(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 369 132**

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

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Número de solicitud europea: **10176040 .3**
(96) Fecha de presentación: **15.06.2000**
(97) Número de publicación de la solicitud: **2254311**
(97) Fecha de publicación de la solicitud: **24.11.2010**

(54) Título: **MANTENIMIENTO DE CONVERSIÓN DE DIRECCIONES PARA DATOS DE COMUNICACIÓN.**

(30) Prioridad:
15.06.1999 US 333829

(45) Fecha de publicación de la mención BOPI:
25.11.2011

(45) Fecha de la publicación del folleto de la patente:
25.11.2011

(73) Titular/es:
Tectia Oyj
Kumpulantie 3
00520 Helsinki

(72) Inventor/es:
Kivinen, Tero y
Ylonen, Tatu

(74) Agente: **de Elzaburu Márquez, Alberto**

DESCRIPCIÓN

Mantenimiento de conversión de direcciones para datos de comunicación.

La invención se refiere en general al campo de las comunicaciones seguras entre ordenadores en redes de transmisión de datos basadas en conmutación de paquetes. Más concretamente, el invento se refiere al campo del establecimiento y mantenimiento de conexiones de comunicaciones seguras a través de una Conversión o Transformación de direcciones de red o conversión de protocolo.

El Grupo de Trabajo de Ingeniería de Internet (IETF) ha normalizado el conjunto de programas de protocolo IPSEC (Internet Protocol Security); las normas se conocen bien a través de los Request For Comments o documentos RFC números RFC2401, RFC2402, RFC2406, RFC2407, RFC2408 y RFC2409 que se mencionan en la lista de referencias anexa. Los protocolos IPSEC proporcionan seguridad al Protocolo de Internet o IP especificado en sí mismo en el documento RFC791. IPSEC realiza la autenticación y encriptación a nivel de paquete generando una nueva cabecera IP que añade delante del paquete una Cabecera de Autenticación (AH) o una cabecera de Carga (Payload) de Seguridad de Encapsulación (ESP). El paquete original se autentica criptográficamente y puede ser encriptado opcionalmente. El método usado para autenticar y encriptar opcionalmente un paquete se identifica mediante un valor de índice de parámetros de seguridad (SPI) almacenado en las cabeceras AH y ESP. El documento RFC número RFC2401 especifica un modo de transporte y un modo de entunelación (tunnelling) para paquetes; el presente invento puede aplicarse con independencia de cuál de estos modos sea el utilizado.

En los últimos años, cada vez más vendedores y proveedores de servicios de Internet han empezado a desarrollar la conversión de direcciones de red (NAT). Se encuentran referencias a NAT al menos en el documento RFC número RFC1631, así como en los documentos que están identificados en la lista de referencias anexa como Srisuresh98Terminology, SrisureshEgevang98, Srisuresh98Scurity, HoldregeSrisuresh99, TYS99, Rekhter99, LoBorella99 y BorellaLo99. Hay dos formas fundamentales de conversión de direcciones, ilustradas de forma esquemática en Figs. 1a y 1b: el anfitrión NAT 101 y el puerto NAT 151. El anfitrión NAT 101 sólo convierte las direcciones IP en un paquete entrante 102 de modo que un paquete saliente 103 tiene una dirección IP distinta. El puerto NAT también toca los números puerto de TCP y de UDP (Protocolo de Control de Tráfico; Protocolo de Datagrama de Usuario) en un paquete entrante 152, multiplexando varias direcciones IP a una sola dirección IP en un paquete saliente 153 y desmultiplexando correspondientemente una sola dirección IP en diversas direcciones IP para paquetes que viajan en el sentido contrario (no mostrado). Los puertos NAT son especialmente habituales en los entornos domésticos y de pequeñas oficinas. En las Figs. 1a y 1b se muestra, solamente con fines de claridad gráfica, la separación física de las conexiones de entrada y salida para dispositivos de NAT; en la práctica hay muchas formas posibles de conectar físicamente una NAT.

