(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 384 964**

(51) Int. CI.:
H04L 29/12 (2006.01)
H04L 29/08 (2006.01)
H04L 29/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Número de solicitud europea: **03007907 .3**
(96) Fecha de presentación: **07.04.2003**
(97) Número de publicación de la solicitud: **1361728**
(97) Fecha de publicación de la solicitud: **12.11.2003**

(54) Título: **Infraestructura de seguridad y procedimiento para un protocolo de resolución de nombres del mismo nivel (PNRP)**

(30) Prioridad:
29.04.2002 US 134780

(45) Fecha de publicación de la mención BOPI:
16.07.2012

(45) Fecha de la publicación del folleto de la patente:
16.07.2012

(73) Titular/es:
MICROSOFT CORPORATION
ONE MICROSOFT WAY
REDMOND, WA 98052, US

(72) Inventor/es:
Gupta, Rohit;
Gavrilescu, Alexandru;
Miller, John L. y
Wheeler, Graham A.

(74) Agente/Representante:
Carpintero López, Mario

DESCRIPCIÓN

Infraestructura de seguridad y procedimiento para un protocolo de resolución de nombres del mismo nivel (PNRP)

Campo de la invención

La presente invención versa, en general, acerca de protocolos del mismo nivel, y más en particular acerca de infraestructuras marco de seguridad para los protocolos del mismo nivel.

Antecedentes de la invención

La comunicación entre iguales y, de hecho, todos los tipos de comunicaciones dependen de la posibilidad de establecer conexiones válidas entre entidades seleccionadas. Sin embargo, las entidades pueden tener una o varias direcciones que pueden variar debido a que las entidades se mueven en la red, debido a cambios de topología, o debido a que no se puede renovar el alquiler de la dirección. Una solución arquitectónica clásica para este problema de asignación de direcciones es, por lo tanto, asignar a cada entidad un nombre estable, y "convertir" este nombre en una dirección actual cuando se necesita una conexión. Esta conversión de nombre a dirección debe ser muy robusta, y también debe permitir actualizaciones sencillas y rápidas.

Para aumentar la probabilidad de que los que buscan conectarse con la dirección de una entidad puedan encontrarla, muchos protocolos del mismo nivel permiten a las entidades publicar su dirección a través de diversos mecanismos. Algunos protocolos también permiten a un cliente adquirir conocimiento de las direcciones de otras entidades a través del procesamiento de peticiones de otros en la red. De hecho, esta adquisición de conocimiento de direcciones permite una operación con éxito de estas redes del mismo nivel. Es decir, cuanto mejor sea la información acerca de otras unidades del mismo nivel en la red, mayor será la probabilidad de que una búsqueda de un recurso particular converja.

Sin embargo, sin una infraestructura robusta de seguridad subyacente al protocolo del mismo nivel, las entidades maliciosas pueden alterar fácilmente la capacidad de que tales sistemas del mismo nivel converjan. Tales alteraciones pueden estar causadas, por ejemplo, por una entidad que se dedica al robo de identidades. En tal ataque de robo de identidad en la red del mismo nivel, un nodo malicioso publica información de direcciones para ID con las que no tiene una relación autorizada, es decir, no es ni el dueño ni un miembro del grupo, etc. Una entidad maliciosa también podría interceptar y/o responder antes de que responda el nodo bueno, aparentando ser de esta manera el nodo bueno.

Una entidad maliciosa también podría dificultar la resolución PNRP al inundar la red con información mala, de forma que otras entidades en la red tenderían a remitir peticiones a nodos inexistentes (lo que afectaría de forma adversa a la convergencia de búsquedas), o a nodos controlados por el atacante. Esto también podría conseguirse al modificar el paquete de CONVERSIÓN utilizado para descubrir recursos antes de remitirlo, o al enviar una RESPUESTA inválida de vuelta al solicitante que generó el paquete de CONVERSIÓN. Una entidad maliciosa también podría intentar alterar la operación de la red del mismo nivel al intentar asegurarse de que las búsquedas no convergerán, por ejemplo, en vez de remitir la búsqueda a un nodo en su memoria intermedia que está más cerca de la ID para ayudar en la convergencia de la búsqueda, remitir la búsqueda a un nodo que está más lejos de la ID solicitada. De forma alternativa, la entidad maliciosa simplemente podría no responder en absoluto a la solicitud de búsqueda. Además, la resolución PNRP podría ser dificultada por un nodo malicioso que envíe un mensaje BYE inválido en nombre de una ID válida. Como resultado, otros nodos en la nube eliminarán esta ID válida de su memoria intermedia, reduciendo el número de nodos válidos almacenados en la misma.

Aunque la validación de un certificado de dirección puede evitar el problema del robo de la identidad, esta es ineficaz contra este segundo tipo de ataque que dificulta la resolución PNRP. Un atacante puede continuar generando certificados verificables de direcciones (o puede hacer que se generen previamente) e inundando las ID correspondientes en la nube del mismo nivel. Si cualquiera de los nodos intenta verificar la propiedad de la ID, el atacante podría verificar que es el dueño de las ID inundadas porque, de hecho, lo es. Sin embargo, si el atacante logra generar suficientes ID puede llevar la mayor parte de las búsquedas del mismo nivel a uno de los nodos controlados por él. En este punto, el atacante puede controlar y dirigir bastante bien la operación de la red.

Si el protocolo del mismo nivel requiere que toda la información de direcciones nuevas sea verificada en primer lugar para evitar el problema de robo de la identidad expuesto anteriormente, un tercer tipo de ataque se pone a disposición de las entidades maliciosas. Este ataque al que son susceptibles estos tipos de redes del mismo nivel es una forma de ataque de una denegación de servicio (DoS). Si todos los nodos que se enteran de nuevos registros intentan llevar a cabo la comprobación de la propiedad de la ID, se producirá una tormenta de actividad de la red contra el dueño anunciado de la ID. Sacando partido de esta debilidad, un atacante podría montar un ataque IP de DoS contra un cierto objetivo al hacer muy popular este objetivo. Por ejemplo, si una entidad maliciosa anuncia la dirección IP de la Web de Microsoft como el IP de las ID, todos los nodos en la red del mismo nivel que reciban este IP anunciado intentarán conectarse a ese IP (IP del servidor Web de Microsoft) para verificar la autenticidad del registro. Por supuesto, el servidor de Microsoft no podrá verificar la propiedad de la ID dado que el atacante generó

esta información. Sin embargo, el daño ya está hecho. Es decir, el atacante acaba de conseguir convencer a una buena parte de la comunidad del mismo nivel a atacar a Microsoft.

Otro tipo de ataque de DoS que abruma a un nodo o una nube al agotar uno o más recursos es perpetrado por un nodo malicioso que envía un gran volumen de PAC inválidos/válidos a un único nodo, por ejemplo, utilizando paquetes de INUNDACIÓN/CONVERSIÓN/SOLICITUD). El nodo que recibe estos PAC consumirá toda su CPU intentando verificar todos los PAC. De forma similar, al enviar paquetes inválidos de INUNDACIÓN/CONVERSIÓN, un nodo malicioso conseguirá una multiplicación de los paquetes en la nube. Es decir, el nodo malicioso puede consumir el ancho de banda de la red para una nube de PNRP utilizando un número reducido de tales paquetes dado que el nodo al que son enviados estos paquetes responderá enviando paquetes adicionales. Un nodo malicioso también puede conseguir la multiplicación del ancho de banda de la red al enviar mensajes falsos de PETICIÓN a los que los nodos buenos responderán mediante INUNDACIÓN de los PAC, que tienen un tamaño mayor que la PETICIÓN.

Un nodo malicioso también puede perpetrar un ataque en la nube de PNRP al dificultar la sincronización inicial del nodo. Es decir, para unirse a la nube de PNRP un nodo intenta conectarse a uno de los nodos ya presentes en la nube de PNRP. Si el nodo intenta conectarse al nodo malicioso, puede ser controlado completamente por ese nodo malicioso. Además, un nodo malicioso puede enviar paquetes inválidos de PETICIÓN cuando hay implicados dos nodos buenos en el procedimiento de sincronización. Este es un tipo de ataque de DoS que dificultará la sincronización dado que los paquetes inválidos de PETICIÓN iniciarán la generación de mensajes de INUNDACIÓN en respuesta.

El documento: E. Sit, R. Morris: “Security Considerations for Peer-to-Peer Distributed Hash Tables”, en los Proceedings of the First International Workshop on Peer-to-Peer Systems, (IPTPSO2), 7 de marzo de 2002, Cambridge, Massachusetts, EE. UU., menciona que los sistemas del mismo nivel presentan un problema interesante de seguridad dado que no hay ningún sistema central que proteger. Un nodo malicioso puede diseñar una situación en la que puede enviar una respuesta no solicitada a una consulta. La mejor defensa contra esto sería emplear técnicas estándar de autentificación tales como firmas digitales o códigos de autentificación de mensajes. Una defensa más razonable podría ser incluir un valor aleatorio de uso único con cada consulta y hacer que el extremo remoto repita el valor de uso único en su respuesta.

Por lo tanto, existe una necesidad en la técnica de mecanismos de seguridad que garantizarán la integridad de la nube de P2P al evitar o mitigar el efecto de tales ataques.

Breve resumen de la invención

Los conceptos inventivos dados a conocer en la presente solicitud implican un procedimiento nuevo y mejorado para inhibir la capacidad de un nodo malicioso para alterar una operación normal de una red del mismo nivel. Específicamente, la presente invención presenta procedimientos para abordar diversos tipos de ataques que pueden ser lanzados por un nodo malicioso, incluyendo ataques de robo de identidad, ataques de denegación de servicio, ataques que simplemente intentan dificultar la resolución de direcciones en la red del mismo nivel, al igual que ataques que intentan dificultar la capacidad de un nuevo nodo para unirse a la red del mismo nivel, y a participar en la misma.

Los procedimientos y la infraestructura de seguridad presentados permiten que los nodos utilicen tanto identidades seguras como no seguras al hacer que sean autoverificables. Cuando es necesario u oportuno, se valida la propiedad de la ID al superponer la validación en los mensajes existentes o, si es necesario, al enviar un pequeño mensaje de consulta. Se reduce la probabilidad de conectarse inicialmente con un nodo malicioso al seleccionar de forma aleatoria a qué nodo conectarse. Además, se identifica la información de los nodos maliciosos y puede no ser tenida en cuenta al mantener la información acerca de comunicaciones anteriores que requerirán una respuesta futura. Los ataques de denegación de servicio son inhibidos al permitir que el nodo ignore solicitudes cuando su uso de recursos supera un límite predeterminado. Se reduce la capacidad para que un nodo malicioso elimine un nodo válido al requerir que los certificados de revocación sean firmados por el nodo que va a ser eliminado.

Según una realización de la presente invención, se presenta un procedimiento para generar un certificado no seguro autoverificable de dirección del mismo nivel (PAC) que evitará que un nodo malicioso publique la identificación segura de otro nodo en un PAC no seguro en la red del mismo nivel. Este procedimiento comprende las etapas de generar un PAC no seguro para un recurso descubrible en la red del mismo nivel. El recurso tiene una identificación (ID) del mismo nivel. Además, el procedimiento incluye la etapa de incluir un identificador de recursos uniforme (URI) en el PAC no seguro del que se deriva la ID del mismo nivel. Preferentemente, el URI tiene el formato “p2p//URI”. La ID del mismo nivel también puede ser no segura.

En una realización adicional, se presenta un procedimiento para validar de forma oportuna un certificado de dirección del mismo nivel en un primer nodo en una red del mismo nivel. Este primer nodo utiliza una memoria intermedia de múltiples niveles para el almacenamiento de certificados de direcciones del mismo nivel, y el procedimiento comprende las etapas de recibir un certificado de dirección del mismo nivel (PAC) supuestamente de un segundo nodo y determinar en qué nivel de la memoria intermedia de múltiples niveles va a almacenarse el PAC.

Cuando se va a almacenar el PAC en uno de los dos niveles más bajos de la memoria intermedia, el procedimiento pone el PAC en una lista de reserva, genera un mensaje de CONSULTA que contiene una ID del PAC que va a ser validado, y transmite el mensaje de CONSULTA al segundo nodo. Cuando se va a almacenar el PAC en un nivel superior de la memoria intermedia distinto de uno de los dos niveles más bajos de la memoria intermedia, el procedimiento almacena el PAC en el nivel superior de la memoria intermedia marcado como “no validado”. En este caso, el PAC será validado la primera vez que sea utilizado. El procedimiento también puede solicitar una cadena de certificación para el PAC.

En una realización preferente, la generación del mensaje de CONSULTA comprende la etapa de generar una ID de transacción que será incluida en el mensaje de CONSULTA. Cuando se recibe un mensaje de AUTORIZACIÓN procedente del segundo nodo en respuesta al mensaje de CONSULTA, se elimina el PAC de la lista de reserva y es almacenado en uno de los dos niveles más bajos de la memoria intermedia. Si se solicitó una cadena de certificación, se analiza el mensaje de AUTORIZACIÓN para determinar si la cadena de certificación está presente y es válida. Si lo está, se almacena el PAC en uno de los dos niveles más bajos de la memoria intermedia, y si no lo está se borra. También se puede utilizar una ID de transacción en una realización de la invención para garantizar que el mensaje de AUTORIZACIÓN es en respuesta a una comunicación anterior.

En una realización adicional de la presente invención, se presenta un procedimiento para descubrir un nodo en una red del mismo nivel de forma que reduzca la probabilidad de conectarse a un nodo malicioso. Este procedimiento comprende las etapas de radiodifundir un mensaje de descubrimiento en la red del mismo nivel sin incluir ninguna ID registrada localmente, recibir una respuesta de un nodo en la red del mismo nivel, y establecer una relación entre iguales con el nodo. En una realización, la etapa de recibir una respuesta de un nodo comprende la etapa de recibir una respuesta de al menos dos nodos en la red del mismo nivel. En esta situación, la etapa de establecer una relación entre iguales con el nodo comprende las etapas de seleccionar forma aleatoria uno de los al menos dos nodos y establecer una relación entre iguales con el uno seleccionado de forma aleatoria de los al menos dos nodos.

