo autoidentificador asociado reconoce información determinante de entre la información conducida,

en el que, para buscar un nombre de protocolo hijo de naturaleza implícita, el núcleo informático somete (2006) la información conducida a cada mecanismo autoidentificador asociado a un nombre de la lista de nombres de protocolos hijos hasta que uno de los mecanismos autoidentificadores declare (2007) reconocer información determinante partiendo de unas reglas de lenguaje del protocolo hijo o hasta que ningún mecanismo autoidentificador pueda declarar reconocer (2012) información determinante,

y en el que, para buscar un nombre de protocolo hijo de naturaleza explícita, el núcleo informático somete (2002) la información conducida al mecanismo autoidentificador asociado al último nombre de dicha sucesión ordenada, al objeto de que ese mecanismo autoidentificador encuentre el nombre del protocolo hijo entre la información determinante del protocolo padre.

2. Procedimiento para clasificar conexiones según la reivindicación 1, caracterizado porque el núcleo informático elabora (1001) una firma actual para cada conexión detectada, comprendiendo la elaboración de la firma actual el sometimiento de la totalidad o parte de la información conducida a al menos un mecanismo autoidentificador asociado a uno de los nombres de pequeño rango dentro de dicha sucesión ordenada, al objeto de que ese mecanismo autoidentificador encuentre, entre la información determinante, indicadores de origen y de destino, siendo elaborada dicha firma actual por el núcleo informático al objeto de contener dichos indicadores de origen y de destino.

3. Procedimiento para clasificar conexiones según la reivindicación 1, caracterizado porque el núcleo informático cataloga cada primera estructura de datos en una primera tabla (1) estableciendo una primera correspondencia asociativa entre cada primera estructura de datos y la firma actual elaborada (1001) para la conexión asociada.

4. Procedimiento para clasificar conexiones según la reivindicación 3, caracterizado porque el núcleo informático establece (1003, 2011) en dicha primera tabla (1) una segunda correspondencia asociativa entre cada firma actual y una firma par cuyos indicadores de origen son los indicadores de destino de la firma actual y cuyos indicadores de destino son los indicadores de origen de la firma actual.

5. Procedimiento para clasificar conexiones según una de las reivindicaciones 3 ó 4, caracterizado porque:

- el núcleo informático recolecta (1000), dentro de paquetes de datos que pasan por el sistema informático en el seno de conexiones que han de detectarse, la información conducida útil para elaborar una firma al objeto de elaborar (1001) la firma actual cada vez que la información conducida útil es suficiente,

- el núcleo informático utiliza la firma actual así elaborada en tiempo real para detectar una conexión, al objeto de buscar (1002) dentro de dicha primera tabla (1), la primera estructura de datos que corresponde a la firma actual, de asociar (1003) una nueva primera estructura de datos con la conexión detectada cuando no existe ninguna primera estructura de datos que corresponde a la firma actual y de empezar a construir o seguir (2000) construyendo la primera estructura de datos cuando existe una primera estructura de datos que corresponde (1002) a la firma actual, recolectando (1000), dentro de los paquetes de datos, la información conducida útil para construir una primera estructura de datos.

6. Procedimiento para clasificar conexiones según la reivindicación 5, caracterizado porque, cuando la información útil recolectada dentro de un paquete de datos no es suficiente para elaborar una firma, el núcleo informático cataloga la información útil en una segunda tabla estableciendo una correspondencia asociativa entre la información útil que comprende entonces nexos de pertenencia a una misma conexión, hasta que la información útil sea suficiente para elaborar la firma actual.

7. Procedimiento para clasificar conexiones según una de las anteriores reivindicaciones, caracterizado porque:

- el núcleo informático repasa (1006) los nombres de protocolos utilizados de la sucesión ordenada en la estructura de datos que construye para detectar (1007) cada nombre de protocolo con conexión dinámica,

- para cada nombre de protocolo con conexión dinámica detectado, el núcleo informático somete (1008) la información conducida al mecanismo autoidentificador asociado al nombre detectado con objeto de determinar si existe una conexión dinámica ulterior y, si existe una conexión dinámica ulterior, de asociarle (1009) una segunda estructura de datos establecida para contener una sucesión ordenada de potenciales nombres de protocolos que comienza con el nombre de protocolo llamado de base.

8. Procedimiento para clasificar conexiones según las reivindicaciones 3 y 7, caracterizado porque el núcleo informático cataloga (1009) cada segunda estructura de datos en una segunda tabla (2) estableciendo una correspondencia asociativa entre cada segunda estructura de datos y una firma potencial elaborada por el mecanismo autoidentificador asociado al nombre detectado.