La conversión de direcciones tiene lugar más frecuentemente en el borde de una red local (es decir, conversión entre múltiples direcciones locales privadas por un lado y unas pocas direcciones públicas encaminables globalmente en el otro). La mayoría de las veces se utiliza un puerto NAT y hay una sola dirección encaminable globalmente. En la Fig. 1b se ilustra de forma esquemática un red local 154. Este tipo de disposiciones se están haciendo extremadamente comunes en los mercados domésticos y pequeñas oficinas. Algunos proveedores de servicios de Internet han empezado también a dar a sus clientes direcciones privadas y a realizar la conversión de direcciones de red de dichas direcciones en sus redes básicas. En general, la conversión de direcciones de red se ha analizado ampliamente y en profundidad, por ejemplo en el grupo de trabajo de NAT dentro del Grupo de Trabajo de Ingeniería de Internet. Los principios operativos de un dispositivo de NAT son ampliamente conocidos, y existen muchas implementaciones de múltiples vendedores disponibles en el mercado, incluidas varias implementaciones con códigos fuentes de acceso gratis. La operación típica de una NAT se puede describir de modo que pone en correspondencia direcciones IP y combinaciones de puertos con diferentes direcciones IP y combinaciones de puertos. La puesta en correspondencia se mantendrá constante durante la duración de una conexión de red, pero puede cambiar (despacio) con el tiempo. En la práctica, la funcionalidad de NAT se integra con frecuencia en un cortafuego o un encaminador.

La Fig. 1c ilustra un ejemplo práctico del caso de una comunicación de red en la que un nodo transmisor 181 está ubicado en una primera red de área local (también conocida como la primera red privada) 182, que tiene un puerto NAT 183 para conectarse a una red general 184 de transmisión de datos basada en conmutación de paquetes de amplia área, como la Internet. Esta última consta de un gran número de nodos interconectados de forma arbitraria. Un nodo receptor 185 está situado en una segunda red de área local 186 que está a su vez acoplada a una red de área-ancha a través de una NAT 187. Las denominaciones “nodo transmisor” y “nodo receptor” son algo engañosas, dado que se necesita una comunicación bidireccional para establecer servicios de seguridad de la red. El nodo transmisor es el que inicia la comunicación. También se utilizan los términos “Iniciador” y “Respondedor” para el nodo transmisor y el nodo receptor, respectivamente.

El propósito de la Fig. 1c es enfatizar el hecho de que los nodos de comunicación no se percatan ni del número ni de la naturaleza de los dispositivos intermedios a través de los cuales se comunican ni de la naturaleza de las transformaciones que tienen lugar. Además de las NAT, hay otro tipo de dispositivos en la red de Internet que pueden modificar legalmente paquetes durante su transmisión. Un ejemplo típico es un convertidor de protocolo, cuya principal función es convertir el paquete en un protocolo diferente sin perturbar la operación normal. Su uso

conlleva problemas muy similares al caso de la NAT. Un ejemplo bastante sencillo pero importante es la conversión entre IPv4 e IPv6, que son diferentes versiones del Protocolo de Internet. Los mencionados convertidores serán extremadamente importantes y comunes en un futuro próximo. Un paquete puede sufrir numerosas conversiones de este tipo a lo largo de su recorrido, y es posible que los puntos finales de las comunicaciones utilicen de hecho un protocolo diferente. Lo mismo que NAT, la conversión de protocolo tiene lugar habitualmente en encaminadores y cortafuegos.

En la comunidad IPSEC se sabe bien que el protocolo IPSEC no funciona adecuadamente en las conversiones de direcciones de red. Este problema se ha analizado en al menos los documentos referidos como HoldregeSrisuresh99 y Rekhter99.