En una realización adicional más de la presente invención, se presenta un procedimiento para inhibir un ataque de denegación de servicio en base a un procedimiento de sincronización en una red del mismo nivel. Este procedimiento comprende las etapas de recibir un mensaje de SOLICITUD que solicita una sincronización de la memoria intermedia de un primer nodo que contiene un certificado de dirección del mismo nivel (PAC), analizar el PAC para determinar su validez, y desechar el paquete de SOLICITUD cuando la etapa de analizar el PAC determina que el PAC no es válido. Preferentemente, cuando la etapa de analizar el PAC determina que el PAC es válido, el procedimiento comprende, además, las etapas de generar un valor de uso único, codificar el valor de uso único con una clave pública del primer nodo, generar un mensaje de ANUNCIO que incluye el valor de uso único codificado, y enviar el mensaje de ANUNCIO al primer nodo. Cuando se recibe un mensaje de PETICIÓN procedente del primer nodo, el procedimiento analiza el mensaje de PETICIÓN para determinar si el primer nodo pudo decodificar el valor de uso único codificado y procesa el mensaje de PETICIÓN cuando el primer nodo pudo decodificar el valor de uso único codificado.

Preferentemente, este procedimiento comprende, además, las etapas de mantener la información de conexión que identifica específicamente a la comunicación con el primer nodo, analizar el mensaje de PETICIÓN para garantizar que está relacionado específicamente con el mensaje de ANUNCIO, y rechazar el mensaje de PETICIÓN cuando no está relacionado específicamente con el mensaje de ANUNCIO. En una realización la etapa de mantener la información de conexión que identifica específicamente la comunicación con el primer nodo comprende las etapas de calcular una primera posición del bit como la clave calculada del valor de uso único y de la identidad del primer nodo, y poner un bit en la primera posición del bit en un vector de bits. Cuando esto se realiza, la etapa de analizar el mensaje de PETICIÓN comprende las etapas de extraer el valor de uso único y la identidad del primer nodo del mensaje de PETICIÓN, calcular una segunda posición del bit como la clave calculada del valor de uso único y de la identidad del primer nodo del mensaje de PETICIÓN, analizar el vector de bits para determinar si tiene un bit puesto correspondiente a la segunda posición del bit, e indicar que la PETICIÓN no está relacionada específicamente con el mensaje de ANUNCIO cuando la etapa de analizar el vector de bits no encuentra un bit puesto correspondiente a la segunda posición del bit. De forma alternativa, se puede utilizar directamente el valor de uso único como la posición del bit. En este caso, cuando se recibe la PETICIÓN, se comprueba la posición del bit correspondiente al valor de uso único adjunto. Si está puesto, esta es una PETICIÓN válida y se pone a cero la posición del bit. De lo contrario, esta es una PETICIÓN inválida o un ataque de reproducción, y se descarta la PETICIÓN.

En una realización adicional más de la presente invención, un procedimiento para inhibir un ataque de denegación de servicio en base a un procedimiento de sincronización en una red del mismo nivel comprende las etapas de recibir un mensaje de PETICIÓN supuestamente de un primer nodo, determinar si el mensaje de PETICIÓN es en respuesta a una comunicación anterior con el primer nodo, y rechazar el mensaje de PETICIÓN cuando el mensaje de PETICIÓN no es en respuesta a una comunicación anterior con el primer nodo. Preferentemente, la etapa de determinar si el mensaje de PETICIÓN es en respuesta a una comunicación anterior comprende las etapas de extraer un valor de uso único y una identidad supuestamente del primer nodo del mensaje de PETICIÓN, calcular una posición del bit como la clave calculada del valor de uso único y de la identidad, analizar un vector de bits para determinar si tiene un bit puesto correspondiente a la posición del bit, e indicar que la PETICIÓN no es en respuesta

a una comunicación anterior con el primer nodo cuando no hay ningún bit puesto correspondiente a la posición del bit.

También se presenta un procedimiento para inhibir los ataques de denegación de servicio en base a un consumo de recursos del nodo en una red del mismo nivel. Este procedimiento comprende las etapas de recibir un mensaje de un nodo en la red del mismo nivel, analizar el uso actual de los recursos, y rechazar el procesamiento del mensaje cuando el uso actual de los recursos es superior a un nivel predeterminado. Cuando se recibe un mensaje de CONVERSIÓN, la etapa de rechazar un procesamiento del mensaje comprende la etapa de enviar un mensaje de AUTORIZACIÓN al primer nodo. Este mensaje de AUTORIZACIÓN contiene una indicación de que el mensaje de CONVERSIÓN no será procesado dado que el uso actual de los recursos es demasiado elevado. Cuando se recibe un mensaje de INUNDACIÓN que contiene un certificado de dirección del mismo nivel (PAC) y el procedimiento determina que el PAC debería ser almacenado en uno de los dos niveles más bajos de la memoria intermedia, la etapa de rechazar el procesamiento del mensaje comprende la etapa de colocar el PAC en una lista de reserva para un procesamiento posterior. Si el procedimiento determina que el PAC debería ser almacenado en un nivel superior de la memoria intermedia que los dos niveles más bajos de la memoria intermedia, la etapa de rechazar el procesamiento del mensaje comprende la etapa de rechazar el mensaje de INUNDACIÓN.

En otra realización de la presente invención, se presenta un procedimiento para inhibir ataques de denegación de servicio basado en el consumo del ancho de banda del nodo en una red del mismo nivel. Este procedimiento comprende las etapas de recibir una petición para una sincronización de la memoria intermedia de un nodo en la red del mismo nivel, analizar una métrica que indica un número de sincronizaciones de las memorias intermedias llevadas a cabo en el pasado, y rechazar el procesamiento de la petición de una sincronización de las memorias intermedias cuando el número de sincronizaciones de la memoria intermedia llevadas a cabo en el pasado supera un máximo predeterminado. En una realización adicional, el procedimiento analiza la métrica para determinar el número de sincronizaciones de la memoria intermedia llevadas a cabo durante un periodo de tiempo precedente. En esta realización la etapa de rechazar el procesamiento de la petición comprende la etapa de rechazar el procesamiento de la petición de una sincronización de las memorias intermedias cuando el número de sincronizaciones de la memoria intermedia llevadas a cabo en el periodo de tiempo precedente supera un máximo predeterminado.

En otra realización de la presente invención, un procedimiento para inhibir un ataque de denegación de servicio basado en búsquedas en una red del mismo nivel comprende las etapas de analizar entradas en la memoria intermedia de certificados conocidos de dirección del mismo nivel para determinar nodos apropiados a los que enviar una petición de resolución, seleccionar de forma aleatoria uno de los nodos apropiados, y enviar la petición de resolución al nodo seleccionado de forma aleatoria. En una realización la etapa de seleccionar de forma aleatoria uno de los nodos apropiados comprende la etapa de calcular una probabilidad ponderada para cada uno de los nodos apropiados en base a la distancia de la ID PNRP desde la ID objetivo. Entonces, se determina la probabilidad de escoger un siguiente salto específico como una proporcionalidad inversa a la distancia de la ID entre ese nodo y el nodo objetivo.

En una realización adicional de la presente invención, un procedimiento para inhibir un ataque de denegación de servicio basado en búsquedas en una red del mismo nivel comprende las etapas de recibir un mensaje de RESPUESTA, determinar si el mensaje de RESPUESTA es en respuesta a un mensaje de CONVERSIÓN anterior, y rechazar el mensaje de RESPUESTA cuando el mensaje de RESPUESTA no es en respuesta al mensaje de CONVERSIÓN anterior. Preferentemente, la etapa de determinar si el mensaje de RESPUESTA es en respuesta a un mensaje de CONVERSIÓN anterior comprende las etapas de calcular una posición del bit como una clave calculada de información en el mensaje de RESPUESTA, y analizar un vector de bits para determinar si hay puesto en el mismo un bit correspondiente a la posición del bit.

En una realización en la que el mensaje de RESPUESTA contiene una lista de direcciones, el procedimiento comprende, además, las etapas de determinar si el mensaje de RESPUESTA ha sido modificado en un intento por dificultar la resolución, y rechazar el mensaje de RESPUESTA cuando el mensaje de RESPUESTA ha sido modificado en un intento por dificultar la resolución. Preferentemente, la etapa de determinar si el mensaje de RESPUESTA ha sido modificado en un intento por dificultar la resolución comprende las etapas de calcular una posición del bit como una clave calculada de la lista de direcciones en el mensaje de RESPUESTA, y analizar un vector de bits para determinar si hay puesto en el mismo un bit correspondiente a la posición del bit.

En otra realización de la presente invención, un procedimiento para impedir que un nodo malicioso elimine un nodo válido de la red del mismo nivel comprende las etapas de recibir un certificado de revocación supuestamente del nodo válido que tiene un certificado de dirección del mismo nivel (PAC) almacenado en la memoria intermedia, y verificar que el certificado de revocación está firmado por el nodo válido.

Breve descripción de los dibujos

Los dibujos adjuntos incorporados en la memoria, y que forman parte de la misma, ilustran varios aspectos de la presente invención, y junto con la descripción sirven para explicar los principios de la invención. En los dibujos:

La FIG. 1 es un diagrama de bloques que ilustra, en general, un sistema ejemplar de ordenador en el que reside la presente invención;

la FIG. 2 es un diagrama simplificado de flujo que ilustra aspectos de seguridad del procesamiento de paquetes de AUTORIZACIÓN según una realización de la presente invención;

la FIG. 3 es un diagrama simplificado de flujo de procesamiento de comunicaciones que ilustra aspectos de seguridad de una fase de sincronización de descubrimiento P2P según una realización de la presente invención;

la FIG. 4 es un diagrama simplificado de flujo que ilustra aspectos de seguridad de un procesamiento de paquetes de CONVERSIÓN según una realización de la presente invención;

la FIG. 5 es un diagrama simplificado de flujo que ilustra aspectos de seguridad de un procesamiento de paquetes de INUNDACIÓN según una realización de la presente invención; y

la FIG. 6 es un diagrama simplificado de flujo que ilustra aspectos de seguridad de un procesamiento de paquetes de RESPUESTA según una realización de la presente invención.

Aunque se describirá la invención en conexión con ciertas realizaciones preferentes, no hay ningún intención por limitarla a esas realizaciones. Al contrario, la intención es abarcar todas las alternativas, las modificaciones y los equivalentes que se incluyen dentro del alcance de la invención como se define en las reivindicaciones adjuntas.

Descripción detallada de la invención

Con referencia a los dibujos, en los que los números similares de referencia hacen referencia a elementos similares, se ilustra la invención implementada en un entorno informático adecuado. Aunque no se requiere, se describirá la invención en el contexto general de instrucciones ejecutables por un ordenador, tales como módulos de programas, que son ejecutadas por un ordenador personal. En general, los módulos de programas incluyen rutinas, programas, objetos, componentes, estructuras de datos, etc. que llevan a cabo tareas particulares o implementan tipos particulares de datos abstractos. Además, los expertos en la técnica apreciarán que se puede poner en práctica la invención con otras configuraciones del sistema de ordenador, incluyendo dispositivos de mano, sistemas de múltiples procesadores, electrónica de consumo programable o basada en microprocesadores, PC de red, miniordenadores, ordenadores centrales, y similares. También se puede poner en práctica la invención en entornos informáticos distribuidos en los que se llevan a cabo tareas por medio de dispositivos remotos de procesamiento que están unidos a través de una red de comunicaciones. En un entorno informático distribuido, los módulos de programas pueden estar ubicados tanto en dispositivos locales y remotos de almacenamiento de memoria.

La Figura 1 ilustra un ejemplo de un entorno adecuado 100 de sistema informático en el que se puede implementar la invención. El entorno 100 del sistema informático solo es un ejemplo de un entorno informático adecuado y no se pretende que sugiera ninguna limitación. Tampoco se debería interpretar que el entorno informático 100 tenga ninguna dependencia ni requerimiento relacionado con un componente cualquiera o con una combinación de componentes ilustrados en el entorno operativo ejemplar 100.

La invención es operativa con numerosos entornos o configuraciones adicionales del sistema informático de uso general o de uso especial. Ejemplos de entornos, configuraciones y/o sistemas informáticos bien conocidos que pueden ser adecuados para ser utilizados con la invención incluyen, sin limitación, ordenadores personales, ordenadores servidores, dispositivos de mano o portátiles, sistemas de múltiples procesadores, sistemas basados en microprocesadores, decodificadores, electrónica de consumo programable, PC de red, miniordenadores, ordenadores centrales, entornos informáticos distribuidos que incluyen cualquiera de los anteriores sistemas o dispositivos, y similares.

Se puede describir la invención en el contexto general de instrucciones ejecutables por un ordenador, tales como módulos de programas, que son ejecutadas por un ordenador. En general, los módulos de programas incluyen rutinas, programas, objetos, componentes, estructuras de datos, etc., que llevan a cabo tareas particulares o implementan tipos particulares de datos abstractos. También se puede poner en práctica la invención en entornos informáticos distribuidos en los que las tareas son llevadas a cabo por dispositivos remotos de procesamiento que están unidos a través de una red de comunicaciones. En un entorno informático distribuido, los módulos de programas pueden estar ubicados tanto en medios locales como remotos de almacenamiento en ordenador incluyendo dispositivos de almacenamiento de memoria.

Con referencia a la Figura 1, un sistema ejemplar para implementar la invención incluye un dispositivo informático de uso general en forma de un ordenador 110. Los componentes del ordenador 110 pueden incluir, sin limitación, una unidad 120 de procesamiento, una memoria 130 del sistema, y un bus 121 del sistema que acopla diversos componentes del sistema incluyendo la memoria de sistema a la unidad 120 de procesamiento. El bus 121 del sistema puede ser cualquiera de varios de tipos de estructuras de bus incluyendo un bus de memoria o un controlador de memoria, un bus de periféricos, y un bus local que utiliza cualquiera de una variedad de arquitecturas de bus. A modo de ejemplo, sin limitación, tales arquitecturas incluyen el bus de arquitectura industrial normalizada

(ISA), el bus de arquitectura de microcanal (MCA), el bus de ISA mejorada (EISA), el bus local de la Asociación de estándares de vídeo electrónico (VESA), y el bus de interconexión de componentes periféricos (PCI) también conocido como bus de entresuelo.