9. Procedimiento para clasificar conexiones según la reivindicación 8, caracterizado porque el núcleo informático construye además la primera estructura de datos:

- buscando (2004) las sucesiones ordenadas de potenciales nombres de protocolos en las cuales viene incluida la sucesión ordenada de nombres de protocolos utilizados y

- cuando existe (2005) una sucesión ordenada de potenciales nombres de protocolos cuya firma potencial corresponde a la firma actual, completando (2003) la primera estructura de datos por medio de la segunda estructura de datos.

FIG.1.

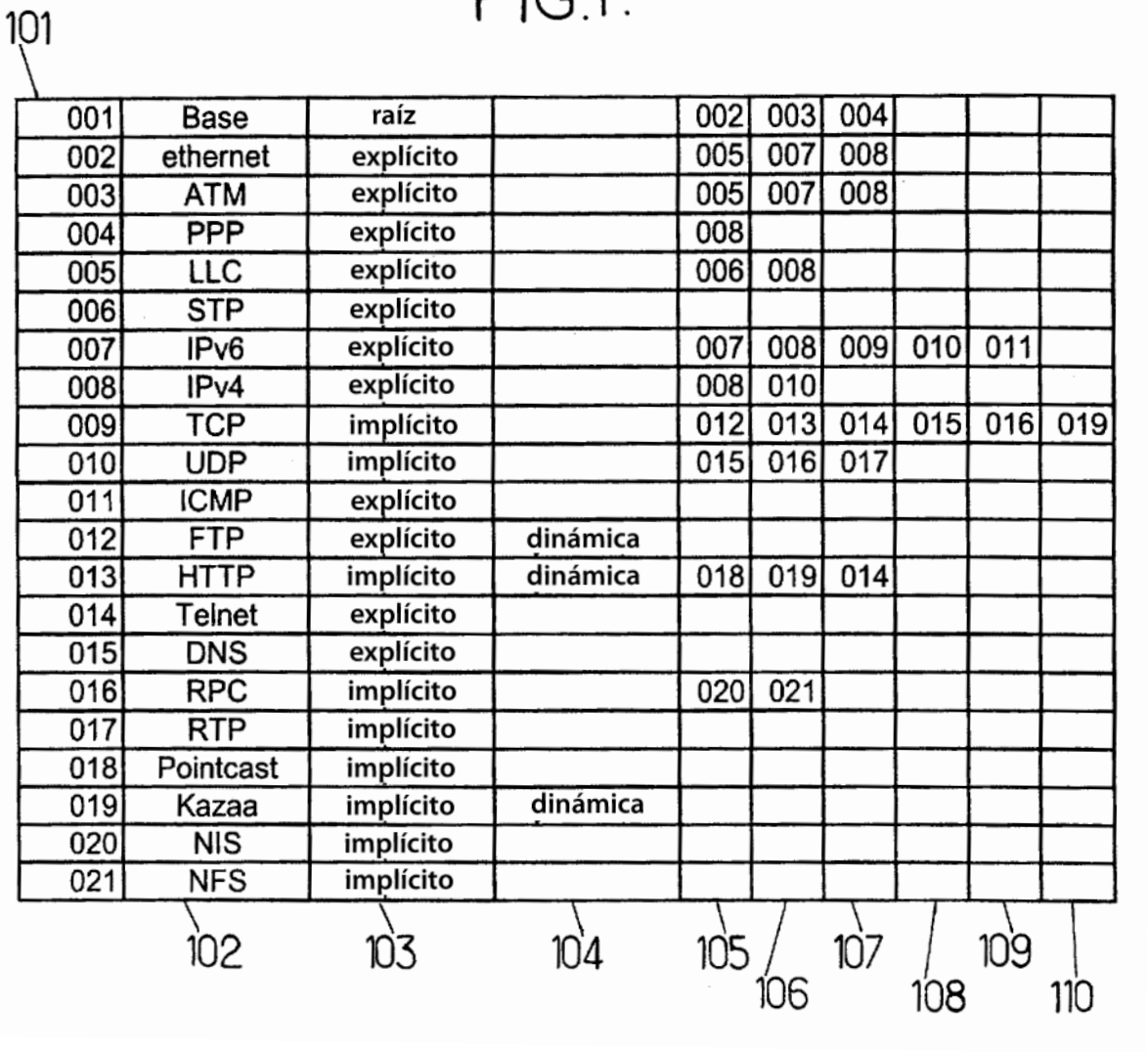

101
001 Base raíz 002 003 004
002 ethernet explícito 005 007 008
003 ATM explícito 005 007 008
004 PPP explícito 008
005 LLC explícito 006 008
006 STP explícito
007 IPv6 explícito 007 008 009 010 011
008 IPv4 explícito 008 010
009 TCP implícito 012 013 014 015 016 019
010 UDP implícito 015 016 017
011 ICMP explícito
012 FTP explícito dinámica
013 HTTP implícito dinámica 018 019 014
014 Telnet explícito
015 DNS explícito
016 RPC implícito 020 021
017 RTP implícito
018 Pointcast implícito
019 Kazaa implícito dinámica
020 NIS implícito
021 NFS implícito
102
103
104
105
106
107
108
109
110

FIG.2.