En la solicitud de patente finlandesa número 974665 y la correspondiente solicitud PCT número FI98/701032 se ha presentado un determinado método para realizar conversión de direcciones IPSEC y un método de autenticación de paquetes que es sensible a las conversiones de direcciones y conversiones de protocolo en ruta del paquete. En dichas solicitudes se ha presentado adicionalmente un dispositivo de red transmisora y un dispositivo de red receptora que son capaces de utilizar las ventajas del método mencionado anteriormente. Sin embargo, en dichas solicitudes de patentes previas permanecen sin resolverse algunos de los problemas relativos al suministro de servicios de seguridad en red sobre conversión de direcciones de red.

La patente US-A-5 793 763 proporciona un sistema y un método para convertir direcciones locales IP a una única dirección IP global de acuerdo con el bien conocido principio de las redes NAT. Los paquetes que llegan de la red Internet se apantallan mediante un algoritmo de seguridad adaptable.

El proyecto de Internet del Grupo de Trabajo de Redes “Terminología de Referencia para Prestaciones de Cortafuegos” (‘Bench-marking Terminology for Firewall Performance’) de Mayo de 1999 por D. Newman, XPO150161991SSN, sección 3.10 “Mantenimiento de Conexión” describe el uso de “mantenero en activación” datos para mantener una conexión a través de un cortafuego en algunas implementaciones de TCP y otros protocolos de conexión orientada durante el período en el que no se han intercambiado datos de usuario.

Es un objetivo de la presente invención presentar un método y un aparato para proporcionar servicios de seguridad de red a una red sobre conversión de direcciones de red de forma fiable y ventajosa.

El método de acuerdo con la invención está definido en las reivindicaciones independientes 1 y 2 y los dispositivos acordes con la invención están definidos en la reivindicaciones independientes 8 y 9. Aspectos más detallados del invento están definidos en las reivindicaciones dependientes.

La Fig. 1a ilustra el uso conocido de un anfitrión NAT,

La Fig. 2b ilustra el uso conocido de un puerto NAT,

La Fig. 1c ilustra una conexión de comunicación conocida entre nodos a través de una red basada en conmutación de paquetes,

La Fig. 2a ilustra una cierta carga neta de ID de Vendedor aplicable dentro del contexto de la invención,

La Fig. 2b. ilustra una cierta carga neta privada aplicable dentro del contexto de la invención,

La Fig. 2c ilustra una cierta estructura de cabecera combinada aplicable dentro del contexto de la invención,

La Fig. 3 ilustra ciertos pasos del método relativos a la aplicación de la invención,

La Fig. 4 ilustra una transformación de estructuras de cabecera acordes con un aspecto de la invención, y

La Fig. 5 ilustra un diagrama de bloques simplificado de un dispositivo de red utilizado para implementar el método de acuerdo con la invención.

La presente invención combina y amplía algunos de los métodos de conversión de direcciones de red, entunelación sobre UDP, IKE y los mecanismos de extensión IKE, de una forma nueva y con carácter inventivo para producir un método para comunicaciones seguras a través de conversiones de direcciones de red y conversiones de protocolo. El método puede hacerse de forma totalmente automática y transparente para el usuario.

Un punto clave relacionado con la aplicabilidad del invento es que – en la fecha de prioridad de la presente solicitud de patente – en general, sólo TCP (descrito en RFC793) y UDP (descrito en RFC768) funcionan sobre NAT. Esto es debido a que la mayoría de las NAT usadas en la práctica son puertos NAT, y ésta es la forma en que la NAT proporciona el máximo de beneficios con respecto a la escasez de direcciones IP encaminables globalmente. La invención no está, sin embargo, limitada al uso de UDP y TCP tal y como se conocen en la fecha de prioridad de esta solicitud de patente: en general puede decirse que la UDP y TCP son ejemplos de protocolos que determinan la información de identificación de conexión (es decir, direccionamiento y numeración de puertos) que es hecha

corresponder a otra forma en el proceso de conversión de dirección. Cabe esperar que en el futuro surjan otros tipos de protocolos de comunicación y de transformaciones de dirección.