Normalmente, el ordenador 110 incluye una variedad de medios legibles por un ordenador. Los medios legibles por un ordenador pueden ser cualquier medio disponible que pueda ser objeto de acceso por el ordenador 110 e incluye tanto medios volátiles como no volátiles, medios extraíbles y no extraíbles. A modo de ejemplo, y no de limitación, los medios legibles por un ordenador pueden comprender medios de almacenamiento de ordenador y medios de comunicaciones. Los medios de almacenamiento de ordenador incluyen tanto medios volátiles como no volátiles, extraíbles y no extraíbles implementados en cualquier procedimiento o tecnología para el almacenamiento de información tales como instrucciones legibles por un ordenador, estructuras de datos, módulos de programas u otros datos. El medio de almacenamiento de ordenador incluye, sin limitación, RAM, ROM, EEPROM, memoria *flash* u otra tecnología de memoria, CD-ROM, discos versátiles digitales (DVD) u otro almacenamiento de disco óptico, cintas magnéticas en casetes, cintas magnéticas, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, o cualquier otro medio que pueda ser utilizado para almacenar la información deseada y que pueda ser objeto de acceso por el ordenador 110. Normalmente, el medio de comunicación implementa instrucciones legibles por un ordenador, estructuras de datos, módulos de programas u otros datos en una señal modulada de datos tal como una onda portadora u otro mecanismo de transporte e incluye cualquier medio de suministro de información. La expresión “señal modulada de datos” significa una señal que tiene una o más de sus características fijadas o cambiadas de tal forma que se codifique información en la señal. A modo de ejemplo, y no de limitación, el medio de comunicación incluye medios alámbricos tales como una red alámbrica o una conexión cableada directa, y medios inalámbricos tales como acústico, RF, infrarrojo y otros medios inalámbricos. Las combinaciones de cualquiera de los anteriores también deberían ser incluidas dentro del alcance de los medios legibles por un ordenador.

La memoria 130 del sistema incluye medios de almacenamiento de ordenador en forma de memoria volátil y/o no volátil tales como memoria 131 de solo lectura (ROM) y memoria 132 de acceso aleatorio (RAM). Normalmente hay almacenado un sistema básico 133 de entrada/salida (BIOS), que contiene las rutinas básicas que ayudan a transferir información entre elementos dentro del ordenador 110, tal como durante el arranque, en la ROM 131. Normalmente, la RAM 132 contiene datos y/o módulos de programas que son inmediatamente accesibles por la unidad 120 de procesamiento y/u operados por ella en la actualidad. A modo de ejemplo, y no de limitación, la Figura 1 ilustra un sistema operativo 134, programas 135 de aplicación, otros módulos 136 de programas, y datos 137 de programas.

El ordenador 110 también puede incluir otros medios extraíbles/no extraíbles, volátiles/no volátiles de almacenamiento de ordenador. Únicamente a modo de ejemplo, la Figura 1 ilustra una unidad 141 de disco duro que lee de medios magnéticos no extraíbles no volátiles, o escribe en los mismos, una unidad 151 de disco magnético que lee de un disco magnético 152 extraíble no volátil, o escribe en el mismo, y una unidad 155 de disco óptico que lee de un disco óptico 156 extraíble no volátil tal como un CD ROM u otros medios ópticos. Otros medios extraíbles/no extraíbles, volátiles/no volátiles de almacenamiento de ordenador que pueden ser utilizados en el entorno operativo ejemplar incluyen, sin limitación, cintas magnéticas en casetes, tarjetas de memoria *flash*, discos versátiles digitales, cinta de vídeo digital, RAM de estado sólido, ROM de estado sólido, y similares. Normalmente, la unidad 141 de disco duro está conectada al bus 121 del sistema a través de una interfaz de memoria no extraíble tal como la interfaz 140, y normalmente la unidad 151 de disco magnético y la unidad 155 de disco óptico están conectadas al bus 121 del sistema por medio de una interfaz de memoria extraíble, tal como la interfaz 150.

Las unidades y sus medios asociados de almacenamiento de ordenador expuestos anteriormente e ilustrados en la Figura 1, proporcionan un almacenamiento de instrucciones legibles por un ordenador, estructuras de datos, módulos de programas y otros datos para el ordenador 110. En la Figura 1, por ejemplo, se ilustra la unidad 141 de disco duro como un sistema operativo 144 de almacenamiento, programas 145 de aplicación, otros módulos 146 de programas, y datos 147 de programas. Se debe hacer notar que estos componentes pueden ser bien los mismos o bien distintos del sistema operativo 134, de los programas 135 de aplicación, de otros módulos 136 de programas, y de datos 137 de programas. En el presente documento, se les dan distintos números al sistema operativo 144, a los programas 145 de aplicación, a otros módulos 146 de programas, y a los datos 147 de programas para ilustrar que, como mínimo, son distintas copias. Un usuario puede introducir instrucciones e información en el ordenador 110 a través de dispositivos de entrada tales como un teclado 162 y un dispositivo 161 de puntero, denominado habitualmente ratón, bola de mando o alfombrilla táctil. Otros dispositivos (no mostrados) de entrada pueden incluir un micrófono, una palanca de juego, un mando de juegos, una antena parabólica, un escáner, o similares. Estos y otros dispositivos de entrada están conectados a menudo a la unidad 120 de procesamiento a través de una interfaz 160 de entrada del usuario que está acoplada al bus del sistema, pero puede estar conectada por medio de otra interfaz y estructuras de bus, tales como un puerto paralelo, un puerto para juegos, o un bus serie universal (USB). También hay conectado un monitor 191 u otro tipo de dispositivo de visualización al bus 121 del sistema por medio de una interfaz, tal como una interfaz 190 de vídeo. Además del monitor, los ordenadores también pueden incluir otros dispositivos periféricos de salida tales como altavoces 197 y una impresora 196, que puede estar conectada a través de una interfaz 195 de salida de periféricos.

El ordenador 110 puede operar en un entorno de red utilizando conexiones lógicas a uno o más ordenadores remotos, tales como un ordenador remoto 180. El ordenador remoto 180 puede ser otro ordenador personal, un servidor, un dispositivo de encaminamiento, un PC de red, un dispositivo del mismo nivel u otro nodo común de red, y normalmente incluye muchos de los elementos, o todos ellos, descritos anteriormente con respecto al ordenador personal 110, aunque únicamente se ha ilustrado un dispositivo 181 de almacenamiento de memoria en la Figura 1. Las conexiones lógicas mostradas en la Figura 1 incluyen una red 171 de área local (LAN) y una red 173 de área amplia (WAN), pero también pueden incluir otras redes. Tales entornos de red son comunes en oficinas, redes de ordenadores de empresa, intranets e Internet.

Cuando se utiliza en un entorno de red de LAN, el ordenador personal 110 está conectado a la LAN 171 a través de un adaptador o interfaz 170 de red. Cuando se utiliza en un entorno de red de WAN, el ordenador incluye normalmente un módem 171 u otro medio para establecer una comunicación en la WAN 173, tal como Internet. El módem 172, que puede ser interno o externo, puede estar conectado al bus 121 del sistema por medio de la interfaz 160 de entrada del usuario, u otro mecanismo apropiado. En un entorno en red, los módulos de programas mostrados con respecto al ordenador personal 110, o porciones de los mismos, pueden estar almacenados en el dispositivo remoto de almacenamiento de memoria. A modo de ejemplo, y no de limitación, la Figura 1 ilustra programas remotos 185 de aplicación que residen en el dispositivo 181 de memoria. Se apreciará que las conexiones de red mostradas son ejemplares y se pueden utilizar otros medios para establecer un enlace de comunicaciones entre los ordenadores.

En la siguiente descripción, se describirá la invención con referencia a acciones y representaciones simbólicas de operaciones que son llevadas a cabo por uno o más ordenadores, a no ser que se indique lo contrario. Como tal, se comprenderá que tales acciones y operaciones, de las que a veces se dice que son ejecutadas por un ordenador, incluyen la manipulación por parte de la unidad de procesamiento del ordenador de señales eléctricas que representan datos en una forma estructurada. Esta manipulación transforma los datos o los mantiene en ubicaciones en el sistema de memoria del ordenador, que vuelve a configurar o altera de otra manera la operación del ordenador de una forma bien comprendida por los expertos en la técnica. Las estructuras de datos en las que se mantienen los datos son ubicaciones físicas de la memoria que tienen propiedades particulares definidas por el formato de los datos. Sin embargo, aunque se describe la invención en el anterior contexto, no se pretende que sea limitante como apreciarán los expertos en la técnica que diversas de las acciones y la operación descritas más adelante también puedan ser implementadas en *hardware*.

Como se ha presentado anteriormente, el éxito de un protocolo del mismo nivel (P2P) depende de la capacidad del protocolo para establecer conexiones válidas entre entidades seleccionadas. Debido a que un usuario particular puede conectarse a la red de diversas formas en diversas ubicaciones que tienen distintas direcciones, un enfoque preferente es asignar una identidad única al usuario, y luego convertir esa identidad en una dirección particular a través del protocolo. Tal protocolo de resolución de nombres del mismo nivel (PNRP), en el que la infraestructura de seguridad de la presente invención encuentra aplicabilidad particular, se describe en la solicitud en tramitación como la presente US 2002/0143989 A1, titulada Peer-To-Peer Name Resolution Protocol (PNRP) And Multilevel Cache For Use Therewith, presentada el 29 de agosto de 2001. Sin embargo, un experto en la técnica reconocerá a partir de las siguientes enseñanzas que los procedimientos y la infraestructura de seguridad de la presente invención no están limitados al protocolo particular del mismo nivel de esta solicitud en tramitación como la presente, sino que pueden ser aplicados a otros protocolos de la misma forma.

Como se expone en la solicitud en tramitación como la presente mencionada anteriormente, el protocolo de resolución de nombres del mismo nivel (PNRP) es un protocolo de resolución de nombre a dirección basada en unidades del mismo nivel. Los nombres son números de 256 bits denominados ID PNRP. La dirección consiste en una dirección IPv4 o IPv6, un puerto, y un número de protocolo. Cuando se convierte una ID PNRP en una dirección, se devuelve un certificado de dirección del mismo nivel (PAC). Este certificado incluye la ID PNRP del objetivo, una dirección IP actual, una clave pública, y muchos otros campos. Un ejemplo del protocolo PNRP es denominado nodo. Un nodo puede tener una o más ID PNRP registradas localmente. Un nodo hace que una correlación entre ID y dirección sea descubrible en PNRP por medio de un registro. Cada registro incluye un certificado construido localmente emitido por una unidad del mismo nivel, y requiere una vista apropiada de la memoria intermedia del PNRP. Los anfitriones que no son nodos de PNRP pueden convertir ID PNRP en direcciones IP por medio de una pasarela DNS PNRP. Una pasarela DNS PNRP acepta consultas DNS “A” y “AAAA”, lleva a cabo una búsqueda PNRP para un subconjunto del nombre del anfitrión especificado, y devuelve los resultados como una respuesta a la consulta DNS.

Como se ha indicado anteriormente, el PNRP proporciona un mecanismo basado en unidades del mismo nivel que asocia P2P e ID PNRP con certificados de direcciones del mismo nivel (PAC). Una ID P2P es un identificador persistente de 128 bits. Las ID P2P son creadas al crear una clave calculada de un nombre P2P formateado correctamente. Existen dos tipos de ID P2P, segura y no segura. Una ID P2P segura es una ID con una relación verificable con una clave pública. Una ID P2P es cualquier ID que no sea segura. Se puede publicar una ID P2P dada por medio de muchos nodos distintos. El PNRP utiliza un sufijo de “ubicación del servicio” para garantizar que cada ejemplo publicado tiene una ID PNRP única. Una “ubicación del servicio” es un número de 128 bits que se corresponde a un punto final único del servicio de red. Las ubicaciones del servicio tienen algunos elementos

reconocibles, pero deberían ser consideradas opacas por clientes PNRP. Una ubicación de servicio tiene dos propiedades importantes. En cualquier momento dado, únicamente un *socket* en la nube se corresponde con una ubicación dada de servicio. Cuando se comparan dos ubicaciones de servicio, la longitud del prefijo común para cada una es una medida razonable de proximidad en la red. Dos ubicaciones de servicio que comienzan con los mismo cuatro bits no están más alejadas que dos que comienzan con los mismos tres bits.

Se identifica de forma única a una ID P2P por su concatenación con la ubicación del servicio. Se denomina al identificador resultante de 256 bits (32 bytes) una ID PNRP. Los nodos de PNRP registran una ID PNRP al ejecutar servicios de PNRP con un nombre P2P, autorización, y varios parámetros distintos. Entonces, los servicios PNRP crean y mantienen un certificado de dirección del mismo nivel (PAC) que contiene los datos sometidos. Los PAC incluyen como mínimo una ID PNRP, el intervalo de validez del certificado, la dirección del servicio y de PNRP, la clave pública, y una firma criptográfica generada sobre contenidos selectos del PAC.

La creación y el registro de las ID PNRP solo es una parte del servicio de PNRP. La ejecución del servicio de PNRP puede dividirse en cuatro fases. La primera es el descubrimiento de la nube de PNRP. Durante esta fase un nuevo nodo debe encontrar un nodo existente en la nube a la que desea unirse. La nube puede ser la nube global de PNRP, una nube local al sitio (empresarial) o una nube de enlace local. Una vez se encuentra, se entra en la segunda fase de conexión a una nube de PNRP. Una vez el nodo nuevo ha encontrado un nodo existente, lleva a cabo un procedimiento de SINCRONIZACIÓN para obtener una copia del nivel superior de la memoria intermedia de los nodos existentes. Un único nivel de la memoria intermedia proporciona base suficiente para que un nuevo nodo comience a participar en la nube. Una vez se ha conseguido la SINCRONIZACIÓN, puede iniciarse la siguiente fase, la participación activa en la nube. Después de que se ha completado la inicialización, el nodo puede participar en el registro y en la resolución de la ID PNRP. Durante esta fase, la unidad del mismo nivel también lleva a cabo un mantenimiento regular de la memoria intermedia. Cuando el nodo ha terminado, comienza la cuarta fase, dejando la nube. El nodo da de baja cualquier ID PNRP registrada localmente; luego termina.

El protocolo PNRP consiste en nueve tipos distintos de paquetes, algunos de los cuales han sido presentados anteriormente. Sin embargo, se debe hacer notar que en la presente solicitud los nombres de los paquetes son utilizados simplemente para facilitar una comprensión de su funcionalidad, y no deberían ser considerados limitantes de la forma o el formado del paquete o del propio mensaje. El paquete de CONVERSIÓN pide la resolución de la ID PNRP objetivo en un PAC. Un paquete de RESPUESTA es el resultado de una petición completada de CONVERSIÓN. El paquete de INUNDACIÓN contiene un PAC previsto para la memoria intermedia PNRP del destinatario. Se utiliza un paquete de SOLICITUD para pedir que un nodo PNRP ANUNCIE su nivel superior de la memoria intermedia. El paquete pedido de ANUNCIO contiene una lista de ID PNRP para PAC en el nivel superior de la memoria intermedia de un nodo. Se utiliza un paquete de PETICIÓN para pedir a un nodo que inunde un subconjunto de PAC ANUNCIADO. Se utiliza un paquete de CONSULTA para preguntar de forma no segura a un nodo si una ID PNRP específica está registrada en ese nodo. Para confirmar el registro local de una ID PNRP, se utiliza un paquete de AUTORIZACIÓN. Este paquete proporciona, opcionalmente, una cadena de certificación para ayudar a validar el PAC para esa ID. Un paquete ACK da acuse de la recepción y/o el procesamiento con éxito de ciertos mensajes. Finalmente, se utiliza el paquete de REPARACIÓN para intentar unir nubes que pueden estar divididas.