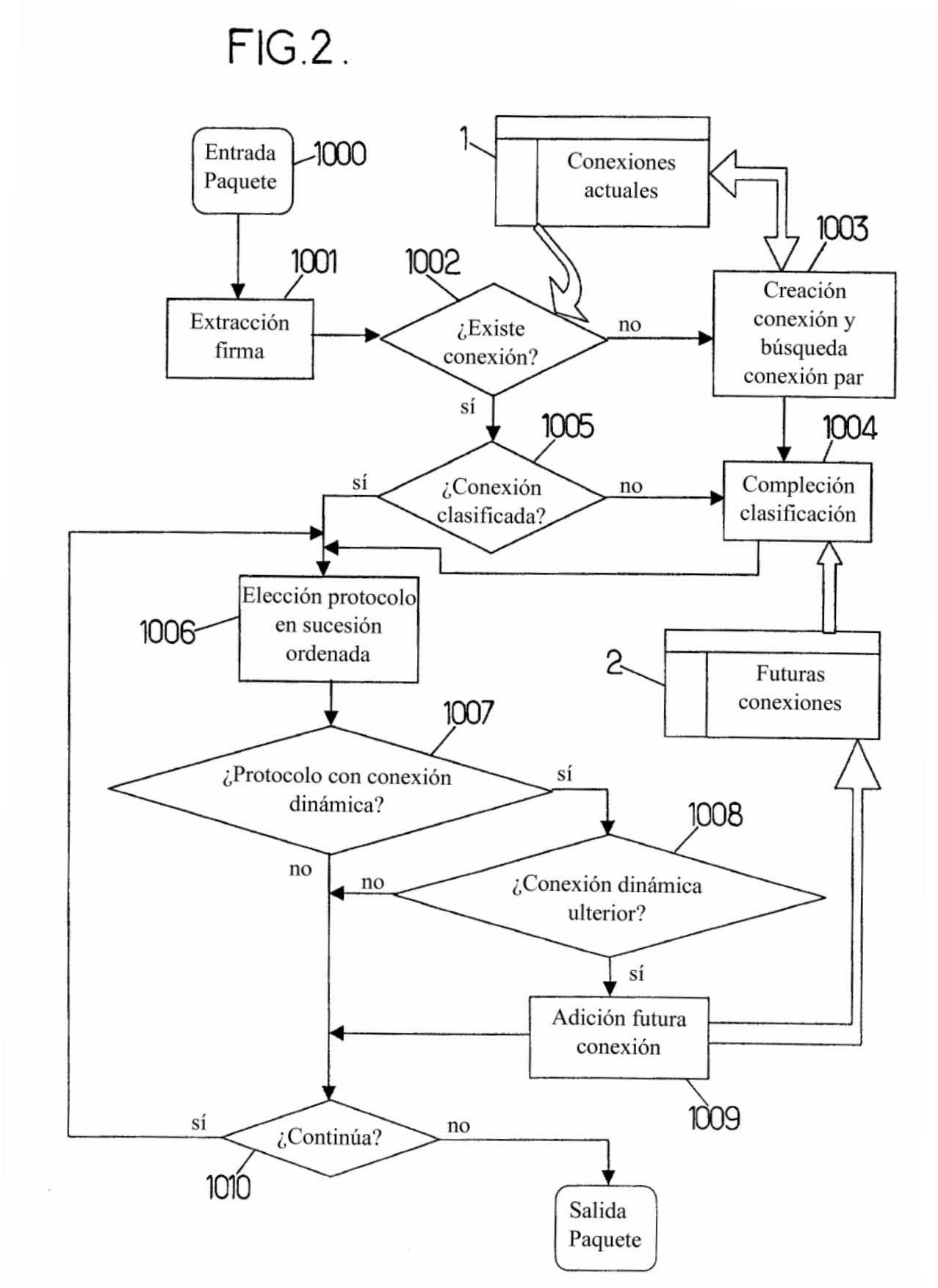
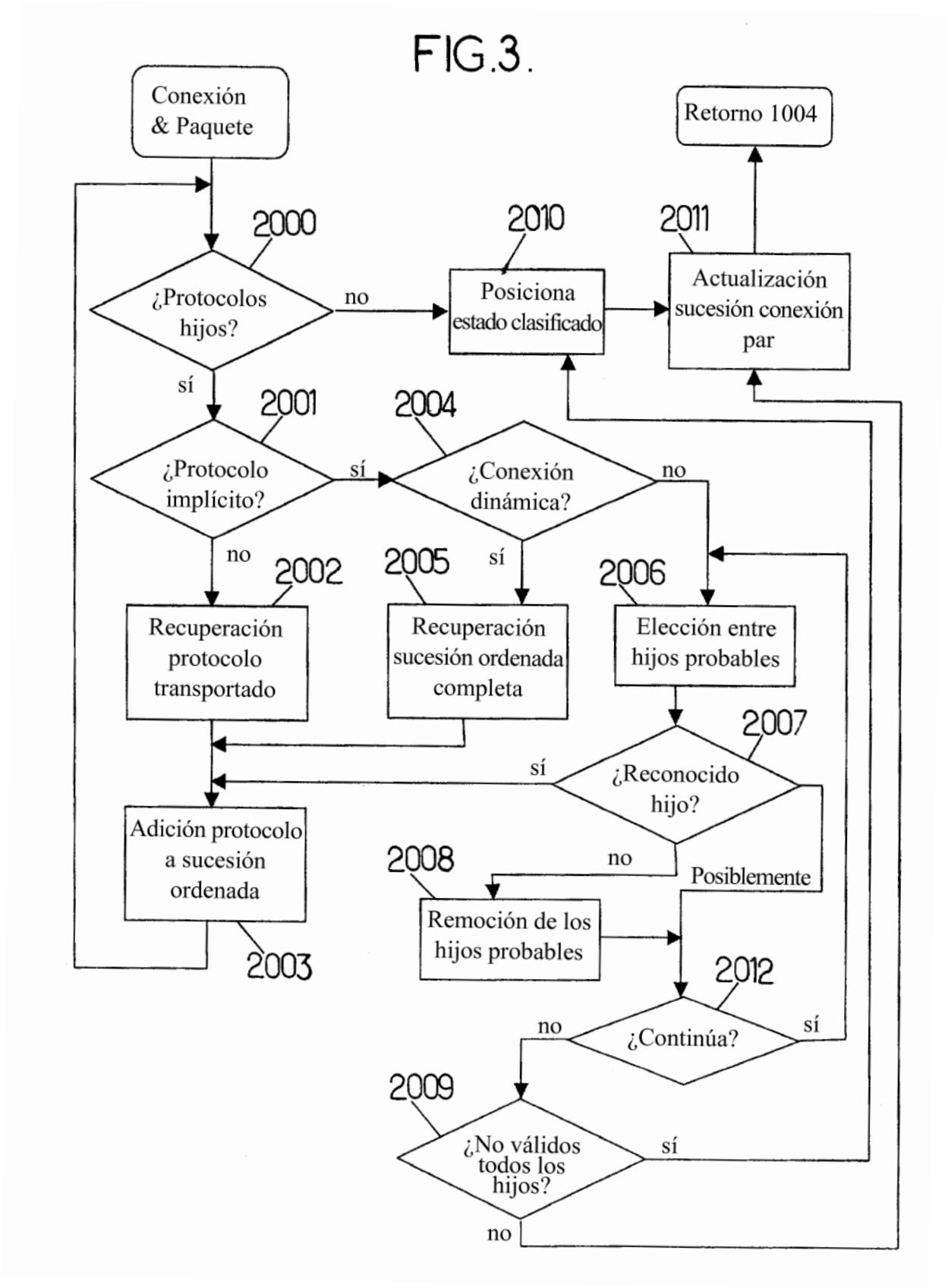

Entrada Paquete
1000
1
Conexiones actuales
1003
1001
1002
Extracción firma
¿Existe conexión?
no
Creación conexión y búsqueda conexión par
sí
1005
1004
sí
¿Conexión clasificada?
no
Compleción clasificación
Elección protocolo en sucesión ordenada
1006
2
Futuras conexiones
1007
¿Protocolo con conexión dinámica?
sí
1008
no
no
¿Conexión dinámica ulterior?
sí
Adición futura conexión
1009
sí
¿Continúa?
no
1010
Salida Paquete

FIG.3.


Conexión & Paquete
Retorno 1004
2000
¿Protocolos hijos?
no
2010
Posiciona estado clasificado
2011
Actualización sucesión conexión par
sí
2001
¿Protocolo implícito?
sí
2004
¿Conexión dinámica?
no
no
sí
2002
Recuperación protocolo transportado
2005
Recuperación sucesión ordenada completa
2006
Elección entre hijos probables
2007
¿Reconocido hijo?
sí
Adición protocolo a sucesión ordenada
2003
2008
no
Posiblemente
Remoción de los hijos probables
2012
¿Continúa?
no
sí
2009
¿No válidos todos los hijos?
sí
no