Los diversos aspectos de la invención se refieren a:

- determinar si un anfitrión distante soporta un cierto método que es típicamente un método de comunicación segura acorde con la invención (el aspecto “métodos soportados”),

- determinar qué conversiones de direcciones de red y/o conversiones de protocolo tienen lugar en los paquetes, en caso de haber alguna (el aspecto “conversiones en curso”),

- entunelar paquetes dentro de un protocolo cuidadosamente seleccionado, típicamente UDP, para hacerlos recorrer la NAT (el aspecto “entunelación seleccionada”),

- usar un método de mantenimiento en activación para asegurar que los dispositivos NAT involucrados y otros dispositivos que usan tiempos límite para mapeos, no pierdan el mapeo para los equipos de comunicación (el aspecto “mantenimiento en activación”),

- compensar las conversiones que tienen lugar antes verificando el código de autenticación de mensaje para paquetes AH (el aspecto “compensación/autenticación”) y

- realizar conversiones de direcciones ya sea en el nodo emisor o en el receptor para compensar los diversos anfitriones que han sido hechos corresponder a una sola dirección pública (el aspecto “compensación/puesta en correspondencia”).

Se llama entunelación al proceso de encapsulación de paquetes de datos para transmisión sobre una red lógica distinta. Típicamente, en el caso de protocolo IP, la entunelación implica añadir una nueva cabecera IP delante del paquete inicial, estableciendo apropiadamente el campo de protocolo en la nueva cabecera, y enviando el paquete al destino deseado (extremo del túnel). También se puede realizar la entunelación modificando los campos de la cabecera del paquete inicial o remplazándolos por otra cabecera, siempre que en el proceso se conserve la cantidad de información sobre el paquete original suficiente para que al final de túnel se pueda reconstruir dicho paquete de forma suficientemente parecida al paquete inicial que entró en el túnel. La cantidad exacta de información que debe ser hecha pasar con el paquete depende de los protocolos de red, y la información puede ser hecha pasar ya sea de modo explícito (como parte del paquete entunelado) o implícito (por el contexto, como por ejemplo determinado por paquetes transmitidos previamente o por un identificador de contexto en el paquete entunelado).

En el estado de la técnica se conoce bien cómo entunelar paquetes sobre una red. Al menos los documentos que se han referenciado como RFC1226, RFC1234, RFC1241, RFC1326, RFC1701, RFC1853, RFC2003, RFC2004, RFC2107, RFC2344, RFC2401, RFC2406, RFC2473 y RFC2529 están relacionados con el tema de la entunelación. Por ejemplo, RFC1234 presenta un método para entunelar marcos IPX sobre UDP. En ese método, los paquetes se entunelan a un puerto fijo UDP y a la dirección IP del desencapsulador.

El protocolo IPSEC mencionado en la descripción de los antecedentes utiliza casi siempre el protocolo de Intercambio de Clave Internet o IKE (conocido por los documentos RFC2409, RFC2408 y RFC2407) para autenticar las partes que se comunican entre ellas, derivando un secreto compartido conocido solamente por las partes que se comunican, negociando los métodos de autenticación y encriptación que deben usarse en la comunicación, y acordando un valor del índice de parámetro de seguridad (SPI) así como un conjunto de selectores que serán los usados en la comunicación. El protocolo IKE se conocía previamente como el ISAKMP/Oakley, donde el acrónimo ISAKMP responde a Internet Security Association Key Management Protocol. Además de la ya mencionada negociación normal especificada en la norma IKE, IKE también incluye ciertos mecanismos de extensión. La carga neta de ID del Vendedor, divulgada en el documento de referencia RFC2408, permite a las partes en comunicación determinar si la otra parte soporta un particular mecanismo de extensión privada. El IPSEC DOI (Domain of Interpretation), conocido como RFC2407, reserva ciertos valores numéricos para dichas extensiones privadas.