Una vez se ha inicializado completamente un nodo, puede participar en la nube de PNRP al llevar a cabo cinco tipos de actividades. En primer lugar, un nodo puede dar de alta y de baja una ID PNRP. Cuando se registra una ID PNRP, el servicio de PNRP crea un certificado de dirección del mismo nivel (PAC) que asocia la ID PNRP, el puerto y el protocolo de la dirección del servicio, el puerto y el protocolo de dirección de PNRP, y una clave pública. Se introduce este PAC en la memoria intermedia local, y se inicia una CONVERSIÓN utilizando el nuevo PAC como la fuente, e [ID PNRP + 1] como el objetivo. Esta CONVERSIÓN es procesada por un número de nodos con ID PNRP muy similares a la ID registrada. Cada destinatario de la CONVERSIÓN añade el PAC del nuevo nodo a su memoria intermedia, anunciando de ese modo la nueva ID PNRP en la nube. Cuando se da de baja una ID PNRP, se crea un PAC actualizado con una bandera de “revocación” puesta. Se inunda el PAC actualizado a todas las entradas en el nivel más bajo de la memoria intermedia local. Cada destinatario de la INUNDACIÓN comprueba su memoria intermedia en busca de una versión anterior del PAC. Si se encuentra, el destinatario elimina el PAC de su memoria intermedia. Si se elimina el PAC del nivel más bajo de la memoria intermedia, el destinatario INUNDA, a su vez, la revocación a los nodos de PNRP representados por todos los otros PAC en el nivel más bajo de la memoria intermedia.

El nodo de PNRP también puede participar en la resolución de la ID PNRP. Como se ha expuesto en la solicitud mencionada anteriormente, las ID PNRP son convertidas en PAC al encaminar mensajes de CONVERSIÓN sucesivamente más cercanos a la ID PNRP objetivo. Cuando un nodo recibe una CONVERSIÓN, puede rechazar la CONVERSIÓN, devolviéndola al salto anterior con una RESPUESTA, o puede remitir la CONVERSIÓN a un nodo cuya ID PNRP es más cercana a la ID objetivo que la del propio nodo. El nodo también recibe y remite paquetes de RESPUESTA como parte de la resolución. El nodo de PNRP también puede iniciar CONVERSIONES en nombre de un cliente local. El servicio de PNRP proporciona una API para permitir peticiones asíncronas de resolución. El nodo local origina paquetes de CONVERSIÓN, y finalmente recibe una RESPUESTA correspondiente.

El nodo de PNRP también atiende peticiones de sincronización de las memorias intermedias. Tras recibir un paquete de SOLICITUD, el nodo responde con un paquete de ANUNCIO, enumerando las ID PNRP en su nivel más alto de la memoria intermedia. Entonces, el nodo solicitante envía una PETICIÓN enumerando las ID PNRP para cualquier PAC ANUNCIADO que desee. Entonces, se INUNDA cada entrada PEDIDA de memoria intermedia al PETICIONARIO. Finalmente, y como se expondrá con más detalle a continuación, el PNRP también lleva a cabo una validación de identidad. La validación de identidad es un dispositivo de mitigación de amenazas utilizado para validar PAC. La validación de identidad tiene básicamente dos propósitos. En primer lugar, la validación de identidad garantiza que el nodo de PNRP especificado en un PAC tiene la ID PNRP de ese PAC registrado localmente. En segundo lugar, para ID PNRP seguras (expuestas a continuación), la validación de identidad garantiza que el PAC fue firmado utilizando una clave con una relación probable criptográficamente con la autoridad en la ID PNRP.

Habiendo proporcionado un conocimiento operativo del sistema de PNRP para el que una realización de la infraestructura de seguridad de la presente invención encuentra particular relevancia, se presta atención ahora a los mecanismos de seguridad proporcionados por la infraestructura de seguridad de la presente invención. Estos mecanismos son proporcionados por el sistema de la presente invención para eliminar, o como mínimo mitigar, el efecto de los diversos ataques que pueden ser planteados por un nodo malicioso en una nube de P2P como se ha expuesto anteriormente. El protocolo PNRP no tiene ningún mecanismo para evitar estos ataques, ni tampoco hay una única solución para abordar todas estas amenazas. Sin embargo, la infraestructura de seguridad de la presente invención minimiza la alteración que puede ser causada por un nodo malicioso, y puede ser incorporada en el protocolo PNRP.

Como muchos protocolos de P2P que tienen éxito, las entidades pueden ser publicadas para un descubrimiento sencillo. Sin embargo, para proporcionar seguridad e integridad al protocolo de P2P cada identidad incluye, preferentemente, un certificado adjunto de identidad. Sin embargo, una arquitectura robusta de seguridad podrá gestionar entidades tanto seguras como no seguras. Según una realización de la presente invención, se proporciona esta robustez mediante el uso de PAC autoverificables.

Se crea un PAC seguro autoverificable al proporcionar una correlación entre la ID y una clave pública. Esto evitará que cualquiera publique un PAC seguro sin tener la clave privada para firmar ese PAC y, por lo tanto, evitará un gran número de ataques de robo de identidad. El guardián de la clave privada de la ID utiliza el certificado para adjuntar información adicional a la ID, tal como la dirección IP, un nombre fácil de recordar, etc. Preferentemente, cada nodo genera su propio par de claves privada-pública, aunque las tales pueden ser proporcionadas por un proveedor de confianza. Entonces, la clave pública se incluye como parte del identificador del nodo. Únicamente el nodo que creó el par de claves tiene la clave privada con la que puede probar que es el creador de la identidad del nodo. De esta forma, se puede descubrir el robo de identidad y, por lo tanto, puede ser impedido.

Se puede representar un formato genérico para tales certificados como [Versión, ID, <Info relacionada con la ID>, Validez, Algoritmos, P_{Emisor}]K_{Emisor}. De hecho, el nombre P2P/URL es parte del formato básico del certificado, con independencia de si es una ID segura o no segura. Según se utiliza en esta representación de certificado, Versión es la versión del certificado, ID es el identificador que va a ser publicado, <Info relacionada con la ID> representa información que ha de estar asociada con la ID, Validez representa el periodo de validez expresado en un par de fechas Desde-Hasta expresadas como tiempo universal coordinado (también conocido como GMT), Algoritmos hace referencia a los algoritmos utilizados para generar los pares de claves, y para firmar, y P_{Emisor} es la clave pública del emisor del certificado. Si el emisor del certificado es el mismo que el dueño de la ID entonces esta P_{ID} es la clave pública del dueño de la ID. El término K_{Emisor} es la clave privada correspondiente a la P_{Emisor}. Si el emisor del certificado es el dueño de la ID entonces esta K_{ID} es la clave privada del dueño de la ID.

En una realización preferente, la <info relacionada con la ID> comprende la tupla de direcciones en la que se puede encontrar esta ID, y la tupla de direcciones para el servicio de PNRP del emisor. En esta realización, el certificado de dirección se convierte en [Versión, ID, <Dirección>$_{ID}$, <Dirección>$_{PNRP}$, Validez, Bandera de revocación, Algoritmos, P_{Emisor}]K_{Emisor}. En esta representación expandida, la ID es el identificador que va a ser publicado, que puede ser una ID del grupo o una ID de la unidad del mismo nivel. La <Dirección> es la tupla de la dirección IPv6, del puerto, y del protocolo. <Dirección>$_{ID}$ es la tupla de direcciones que ha de asociarse con la ID. <Dirección>$_{PNRP}$ es la tupla de direcciones del servicio de PNRP (u otro servicio de P2P) en la máquina emisora. Esta es preferentemente la dirección de la dirección de PNRP del emisor. Será utilizada por los otros nodos de PNRP para verificar la validez del certificado. La Validez es el periodo de validez expresado en un par de fechas Desde-Hasta. La Bandera de revocación, cuando está puesta, marca un certificado de revocación. La P_{Emisor} es la clave pública del emisor del certificado, y la K_{Emisor} es la clave privada correspondiente a la P_{Emisor}. Si el emisor del certificado es el dueño de la ID, entonces esta K_{ID} es la clave privada de la ID.

En una realización preferente de la presente invención, se tienen que cumplir las siguientes condiciones para que un certificado sea válido. La firma del certificado debe ser válida, y el certificado no puede haber caducado. Es decir, la fecha actual expresada como UDT debe encontrarse en el intervalo especificado por el campo de Validez. La clave calculada de la clave pública también debe corresponderse con la ID. Si el Emisor es el mismo que el dueño de la ID, entonces tiene que verificarse la clave calculada de la clave pública del emisor en la ID. Si la P_{Emisor} es distinta de la P_{ID} entonces tiene que haber una cadena de certificados que lleva a un certificado firmado con K_{ID}. Tal cadena

verifica la relación entre el emisor y el dueño de la ID. Además, en el caso en el que se publique una lista de revocación de certificados (CRL) para esa clase de ID y la CRL sea accesible, entonces el autentificado puede verificar que ninguno de los certificados en la cadena aparecen en la CRL.

La infraestructura de seguridad de la presente invención también gestiona PAC no seguros. Según la presente invención, se crea un PAC no seguro autoverificable al incluir el identificador de recursos uniforme (URI) del que se deriva la ID. De hecho, las ID tanto seguras como no seguras incluyen el URI en el PAC. El URI tiene el formato "p2p://URI". Esto evitará que un nodo malicioso publique la ID segura de otro nodo en un PAC no seguro.

La infraestructura de la presente invención también permite el uso de ID no seguras. El problema con tales ID no seguras es que son muy fáciles de falsificar. Un nodo malicioso puede publicar una ID no segura de cualquier otro nodo. Las ID no seguras también abren agujeros de seguridad en los que se vuelve posible dificultar el descubrimiento de un nodo bueno. Sin embargo, al incluir un URI según la presente invención, las ID no seguras no pueden afectar a las ID seguras de ninguna forma. Además, la infraestructura de la presente invención requiere que los PAC que contienen ID no seguras tengan el mismo formato que los PAC seguros, es decir, contengan una clave pública y claves privadas. Al imponer la misma estructura a los PAC no seguros que a los PAC seguros, no se reduce el listón para la generación de PAC. Además, al incluir un URI en el PAC no es factible generar por cálculo un URI que correlaciona a una ID segura específica.

Un problema que surge es cuándo deberían ser verificados los PAC, reconociendo un compromiso entre una mayor seguridad de la nube de P2P y una mayor sobrecarga. Sin embargo, el PAC contenido en los diversos paquetes expuestos anteriormente tiene que ser verificado en algún momento. Esta verificación del PAC incluye comprobar si la firma de la ID es válida o no y comprobar si la ID se corresponde con la clave pública para las ID seguras. Para equilibrar la sobrecarga con respecto a los problemas de seguridad, una realización de la presente invención verifica los PAC antes de que se lleve a cabo ningún procesamiento de ese paquete. Esto garantiza que nunca se procesan datos inválidos. Sin embargo, reconociendo que la verificación del PAC puede ralentizar el procesamiento de paquetes, lo que puede no ser adecuado para ciertas clases de paquetes, por ejemplo paquetes de CONVERSIÓN, una realización alterna de la presente invención no verifica el PAC en estos paquetes.

Además de la verificación del PAC, la infraestructura de seguridad de la presente invención también lleva a cabo una comprobación de la propiedad de la ID para validar el PAC. Como se ha expuesto anteriormente, se puede descubrir el robo de identidad utilizando esa dirección en PNRP y otros protocolos de P2P. Esta validación puede conllevar verificar simplemente que la ID es la clave calculada de la clave pública incluida en el certificado. La validación de la propiedad también puede conllevar la emisión de un paquete de CONSULTA a la dirección en ese PAC. El paquete de CONSULTA contendrá la ID que debe ser verificada, y una ID de transacción. Si la ID está presente en esa dirección, el nodo debería dar acuse esa CONSULTA. Si la ID no está presente en esa dirección, el nodo no debería dar acuse esa CONSULTA. Si se requiere la cadena de certificación para verificar la identidad, el nodo devuelve la cadena completa de certificación. Aunque una validación de firma y de ID -> URL sigue siendo compleja y un uso significativo de recursos, al igual que lo es la validación de la cadena de trust en una cadena suministrada de certificación, el sistema de la presente invención evita cualquier tipo de protocolo de interrogación/respuesta, lo que añadiría un nivel adicional de complejidad a la validación de los PAC. Además, la inclusión de la ID de transacción evita que el nodo malicioso genere previamente la respuesta a las CONSULTAS. Además, este mecanismo prescinde del requerimiento de que el PAC tenga que portar la cadena completa de certificación.

También se facilita la comprobación de la propiedad de la ID en el sistema de la presente invención al modificar el paquete estándar de CONVERSIÓN, de forma que también pueda llevar a cabo la comprobación de la propiedad de la ID. Este paquete modificado de CONVERSIÓN contiene la ID de la dirección a la que se remite la CONVERSIÓN. Si la ID se encuentra en esa dirección enviará un ACK, de lo contrario enviará un NACK. Si la ID no procesa la CONVERSIÓN o si se recibe un NACK, se elimina la ID de la memoria intermedia. De esta forma, se valida un PAC sin recurrir a ningún tipo de protocolo de interrogación/respuesta y sin enviar ningún paquete especial de CONSULTA, esencialmente, al superponer un mensaje de CONSULTA con la CONVERSIÓN. Este procedimiento de superposición será expuesto de nuevo a continuación con respecto a la FIG. 2. Este procedimiento hace que sea sencillo borrar PAC inválidos o caducados.

Esta comprobación de validación de la identidad se produce en dos momentos distintos. El primero es cuando un nodo va a añadir un PAC a uno de sus dos niveles más bajos de la memoria intermedia. La validez del PAC en los dos niveles más bajos de la memoria intermedia es crítica para la capacidad del PNRP para convertir ID PNRP. Llevar a cabo la validación de identidad antes de añadir un PAC a cualquiera de estos dos niveles mitiga varios ataques. La propiedad de la ID no se lleva a cabo si se va a añadir el PAC a cualquier nivel superior de la memoria intermedia debido a la renovación en estos niveles superiores. Se ha determinado que casi el 85% de todas las entradas de PAC en los niveles más altos de la memoria intermedia es sustituido o caduca antes de que jamás sea utilizado. Como tal, la probabilidad de ver algún efecto por tener un PAC inválido en estos niveles superiores es lo suficientemente baja como para justificar llevar a cabo la validación de la ID cuando son introducidos.