En la actualidad, la ya conocida carga neta de datos de identificación de Fabricante o Vendedor se define para que tenga el formato ilustrado en la Fig. 2a, donde la columna de números se corresponde con las posiciones de los bits. El campo de ID 201 del Vendedor es, para los fines de esta invención, la parte más importante de la carga neta de ID de Vendedor. A continuación se explica cómo se puede realizar, en el contexto del protocolo IKE, la negociación sobre si un equipo distante soporta un determinado método para proporcionar comunicaciones seguras sobre una red. La terminología utilizada aquí está tomada de los documentos INE.

El protocolo IKE determina la llamada Fase 1 del intercambio mutuo de mensajes entre el Iniciador (es decir, el nodo que envía en primer lugar un paquete al otro) y el Respondedor (es decir, el nodo que recibe en primer lugar un paquete). La Fig. 3 ilustra un intercambio de los primeros mensajes de Fase 1 entre el Iniciador y el Respondedor. De acuerdo con el aspecto de la invención “métodos soportados”, ambos dispositivos incluyen una cierta carga neta de ID del Vendedor en un cierto mensaje de la Fase 1 que es preferiblemente su primer mensaje de Fase 1. Esta carga neta indica que soportan el método en cuestión. En la Fig. 3 los campos de ID del Vendedor contenidos dentro del primer mensaje de la Fase 1 (u otro distinto) del Iniciador están esquemáticamente mostrados como se ha

mostrado esquemáticamente como 201´ y los otros campos de ID del Vendedor contenidos dentro del primer mensaje de la Fase 1 (u otro distinto) del Respondedor están esquemáticamente mostrados como 201´´. La presencia de un determinado método se indica mediante el campo de ID del Vendedor en la Carga neta de ID del Vendedor es básicamente una identificación de ese método: ventajosamente el hash MD5 de una cadena de identificación previamente conocida, por ejemplo “SSH IPSEC NAT Transversal Version 1”, sin ningún cero posterior ni nuevas líneas. La generación de hashes MD5 de secuencias arbitrarias de caracteres es una técnica ampliamente conocida en el estado de la técnica, por ejemplo de la publicación RFC1321 mencionado en la lista de referencias.

A continuación se abordará el aspecto de la invención “conversiones en curso”. Además de la Fase 1 mencionada anteriormente, el protocolo IKE determina la llamada Fase 2 del intercambio mutuo de mensajes entre el Iniciador y el Respondedor. De acuerdo con el aspecto de la invención “conversiones en curso” las partes pueden determinar qué conversiones tendrán lugar incluyendo las direcciones IP que ven en las cargas netas privadas de ciertos mensajes de Modo Rápido de Fase 2, que son preferiblemente sus primeros mensajes de Modo Rápido de Fase 2. Cualquier número no utilizado en el intervalo de números de la carga neta privada puede usarse para designar dicho uso de la carga neta privada (por ejemplo 157, que es un número no utilizado hasta la fecha de prioridad de la presente solicitud de patente).

La carga neta privada usada para desvelar las conversiones en curso puede tener por ejemplo el formato ilustrado en la Fig. 2b. El campo 211 contiene un código de tipo que identifica los tipos de direcciones que aparecen en los campos 212 y 213. El campo 212 contiene la dirección del Iniciador tal y como lo ve el nodo que envía el mensaje, y el campo 213 contiene la dirección del Respondedor tal y como la ve el nodo que envía el mensaje. La Fig. 3 muestra el intercambio de los (primeros) mensajes Modo Rápido de Fase 2 entre el Iniciador y el Respondedor de modo tal que el mensaje enviado por el primero incluye los campos correspondientes 211´, 212´ y 213´ y el mensaje enviado por el último incluye los campos 211´´, 212´´ y 213´´.