Cuando se determina que una entrada pertenecería en uno de los dos niveles más bajos de la memoria intermedia, se coloca el PAC en una lista de reserva hasta que se pueda validar su identidad. Este primer tipo de validación de

identidad utiliza el mensaje de CONSULTA. Tal validación de identidad confirma que un PAC sigue siendo válido (registrado) en su nodo de origen, y pide información para ayudar a validar la autorización del nodo de origen para publicar ese PAC. Se define una bandera en el mensaje de CONSULTA para el campo de “banderas”, es decir CADENA_ENVÍO_RF, que pide que el receptor envíe una cadena de certificación (si existe) en una respuesta de AUTORIZACIÓN. Si el receptor de la CONSULTA no tiene la autorización para publicar el PAC o si el PAC ya no está registrado localmente, el receptor simplemente desecha el mensaje de CONSULTA. Dado que el nodo local no recibe una respuesta apropiada por medio de un mensaje de AUTORIZACIÓN, el PAC malo nunca será introducido en su memoria intermedia y, por lo tanto, no puede tener un efecto malicioso sobre su operación en la nube de P2P.

Si el receptor de la CONSULTA tiene la autorización para emitir el PAC y si sigue estando registrado localmente, ese nodo responderá 200 al mensaje de CONSULTA con un mensaje de AUTORIZACIÓN, como se ilustra en la FIG. 2. Aunque no se ilustra en la FIG. 2, el nodo receptor en una realización de la presente invención comprueba si el mensaje de AUTORIZACIÓN dice que la ID sigue registrada en el nodo que envió la AUTORIZACIÓN. Una vez determina 202 el nodo local que este mensaje de AUTORIZACIÓN es en respuesta al mensaje de CONSULTA, elimina el PAC de la lista 204 de reserva. Si fue pedida 206 la cadena de certificación, se comprueba el mensaje de AUTORIZACIÓN para ver si la cadena de certificación está presente y es válida 208. Si la cadena de certificación está presente y es válida, entonces se añade el PAC a la memoria intermedia y se marca como válido 210. De lo contrario, se borra el PAC 212. Si no fue pedida 206 la cadena de certificación, entonces se añade simplemente el PAC a la memoria intermedia y se marca como válido 210.

Como será evidente ahora, se utiliza este mensaje de AUTORIZACIÓN para confirmar o negar que una ID PNRP sigue estando registrada en el nodo local, y proporciona opcionalmente una cadena de certificación para permitir que el destinatario de la AUTORIZACIÓN valide el derecho del nodo a publicar el PAC correspondiente a la ID objetivo. Además del mensaje de CONSULTA, el mensaje de AUTORIZACIÓN puede ser una respuesta apropiada a un mensaje de CONVERSIÓN, como se expondrá a continuación. El mensaje de AUTORIZACIÓN incluye diversas banderas que pueden estar puestas por el nodo receptor para indicar una respuesta negativa. Una bandera tal es la bandera RECHAZAR_DEMASIADO_OCUPADA_AF, que solo es válida en respuesta a una CONVERSIÓN. Esta bandera indica que el anfitrión está demasiado ocupado para aceptar una CONVERSIÓN, y dice al remitente que debería remitir la CONVERSIÓN a otro lugar para ser procesada. Aunque no ayuda en la validación de identidad, es otro mecanismo de seguridad de la presente invención para evitar un ataque de DoS, como se expondrá con más detalle a continuación. La bandera de FUENTE_INVÁLIDA_AF, que solo es válida en respuesta a una CONVERSIÓN, indica que el PAC Fuente en la CONVERSIÓN es inválido. La bandera de MEJOR_CORRESPONDENCIA_INVÁLIDA_AF, que también solo es válida en respuesta a una CONVERSIÓN, indica que el PAC de “mejor correspondencia” en la CONVERSIÓN es inválido. La bandera de ID_DESCONOCIDA_AF indica que la ID PNRP “validar” especificada no está dada de alta con este anfitrión. Otras banderas en el mensaje de AUTORIZACIÓN indican al nodo receptor que la información pedida está incluida. La bandera de CADENA_CERT_AF indica que se incluye una cadena de certificación que permitirá la validación de la relación entre la ID PNRP “validar” y la clave pública utilizada para firmar su PAC. Solo se envía el mensaje de AUTORIZACIÓN como un acuse/una respuesta bien para el mensaje de CONSULTA o bien para el mensaje de CONVERSIÓN. Si se recibe en cualquier momento una AUTORIZACIÓN fuera de este contexto, esta es descartada.

La segunda vez que se lleva a cabo la validación de la identidad es de forma oportuna durante el procedimiento de CONVERSIÓN. Como se ha expuesto, las memorias intermedias de PNRP tienen una tasa elevada de renovación. Por consiguiente, la mayor parte de las entradas de la memoria intermedia son sobreescritas en la memoria intermedia antes de que jamás sean utilizadas. Por lo tanto, la infraestructura de seguridad de la presente invención no valida estos PAC hasta que son utilizados realmente, y a no ser que lo sean. Cuando se utiliza un PAC para encaminar una vía de CONVERSIÓN, el sistema de la presente invención superpone la validación de identidad encima del paquete de CONVERSIÓN, como se ha presentado anteriormente. La CONVERSIÓN contiene una ID de “siguiente salto” que es tratada igual que la “ID objetivo” en un paquete de CONSULTA. Entonces, se da acuse de esta CONVERSIÓN con un paquete de AUTORIZACIÓN, al igual que se espera de una CONSULTA expuesto anteriormente. Si falla una oportuna validación de la identidad, el receptor de la CONVERSIÓN no es quien el remitente cree que es. Por consiguiente, se encamina la CONVERSIÓN a otro lugar y se elimina el PAC inválido de la memoria intermedia.

Este procedimiento también se ilustra en la FIG. 2. Cuando un nodo P PNRP recibe un paquete 200 de AUTORIZACIÓN con la cabecera del campo Tipo de Mensaje configurada para la CONVERSIÓN 202, el nodo receptor analiza las banderas de AUTORIZACIÓN para determinar si esta bandera de AUTORIZACIÓN es negativa 214, como se ha expuesto anteriormente. Si cualquiera de las banderas de respuesta negativa están puestas en el mensaje de AUTORIZACIÓN, se borra 216 de la memoria intermedia y se encamina la CONVERSIÓN a otro lugar. La dirección a la que fue enviada la CONVERSIÓN está adjunta a la vía de CONVERSIÓN y marcada RECHAZADA. Entonces, se remite la CONVERSIÓN a un nuevo destino. Si la AUTORIZACIÓN no es negativa y si se solicitó 218 la cadena de certificación, se comprueba la bandera CADENA_CERT_AF del mensaje de AUTORIZACIÓN para ver si está presente la cadena de certificación. Si está presente el nodo receptor debería llevar a cabo una operación de validación de cadena en el PAC metida en memoria intermedia para la ID PNRP especificada en validar. Se debería comprobar la cadena para garantizar que todos los certificados en la misma son válidos, y la relación entre la raíz y la hoja de la cadena es válida. La clave calculada de la clave pública para la raíz

de la cadena debería, como mínimo, ser comparada con la clave utilizada para firmar el PAC para garantizar que coinciden. Si falla cualquiera de estas comprobaciones o si no está presente la cadena de certificación cuando es pedida 220, el PAC debería ser eliminado de la memoria intermedia 222 y se debería volver a procesar la CONVERSIÓN. Si se incluye la cadena de certificación pedida y es validada 220, el PAC correspondiente a la ID PNRP validar debería ser marcado como completamente validado 224. Si se desea, se pueden retener la ID PNRP, la dirección del servicio de PNRP, y los tiempos de validación del PAC y se puede borrar el propio PAC de la memoria intermedia para ahorrar memoria.

Como ejemplo de esta validación de identidad, asume que P es un nodo que pide una validación de identidad de la ID PNRP “T”. N es el nodo que recibe la petición de validación de identidad. Esto podría ocurrir como resultado de que P recibe bien un paquete de CONSULTA con una ID objetivo = T, o bien un paquete de CONVERSIÓN con el siguiente salto = T. N comprueba su lista de ID PNRP registradas localmente. Si T no está en esa lista, entonces se comprueba el tipo recibido de paquete. Si era una CONSULTA, N desecha discretamente la petición de CONSULTA. Después de que caduquen los intentos normales de retransmisión, P descartará el PAC como inválido y se realiza un procesamiento. Si era una CONVERSIÓN, N responde con un paquete de AUTORIZACIÓN que indica que la ID T no está registrada localmente. Entonces, P envía la CONVERSIÓN a otro lugar. Si T está en la lista de ID PNRP en N, N construye un paquete de AUTORIZACIÓN y hace la ID objetivo igual a T. Si T es una ID no segura, entonces N envía el paquete de AUTORIZACIÓN a P. Si T es una ID segura, y la autorización para la ID segura es la clave utilizada para firmar el PAC, entonces N envía el paquete de AUTORIZACIÓN a P. Si ninguno de estos es verdad y si está puesta la bandera de CADENA_ENVÍO_RF, entonces N recupera la cadena de certificación relacionada con la clave utilizada para firmar el PAC para la autoridad para la ID PNRP T. Se inserta la cadena de certificación en el paquete de AUTORIZACIÓN, y luego N envía el paquete de AUTORIZACIÓN a P. En este momento, si T es una ID no segura se completa el procesamiento. De lo contrario, P valida la relación entre la clave de firma del PAC y la autorización utilizada para generar la ID PNRP T. Si falla la validación, se descarta el PAC. Si falla la validación y el mensaje iniciador fue una CONVERSIÓN, P remite la CONVERSIÓN a otro lugar.

Como será evidente ahora por estas dos veces que se lleva a cabo la verificación de la propiedad de la identidad, bien por medio del paquete de CONSULTA o bien por medio del paquete modificado de CONVERSIÓN, no se puede poblar un PAC inválido en toda la nube de P2P utilizando una INUNDACIÓN, y las búsquedas no serán remitidas a ID inexistentes o inválidas. La validación del PAC es necesaria para la INUNDACIÓN porque, si se permite que el paquete de INUNDACIÓN se propague en la red sin ninguna validación, entonces puede causar un ataque de DoS. Por medio de estos mecanismos, un nodo popular no será inundado con comprobaciones de propiedad de la ID porque su ID pertenecerá únicamente a los dos niveles más bajos de la memoria intermedia de muy pocos nodos.

Como se describe con más detalle en la solicitud en tramitación como la presente a la que se ha hecho referencia anteriormente, un nodo PNRP N se entera de una ID nueva de una de cuatro maneras. Puede enterarse de una nueva ID mediante la inundación inicial de la memoria intermedia de un vecino. Específicamente, cuando se presenta un nodo de P2P, contacta con otro nodo miembro de la nube de P2P e inicia una secuencia de sincronización de memorias intermedias. También puede enterarse de una nueva ID como resultado de que un vecino inunde un nuevo registro de su memoria intermedia más baja. Por ejemplo, asume que el nodo N aparece como una entrada en el nivel más bajo de la memoria intermedia del nodo M. Cuando M se entera de una nueva ID, si la ID cabe en su nivel más bajo de la memoria intermedia, inundará con ella a las otras entradas en ese nivel de la memoria intermedia, respectivamente a N. Un nodo también puede enterarse de una nueva ID como resultado de una petición de búsqueda. El originador de una petición de búsqueda inserta su certificado de dirección en la petición, y el PAC para la “mejor correspondencia” hasta el momento para la petición de búsqueda también inserta su PAC en la petición. De esta forma, todos los nodos a lo largo de la vía de petición de búsqueda actualizarán su memoria intermedia con la dirección del originador de la búsqueda, y la dirección de la mejor correspondencia. De forma similar, un nodo puede enterarse de una nueva ID como resultado de una respuesta de búsqueda. El resultado de una petición de búsqueda recorre un subconjunto de la vía de petición en orden inverso. Los nodos a lo largo de esta vía actualizan su memoria intermedia con el resultado de la búsqueda.

Según el PNRP, cuando se presenta el nodo por primera vez, descubre un vecino. Sin embargo, como se ha expuesto anteriormente, si el nodo que se descubre en primer lugar es un nodo malicioso, el nuevo nodo puede ser controlado por el nodo malicioso. Para evitar o minimizar la posibilidad de tal incidencia, la infraestructura de seguridad de la presente invención proporciona dos mecanismos para garantizar un arranque seguro del nodo. El primero es un descubrimiento aleatorizado. Cuando un nodo intenta descubrir otro nodo que le permitirá unirse a la nube de PNRP, la última opción para el descubrimiento es utilizar una multidifusión/radiodifusión porque es el procedimiento más inseguro de descubrimiento de PNRP. Debido a la naturaleza del descubrimiento es muy difícil distinguir entre un nodo bueno y uno malo. Por lo tanto, cuando se requiere este procedimiento de multidifusión/radiodifusión, la infraestructura de seguridad de la presente invención hace que el nodo seleccione de forma aleatoria uno de los nodos que respondieron al mensaje radiodifundido de descubrimiento (MARCOPOLO o un protocolo existente de descubrimiento de multidifusión, por ejemplo SSDP). Al seleccionar un nodo aleatorio, el sistema de la presente invención minimiza la probabilidad de seleccionar un nodo malo. El sistema de la presente invención también lleva a cabo este descubrimiento sin utilizar ninguna de sus ID. Al no utilizar ID durante el

descubrimiento, el sistema de la presente invención evita que el nodo malicioso que tiene como objetivo una ID específica.

Se proporciona un segundo mecanismo de arranque de nodo seguro por medio de una fase modificada de sincronización durante la cual el nodo mantendrá un vector de bits. Este mecanismo de fase modificada de sincronización puede ser comprendido mejor por medio de un ejemplo ilustrado en el diagrama simplificado de flujo de la FIG. 3. Supongamos que Alicia 226 envía una SOLICITUD 228 a Roberto 230 con su PAC en la misma. Si el PAC de Alicia no es válido 232, Roberto 230 simplemente desecha la SOLICITUD 234. Si el PAC es válido, Roberto 230 mantendrá entonces un vector de bits para almacenar el estado de esta conexión. Cuando se recibe esta SOLICITUD, Roberto 230 genera 236 un valor de uso único y crea con él una clave calculada con la ID PNRP de Alicia. Se utilizará el número resultante como un índice en este vector de bits que Roberto establecerá. Entonces, Roberto 230 responde 238 a Alicia 226 con un mensaje de ANUNCIO. Este ANUNCIO contendrá el PAC de Roberto y un valor de uso único codificado con la clave pública de Alicia, además de otra información, y estará firmado por Roberto 230. Cuando Alicia 226 recibe este ANUNCIO, verifica 240 la firma y el PAC de Roberto. Si no puede ser verificado, es desechado 241. Si puede ser verificado, Alicia 226 decodifica 242 entonces el valor de uso único. Entonces, Alicia 226 generará 244 una PETICIÓN que contendrá este valor de uso único y la ID PNRP de Alicia. Roberto 230 procesará 246 esta PETICIÓN al crear la clave calculada de la ID PNRP de Alicia con el valor de uso único enviado en el paquete de PETICIÓN. Si en 248 el bit está puesto en el vector de bits que tiene los resultados de clave calculada como un índice, entonces Roberto pondrá a cero los bits y comenzará a procesar la PETICIÓN 250. De lo contrario, Roberto ignorará la PETICIÓN 252 dado que puede ser un ataque de reproducción.

Esto hace que el arranque del nodo sea un procedimiento seguro dado que la secuencia no puede ser reproducida. Requiere una sobrecarga mínima en términos de recursos consumidos, incluyendo CPU, puertos de red, y tráfico de red. No se requieren temporizadores que deban ser mantenidos para la información de estado, y solo se enviarán datos a la ID que inició la sincronización. De hecho, esta fase modificada de sincronización es asíncrona, lo que permite que un nodo procese de forma simultánea múltiples SOLICITUDES.

Muchas de las amenazas expuestas anteriormente pueden ser minimizadas al controlar la tasa a la que se procesan los paquetes, es decir, limitar el consumo de recursos del nodo. La idea tras esto es que un nodo no debería consumir el 100% de su CPU intentando procesar los paquetes de PNRP. Por lo tanto, según una realización de la presente invención un nodo puede rechazar el procesamiento de ciertos mensajes cuando detecta que tal procesamiento entorpecerá su capacidad de funcionar de forma eficaz.

Un mensaje tal que el nodo puede decidir no procesar es el mensaje de CONVERSIÓN recibido de otro nodo. Se ilustra este procedimiento de forma simplificada en la FIG. 4. Una vez se recibe 254 un mensaje de CONVERSIÓN, el nodo hará una comprobación 256 para ver si está operando en ese momento con una capacidad de la CPU mayor que un límite predeterminado. Si su CPU está demasiado ocupada para procesar el mensaje de CONVERSIÓN, enviará 258 un mensaje de AUTORIZACIÓN con la bandera de RECHAZAR_DEMASIADO_OCUPADA_AF puesta indicando su incapacidad de procesar la petición porque está demasiado ocupada. Si la CPU no está demasiado ocupada 256, el nodo determinará 260 si todos los PAC en el mensaje de CONVERSIÓN son válidos, y rechazará 262 el mensaje si se encuentra que cualquiera de ellos es inválido. Si todos los PAC son válidos 260, el nodo procesará 264 la CONVERSIÓN.

Si el nodo puede responder 266 a la CONVERSIÓN, el nodo convertirá 268 la CONVERSIÓN en una RESPUESTA y la enviará al nodo del que se recibió la CONVERSIÓN. Sin embargo, si la ID objetivo no está registrada localmente, el nodo calculará 270 la posición del bit como la clave calculada de los campos en la CONVERSIÓN y pondrá la posición del bit correspondiente en el vector de bits. Como se ha expuesto brevemente anteriormente, se utiliza este vector de bits como un mecanismo de seguridad para evitar el procesamiento de mensajes erróneos de respuesta cuando el nodo no ha enviado ningún mensaje para el se espera una respuesta. El nodo encuentra el siguiente salto al que remitir la CONVERSIÓN, con las modificaciones apropiadas para dar evidencias de su procesamiento del mensaje. Si en 272 el nodo al que se va a remitir la CONVERSIÓN ya ha sido verificado, el nodo simplemente remite 276 la CONVERSIÓN a este siguiente salto. Si en 272 no se ha verificado aún este siguiente salto seleccionado, el nodo superpone 274 una petición de propiedad de la ID en la CONVERSIÓN y la remite 276 al nodo. En respuesta a la petición superpuesta de propiedad de la ID, el nodo esperará recibir un mensaje de AUTORIZACIÓN como se ha expuesto anteriormente, el procedimiento para lo cual se ilustra en la FIG. 2. Como se ilustra en la FIG. 2, si no se recibe una AUTORIZACIÓN de validación en la etapa 214, se borra 216 el PAC del nodo al que fue remitida la CONVERSIÓN de la memoria intermedia y se vuelve a procesar la CONVERSIÓN desde la etapa 254 de la FIG. 4.

Otro mensaje que el nodo puede decidir no procesar porque su CPU está demasiado ocupada es el mensaje de INUNDACIÓN. En este procedimiento, ilustrado de forma simplificada en la FIG. 5, si en 278 la nueva información presente en la INUNDACIÓN va a cualquiera de los dos niveles más bajos de la memoria intermedia, se comprueba el PAC para determinar si es válido 280. Si el PAC no es válido, se rechaza 284 la INUNDACIÓN. Sin embargo, si el PAC es válido 280, es colocado en una lista 282 de reserva. Las entradas en la lista de reserva son tomadas a intervalos aleatorios y son procesadas cuando la CPU no está demasiado ocupada. Dado que estas entradas van a ser introducidas en los dos niveles más bajos de la memoria intermedia, se llevan a cabo tanto la verificación de ID

como la validación de la propiedad como se ha expuesto anteriormente. Si en 278 la nueva información presente en la INUNDACIÓN va a los niveles superiores de la memoria intermedia y la CPU está demasiado ocupada para procesarlos 286, entonces son descartados 288. Si el nodo tiene capacidad disponible 286 de procesamiento de CPU, se comprueba el PAC para determinar si es válido 290. Si lo es, se añade el PAC a la memoria intermedia 292, de lo contrario se rechaza 294 la INUNDACIÓN.

El arranque (SINCRONIZACIÓN) del nodo es otro procedimiento que consume una cantidad considerable de recursos en un nodo, incluyendo no solo la capacidad de procesamiento de la CPU sino también ancho de banda de la red. Sin embargo, se requiere el procedimiento de sincronización para permitir que un nuevo nodo participe completamente en la nube de P2P. Como tal, el nodo responderá a la petición de otro nodo para el arranque si tiene suficientes recursos disponibles en el momento dado. Es decir, al igual que con los dos mensajes que se acaban de exponer, el nodo puede rechazar participar en el arranque si su uso de la CPU es demasiado elevado. Sin embargo, dado que este procedimiento consume tanta capacidad, un nodo malicioso sigue pudiendo sacar partido de esto al lanzar un gran número de secuencias tales. Como tal, una realización de la infraestructura de seguridad de la presente invención limita el número de sincronizaciones del nodo que pueden ser llevadas a cabo por un nodo dado para evitar este ataque. Además, esta limitación puede estar limitada en el tiempo, de forma que un nodo malicioso no pueda inhabilitar un nodo para que nunca lleve a cabo tal sincronización de nuevo en el futuro.

También se han expuesto anteriormente muchos ataques basados en búsquedas que podrían ser lanzadas o causadas por un nodo malicioso. Para eliminar o minimizar el efecto de tales ataques basados en búsquedas, el sistema de la presente invención proporciona dos mecanismos. El primero es la aleatorización. Es decir, cuando un nodo está buscando un siguiente salto apropiado al que remitir una petición de búsqueda (CONVERSIÓN), identifica un número de posibles nodos candidatos y luego selecciona de forma aleatoria una ID de esas ID candidatas a la que remitir la CONVERSIÓN. En una realización, se identifican tres nodos candidatos para la selección aleatoria. Las ID pueden estar seleccionadas en base a una probabilidad ponderada como una alternativa a una aleatorización total. Un procedimiento tal para el cálculo de una probabilidad ponderada de que la ID pertenece a un nodo no malicioso está basado en la distancia de la ID PNRP desde la ID objetivo. Entonces, se determina la probabilidad como una proporcionalidad inversa a la distancia ID entre ese nodo y el nodo objetivo. En cualquier caso, esta aleatorización reducirá la probabilidad de enviar la petición de CONVERSIÓN a un nodo malicioso.

El segundo mecanismo de seguridad que es eficaz contra ataques basados en búsquedas utiliza el vector de bits expuesto anteriormente para mantener la información de estado. Es decir, un nodo mantiene la información identificando todos los mensajes de CONVERSIÓN que ha procesado para los cuales aún no se ha recibido una respuesta. Los campos utilizados para mantener la información de estado son la ID objetivo y la lista de direcciones en el paquete de CONVERSIÓN. Se utiliza este segundo campo para garantizar que la lista de direcciones no ha sido modificada por un nodo malicioso en un intento por interrumpir la búsqueda. Como se ha expuesto anteriormente con otros casos de uso del vector de bits, el nodo genera una clave calculada de estos campos desde la CONVERSIÓN y pone la posición del bit correspondiente en el vector de bits para mantener un historial del procesamiento de esa CONVERSIÓN.

Como se ilustra en el diagrama simplificado de flujo de la FIG. 6, cuando se recibe 296 un mensaje de RESPUESTA de otro nodo, se crea 298 con los campos en este mensaje de RESPUESTA una clave calculada para calcular la posición del bit. Entonces, el nodo comprueba 300 el vector de bits para ver si está puesta la posición del bit. Si no está puesto el bit, lo que significa que esta RESPUESTA no está relacionada con una CONVERSIÓN procesada anteriormente, entonces se descarta 302 el paquete. Si está puesta la posición del bit, lo que significa que esta RESPUESTA está relacionada con una CONVERSIÓN procesada anteriormente, se pone 304 a cero la posición del bit. Al poner a cero la posición del bit el nodo ignorará mensajes idénticos adicionales de RESPUESTA que pueden ser enviados como parte de un ataque de reproducción desde un nodo malicioso. Entonces, el nodo comprueba para asegurarse que todos los PAC en el mensaje de RESPUESTA son válidos 306 antes de procesar la RESPUESTA y de remitirla al siguiente salto. Si cualquiera de los PAC son inválidos 306, entonces el nodo rechazará 310 el paquete.

El procedimiento de CONVERSIÓN menciona convertir una petición de CONVERSIÓN en una RESPUESTA. Esta gestión de la RESPUESTA que se acaba de exponer implica asegurar que la RESPUESTA se corresponde con una CONVERSIÓN recibida recientemente, y remitir la RESPUESTA al siguiente salto especificado. Como ejemplo, asume que el nodo P recibe un paquete S de RESPUESTA que contiene una ID PNRP objetivo, un PAC MejorCorrespondencia, y una vía que enumera las direcciones de todos los nodos que procesaron la CONVERSIÓN original antes que este nodo, terminando con la dirección de PNRP de este propio nodo. El nodo P da acuse de recibo de la RESPUESTA con un ACK. El nodo P comprueba la vía de RESPUESTA en busca de su propia dirección. Su dirección debe ser la última entrada en la lista de direcciones para que este paquete sea válido. El nodo P también comprueba su vector de bits recibido para garantizar que la RESPUESTA se corresponde con una CONVERSIÓN vista recientemente. Si la RESPUESTA no se corresponde con un campo en el vector de bits recibido, o si la dirección de P no es la última dirección en la lista de vías, se desecha discretamente la RESPUESTA, y se detiene el procesamiento. P valida el PAC MejorCorrespondencia y lo añade a su memoria intermedia local. Si la MejorCorrespondencia es inválida, se desecha discretamente la RESPUESTA, y se detiene el procesamiento. P elimina su dirección del final de la vía de RESPUESTA. Continúa eliminando entradas del final de

la vía de RESPUESTA hasta que la entrada final tiene una bandera puesta que indica un nodo que ACEPTÓ la petición correspondiente de CONVERSIÓN. Si la vía está vacía ahora, la CONVERSIÓN correspondiente se originó localmente. El PNRP realiza una comprobación de validación de la identidad en la MejorCorrespondencia. Si la comprobación de validación de la identidad tiene éxito, se deja pasar la MejorCorrespondencia al gestor de peticiones, de lo contrario se deja pasar una indicación de fallo. Si la vía está vacía, el procesamiento se ha completado. Si la vía no está vacía, el nodo remite el paquete de RESPUESTA a la entrada final en la lista de vías.

Existe una necesidad de una revocación de certificados de direcciones PNRP siempre que el certificado de dirección publicada se vuelve inválido antes de la fecha de caducidad del certificado (campo de Validez/Hasta). Ejemplos de tales eventos son cuando un nodo se desconecta de forma ordenada de la red de P2P, o cuando un nodo deja un grupo, etc. El mecanismo de revocación tiene pues la bandera de revocación, y la fecha de Desde del campo de Validez puesta a la hora actual (o el momento en el que debe ser revocado el certificado) y el campo de Hasta puesto al mismo valor que los certificados anunciados anteriormente. Se considera que son revocados todos los certificados para los que se satisfacen todas las anteriores condiciones: el certificado está firmado por el mismo emisor; la ID se corresponde con la ID en el certificado de revocación; los campos de la Dirección coinciden con los del certificado de revocación; la fecha de Hasta del campo de Validación es la misma que la fecha de Hasta del campo de Validación en el certificado de revocación; y la fecha de Desde del campo de Validación precede a la fecha de Hasta del campo de Validación en el certificado de revocación. Dado que el certificado de revocación está firmado, se garantiza que un nodo malicioso no pueda desconectar a nadie de la nube.

Se ha presentado la anterior descripción de diversas realizaciones de la invención con fines ilustrativos y descriptivos. No se pretende que sea exhaustiva ni que limite la invención a las realizaciones precisas dadas a conocer. Son posibles numerosas modificaciones o variaciones en consideración de las anteriores enseñanzas. Las realizaciones dadas a conocer fueron escogidas y descritas para proporcionar la mejor ilustración de los principios de la invención y su aplicación práctica para permitir, de ese modo, a una persona con un nivel normal de dominio de la técnica a utilizar la invención en diversas realizaciones y con diversas modificaciones que sean adecuadas al uso particular contemplado.

Todas las modificaciones y las variaciones tales se encuentran dentro del alcance de la invención, como se determina por medio de las reivindicaciones adjuntas cuando son interpretadas según el alcance al que tienen derecho justa, legal y equitativamente.