De acuerdo con la práctica habitual, las direcciones del Iniciador y del Respondedor se incluyen en la cabecera del paquete que contiene la carga neta de la Fig. 2b. En la cabecera son sensibles a las conversiones de direcciones y otros procesos mientras que en la carga neta privada no lo son. Cuando se recibe el paquete con la carga neta de la Fig. 2b, las direcciones contenidas en él se comparan con las que se ven en la cabecera. Si son distintas, se produce entonces una conversión de direcciones de red en el paquete. Posteriormente se hará referencia al uso del número de puerto estándar IKE 500 junto con la aplicación de la invención; como un modo adicional de detectar conversiones ocurridas, los números de puerto del paquete recibido pueden ser también comparados con el número de puerto estándar IKE 500 para determinar si se han producido conversiones de puertos.

Un aspecto de cierta importancia a la hora de gestionar las direcciones es que el puerto fuente UDP del paquete puede guardarse para un posterior uso. Generalmente se guardarían con las estructuras de datos para las asociaciones de seguridad Fase 1 ISAKMP, y se utilizarían para establecer el proceso de compensación para las asociaciones de seguridad Fase 2 IPSEC.

Para usar el método descrito anteriormente para implementar el aspecto de la invención de “conversiones ocurridas”, los anfitriones deben modificar sus cargas netas de identificación Fase 2: la carga neta ilustrada en la Fig. 2b no es conocida en las normas existentes. Una posibilidad consiste en restringir las cargas netas a los tipos ID_IPV4_ADDR e ID_IPV6_ADDR que serían apropiadas para una operación anfitrión a anfitrión.

A continuación se hará referencia a los aspectos de la invención “entunelación seleccionada”, “compensación/autenticación” y “compensación/puesta en correspondencia”. De acuerdo con este aspecto de la invención, los paquetes de datos reales pueden entunelarse sobre la misma conexión que se use para establecer las características de seguridad de la conexión de comunicación, por ejemplo la conexión UDP usada para IKE. Esto asegura que los paquetes de datos reales experimenten las mismas conversiones que sufrieron los paquetes IKE cuando se determinó la conversión. Partiendo de que se ha determinado el número de puerto estándar 500 para IKE, esto significaría que todos los paquetes se envían con origen puerto 500 y destino puerto 500, de modo que se necesita un método para distinguir los paquetes IKE auténticos de los que contienen datos encapsulados. Una posible forma de hacerlo consiste en valerse del hecho de que la cabecera de IK usada por los paquetes IKE auténticos contiene un campo de Cookie del Iniciador: se puede especificar que los Iniciadores que soportan este aspecto de la invención no generan nunca “cookies” con todo ceros en sus cuatro primeros bytes. Por tanto se usa el valor cero en los cuatro bytes correspondientes para reconocer el paquete como un paquete de datos entunelados. De este modo, los paquetes de datos entunelados tendrían cuatro bytes ceros al principio de la carga neta UDP, mientras que los paquetes IKE auténticos nunca los tendrían.

La Fig. 4 ilustra la encapsulación de paquetes reales IPSEC en UDP para su transmisión. Básicamente, se insertan en el propio paquete una cabecera UDP 403 y una pequeña cabecera intermedia 404 después de la cabecera IP 401 ya en el paquete (con el campo de protocolo copiado en la cabecera intermedia). La cabecera IP 401 se modifica ligeramente dando lugar a una cabecera IP modificada 401´. La carga neta IP 402 permanece inalterada. No se debe malinterpretar la sencilla ilustración del paquete IPSEC sin encapsular a la izquierda: este paquete no es de texto común sino que ha sido procesado según AH o ESP u otro protocolo de conversión correspondiente antes de su encapsulación en UDP.

En el presente documento y sin carácter limitativo de la generalidad, se supone que la encapsulación de acuerdo con la Fig. 4 la realizan siempre los mismos nodos que realizan el procesamiento IPSEC (ya sea un nodo final o un dispositivo VPN). Debe observarse también que en vez de encapsular los paquetes IPSE en UDP, podrían encapsularse en TCP. Esta opción requeriría probablemente el uso de falsos inicios y terminaciones de sesión de tal modo que el primer paquete tenga el bit SYN y el último paquete tenga el bit FIN, tal y como se especifica en el protocolo TCP.