REIVINDICACIONES

1. Un procedimiento para inhibir un ataque de denegación de servicio basado en búsquedas que intenta dificultar la resolución de direcciones en una red del mismo nivel, que comprende las etapas de:

recibir un mensaje de RESPUESTA (296);

determinar si el mensaje de RESPUESTA es en respuesta a un mensaje anterior de CONVERSIÓN,

pedir la resolución de un nombre objetivo de una unidad del mismo nivel en una dirección;

rechazar el mensaje (302, 310) de RESPUESTA cuando el mensaje de RESPUESTA no es en respuesta al anterior mensaje de CONVERSIÓN,

en el que el mensaje de RESPUESTA contiene una lista de direcciones de todos los nodos en la red del mismo nivel que procesaron el mensaje de CONVERSIÓN antes de que un nodo compruebe el mensaje de RESPUESTA y termina con la dirección del propio nodo, y

comprende, además, las etapas de:

determinar si el mensaje de RESPUESTA ha sido modificado en un intento por dificultar la resolución (306), que comprende las etapas de comprobar si la dirección de este nodo es la última entrada en la lista de direcciones y de eliminar la dirección del nodo del final de la lista, y

rechazar el mensaje (310) de RESPUESTA cuando el mensaje de RESPUESTA ha sido modificado en un intento por dificultar la resolución.

2. El procedimiento de la reivindicación 1, en el que la etapa de determinar si el mensaje de RESPUESTA es en respuesta a un mensaje anterior de CONVERSIÓN comprende las etapas de calcular una posición del bit como una clave calculada de información en el mensaje (298) de RESPUESTA, y analizar un vector (300) de bits mantenido por el nodo que comprueba el mensaje de RESPUESTA para determinar si un bit correspondiente a la posición del bit está puesto en el mismo, en el que la posición del bit es un número utilizado como un índice en el vector de bits.

3. El procedimiento de la reivindicación 1, en el que la etapa de determinar si el mensaje de RESPUESTA ha sido modificado en un intento por dificultar la resolución comprende las etapas de calcular una posición del bit como una clave calculada que incluye la lista de direcciones en el mensaje de RESPUESTA, y analizar un vector de bits mantenido por el nodo que comprueba el mensaje de RESPUESTA para determinar si un bit correspondiente a la posición del bit está puesto en el mismo, en el que la posición del bit es un número utilizado como un índice en el vector de bits.

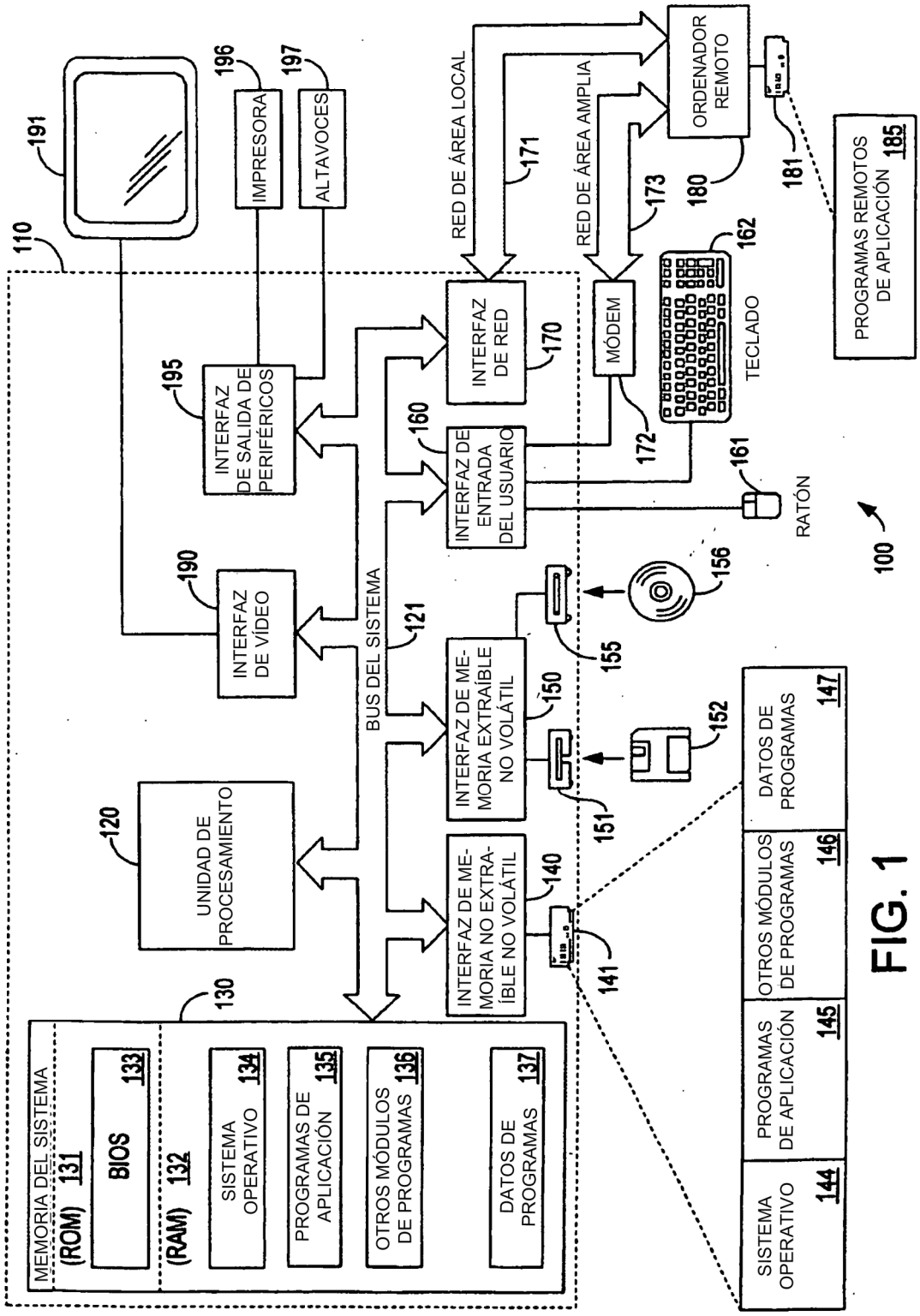

MEMORIA DEL SISTEMA
(ROM) 131
BIOS 133
(RAM) 132
SISTEMA OPERATIVO 134
PROGRAMAS DE APLICACIÓN 135
OTROS MÓDULOS DE PROGRAMAS 136
DATOS DE PROGRAMAS 137
130
UNIDAD DE PROCESAMIENTO
120
INTERFAZ DE MEMORIA NO EXTRAÍBLE NO VOLÁTIL
140
INTERFAZ DE MEMORIA EXTRAÍBLE NO VOLÁTIL
150
INTERFAZ DE VÍDEO
190
INTERFAZ DE SALIDA DE PERIFÉRICOS
195
INTERFAZ DE ENTRADA DEL USUARIO
160
INTERFAZ DE RED
170
BUS DEL SISTEMA
121
110
191
IMPRESORA
196
ALTAVOCES
197
RED DE ÁREA LOCAL
171
RED DE ÁREA AMPLIA
173
MÓDEM
172
ORDENADOR REMOTO
180
PROGRAMAS REMOTOS DE APLICACIÓN
185
181
TECLADO
162
RATÓN
161
100
SISTEMA OPERATIVO
144
PROGRAMAS DE APLICACIÓN
145
OTROS MÓDULOS DE PROGRAMAS
146
DATOS DE PROGRAMAS
147
141
151
152
155
156

FIG. 1

FIG. 2

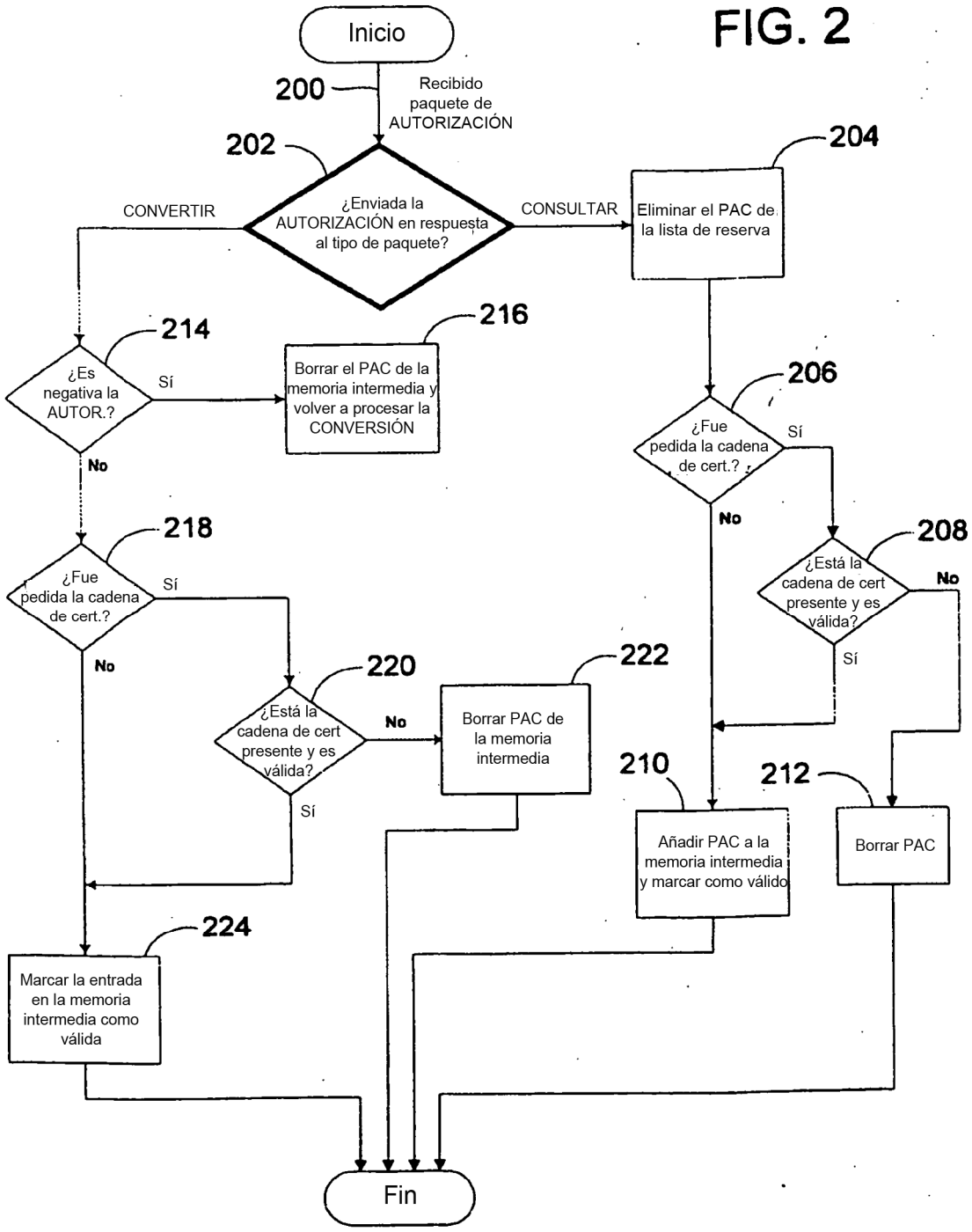

Inicio
200
Recibido paquete de AUTORIZACIÓN
202
¿Enviada la AUTORIZACIÓN en respuesta al tipo de paquete?
CONVERTIR
CONSULTAR
204
Eliminar el PAC de la lista de reserva
214
¿Es negativa la AUTOR.?
Sí
216
Borrar el PAC de la memoria intermedia y volver a procesar la CONVERSIÓN
206
¿Fue pedida la cadena de cert.?
Sí
No
208
¿Está la cadena de cert presente y es válida?
No
Sí
No
218
¿Fue pedida la cadena de cert.?
Sí
No
220
¿Está la cadena de cert presente y es válida?
No
222
Borrar PAC de la memoria intermedia
Sí
210
Añadir PAC a la memoria intermedia y marcar como válido
212
Borrar PAC
224
Marcar la entrada en la memoria intermedia como válida
Fin

FIG. 3

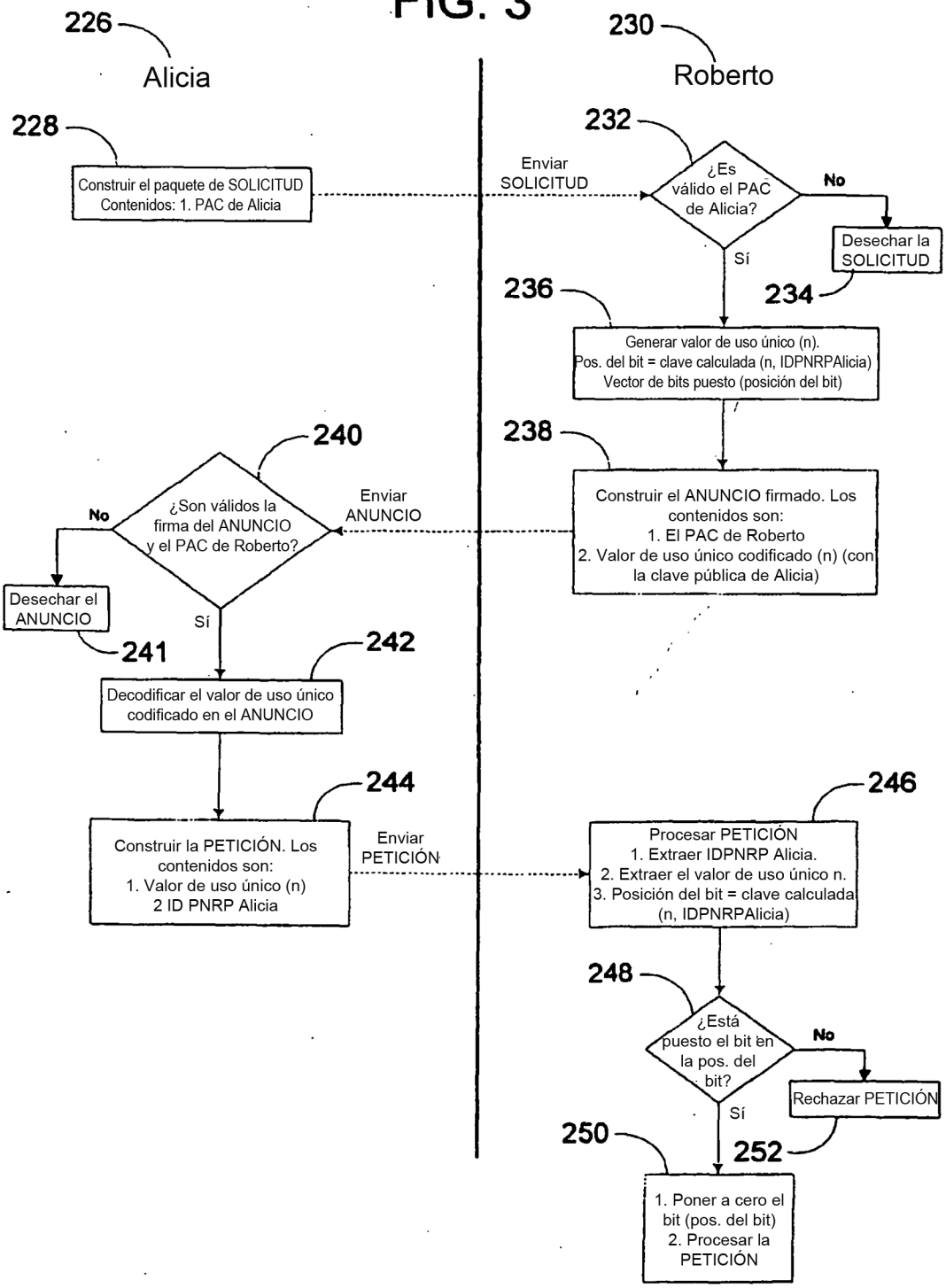

226
Alicia
230
Roberto
228
Construir el paquete de SOLICITUD
Contenidos: 1. PAC de Alicia
Enviar SOLICITUD
232
¿Es válido el PAC de Alicia?
No
Desechar la SOLICITUD
234
Sí
236
Generar valor de uso único (n).
Pos. del bit = clave calculada (n, IDPNRPAlicia)
Vector de bits puesto (posición del bit)
238
Construir el ANUNCIO firmado. Los contenidos son:
1. El PAC de Roberto
2. Valor de uso único codificado (n) (con la clave pública de Alicia)
Enviar ANUNCIO
240
¿Son válidos la firma del ANUNCIO y el PAC de Roberto?
No
Desechar el ANUNCIO
241
Sí
242
Decodificar el valor de uso único codificado en el ANUNCIO
244
Construir la PETICIÓN. Los contenidos son:
1. Valor de uso único (n)
2 ID PNRP Alicia
Enviar PETICIÓN
246
Procesar PETICIÓN
1. Extraer IDPNRP Alicia.
2. Extraer el valor de uso único n.
3. Posición del bit = clave calculada (n, IDPNRPAlicia)
248
¿Está puesto el bit en la pos. del bit?
No
Rechazar PETICIÓN
Sí
250
252
1. Poner a cero el bit (pos. del bit)
2. Procesar la PETICIÓN

FIG. 4

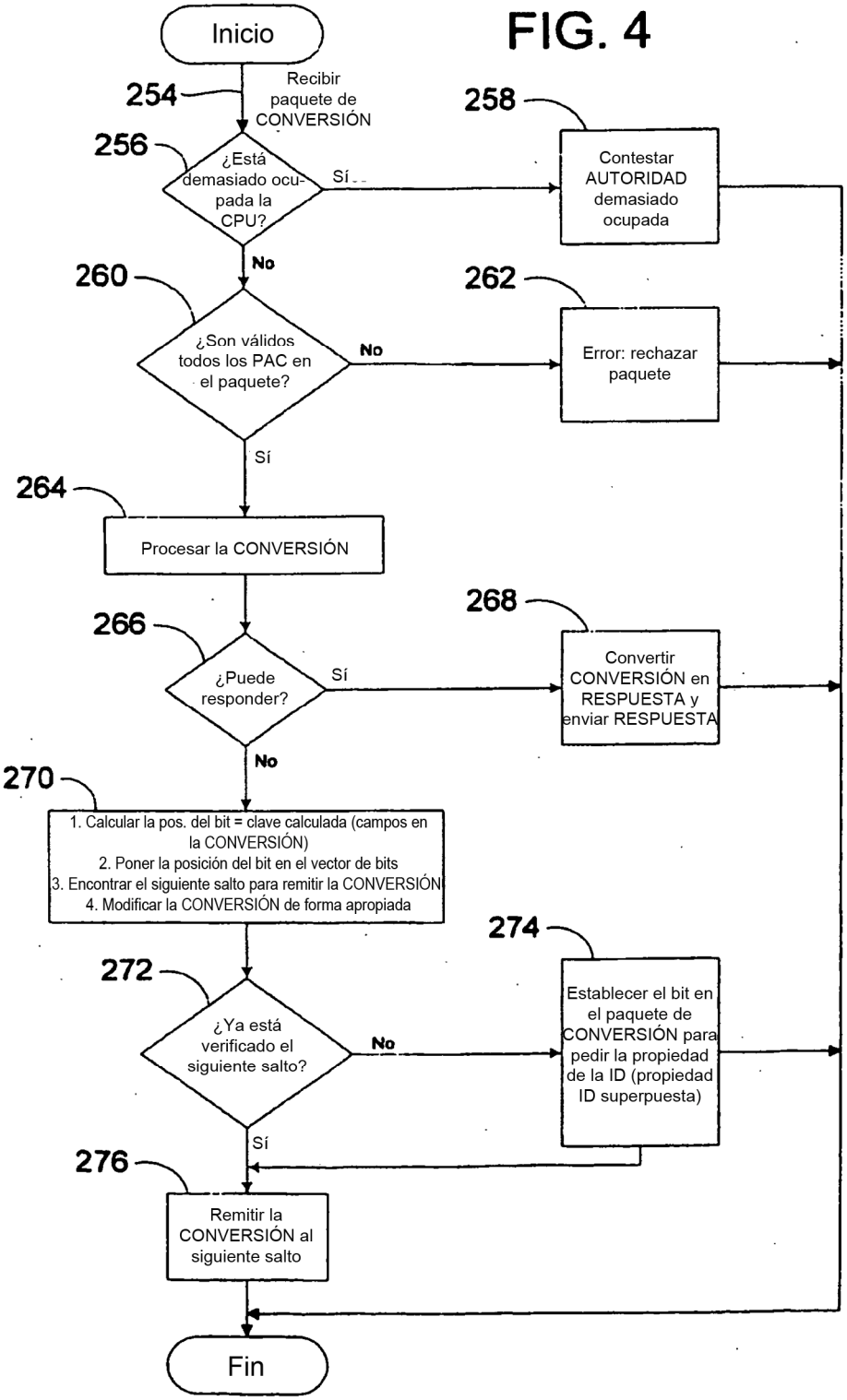

Inicio
254
Recibir paquete de CONVERSIÓN
256
¿Está demasiado ocupada la CPU?
Sí
258
Contestar AUTORIDAD demasiado ocupada
No
260
¿Son válidos todos los PAC en el paquete?
No
262
Error: rechazar paquete
Sí
264
Procesar la CONVERSIÓN
266
¿Puede responder?
Sí
268
Convertir CONVERSIÓN en RESPUESTA y enviar RESPUESTA
No
270
1. Calcular la pos. del bit = clave calculada (campos en la CONVERSIÓN)
2. Poner la posición del bit en el vector de bits
3. Encontrar el siguiente salto para remitir la CONVERSIÓN
4. Modificar la CONVERSIÓN de forma apropiada
272
¿Ya está verificado el siguiente salto?
No
274
Establecer el bit en el paquete de CONVERSIÓN para pedir la propiedad de la ID (propiedad ID superpuesta)
Sí
276
Remitir la CONVERSIÓN al siguiente salto
Fin

FIG. 5

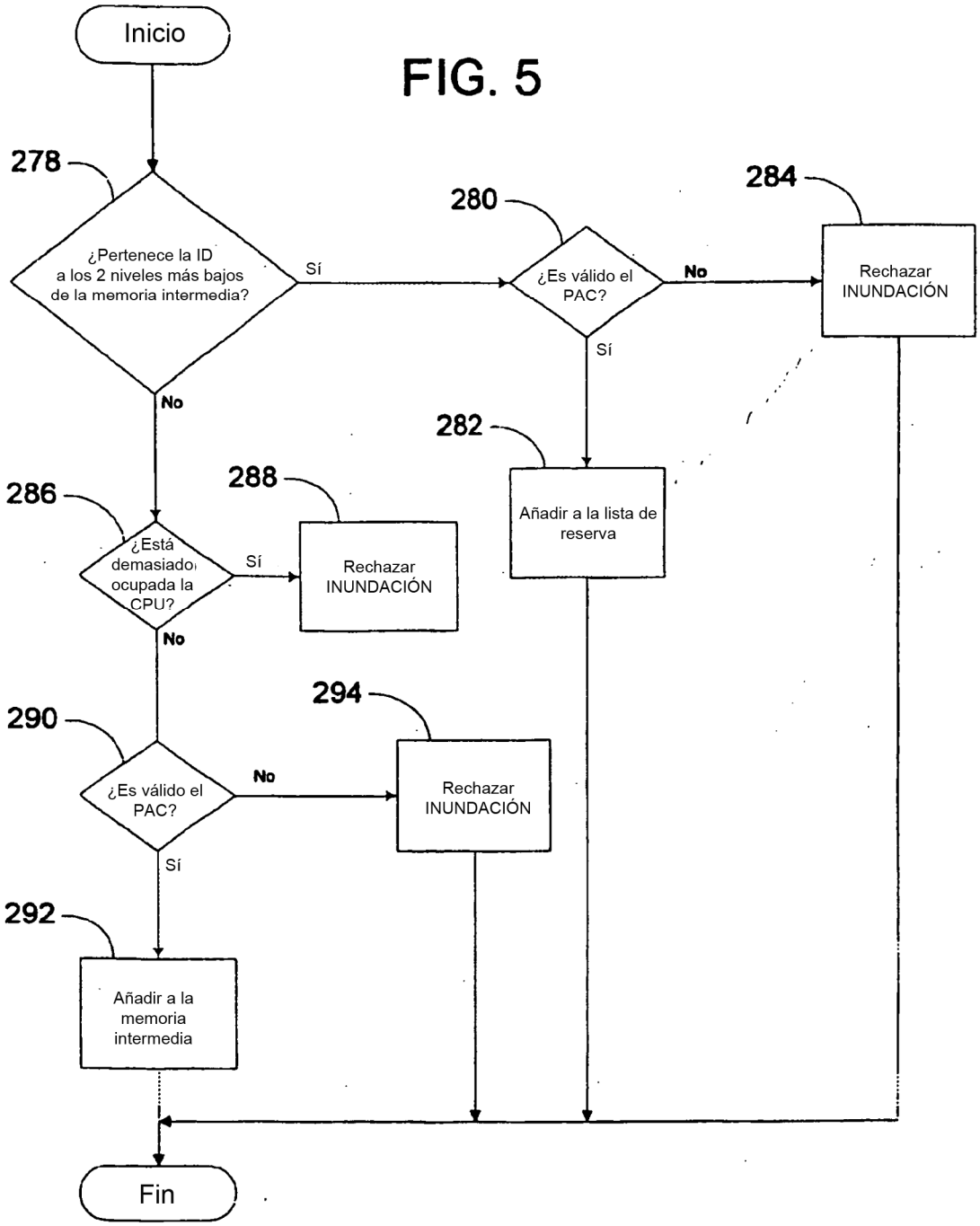
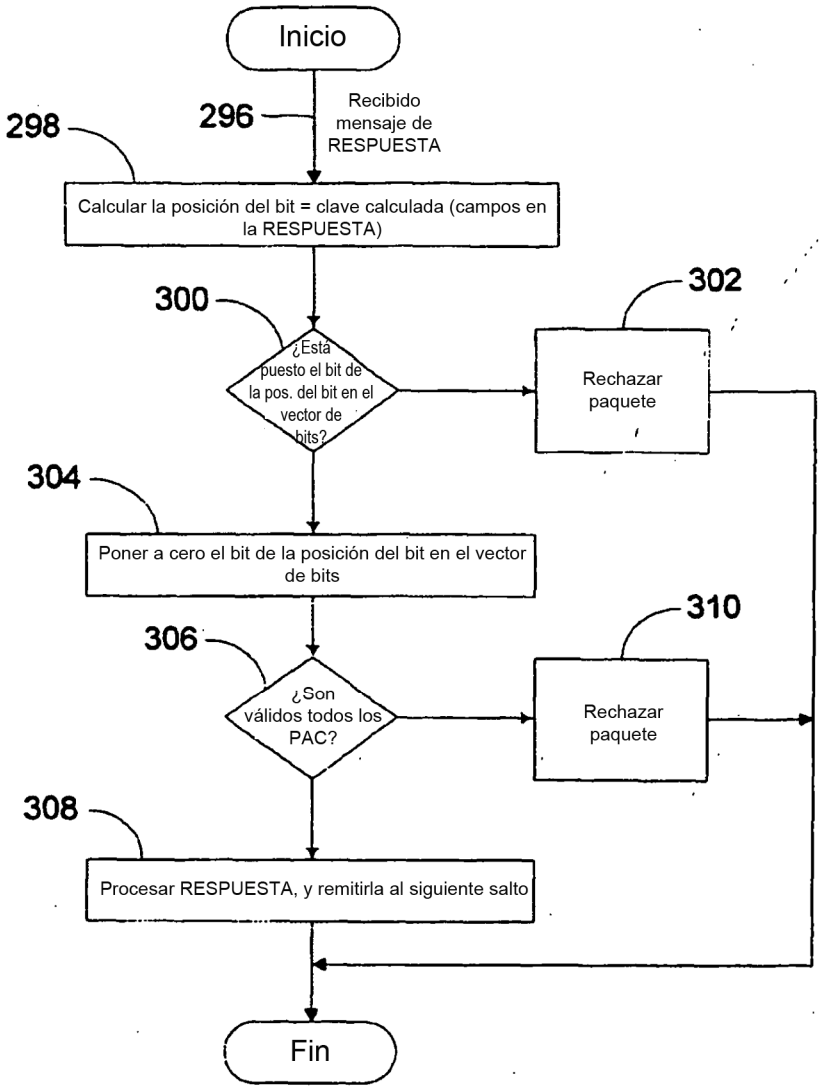

Inicio
278
¿Pertenece la ID a los 2 niveles más bajos de la memoria intermedia?
Sí
280
¿Es válido el PAC?
No
284
Rechazar INUNDACIÓN
Sí
No
282
Añadir a la lista de reserva
288
286
¿Está demasiado ocupada la CPU?
Sí
Rechazar INUNDACIÓN
No
294
290
¿Es válido el PAC?
No
Rechazar INUNDACIÓN
Sí
292
Añadir a la memoria intermedia
Fin

FIG. 6


Inicio
296
Recibido mensaje de RESPUESTA
298
Calcular la posición del bit = clave calculada (campos en la RESPUESTA)
300
¿Está puesto el bit de la pos. del bit en el vector de bits?
302
Rechazar paquete
304
Poner a cero el bit de la posición del bit en el vector de bits
306
¿Son válidos todos los PAC?
310
Rechazar paquete
308
Procesar RESPUESTA, y remitirla al siguiente salto
Fin