Al encapsular un paquete real de datos o un “datagrama” según la Fig. 4, se altera la cabecera IP original 401 – definida en RFC791 – dando lugar a una cabecera IP modificada 401´ de la siguiente manera:

- El campo de Protocolo en la cabecera IP (no mostrado separadamente) se remplaza por el protocolo 17 para UDP de acuerdo con RFC768,

- El campo de Longitud Total en la cabecera IP (no mostrado separadamente) se aumenta en el tamaño combinado de las cabeceras UDP e intermedia (16 bytes en total) y

- Se vuelve a calcular el campo de Cabecera Verificar-suma en la cabecera IP (no mostrado separadamente) siguiendo las normas dadas en RFC791.

Tal y como se ve en la Fig.4, se insertan una cabecera UDP 403 – según se define en RFC768 – y una cabecera intermedia 404 después de la cabecera IP. La cabecera UDP tiene 8 octetos y la cabecera intermedia también tiene 8 octetos, dando un total de 16 octetos. En la explicación que sigue, se tratan ambas cabeceras como si fueran una sola. El formato más conveniente para la cabecera combinada es el mostrado en la Fig. 2c. Los campos en esta cabecera se fijan de la siguiente manera:

- El campo Puerto de Origen 221 se asigna al 500 (el mismo que IKE). Si el paquete va a través de una NAT, éste puede ser diferente cuando se recibe el paquete.

- El campo Puerto de Destino 222 se asigna al número de puerto desde el que el otro extremo parece estar enviando los paquetes. Si el paquete va a través de una NAT, el receptor puede ver aquí un número de puerto distinto.

- El campo Longitud UDP 223 es la longitud de la cabecera UDP más la longitud del campo de datos UDP. En este caso, también se incluye la cabecera intermedia. El valor se calcula en bytes como 16 más la longitud de la carga neta del paquete IP original (sin incluir la cabecera IP original, que se incluye en el campo Longitud de la cabecera IP).

- El campo Verificar-suma UDP 224 se fija preferiblemente en 0. El verificar-suma UDP es opcional, y no interesa calcularlo o comprobarlo con este mecanismo de entunelación. Se supone que la integridad de los datos está protegida por una cabecera AH o ESP dentro del paquete entunelado.

- El campo Cero obligatorio 255: Este campo debe contener un valor fijo acordado previamente, que es preferiblemente todo ceros. El campo se solapa con los cuatro primeros bytes del campo Cookie del Iniciador en una cabecera IKE real. Un Iniciador que soporte este aspecto de la invención no debe usar una cookie en la que los primeros cuatro bytes sean cero. Estos bytes cero se usan para separar los paquetes entunelados de los paquetes ISAKMP auténticos. Naturalmente se puede elegir algún otro valor fijado distinto de “todo ceros”, pero el valor debe fijarse para este uso particular.

- El campo Protocolo 226: El valor de este campo se copia del campo Protocolo ya conocido en la cabecera IP original (no mostrado separadamente en la Fig. 4)

- El campo Reservado 227: preferiblemente enviado como todo ceros; ignorado en la recepción.

El emisor inserta esta cabecera en cualquier paquete entunelado a un destino detrás de una NAT. La información sobre si se está usando una NAT se puede almacenar en el gestor de políticas según el criterio de SA (Asociación de la Seguridad). El encapsulado al que se refiere la Fig. 4 se puede ejecutar ya sea como una transformación nueva o como parte de las ya conocidas transformaciones AH y ESP.

La operación de encapsulado utiliza el número de puerto UDP y la dirección IP del anfitrión distante que se determinaron durante la negociación IKE.

El receptor desencapsula los paquetes de esta encapsulación antes de realizar el procesamiento AH o ESP. La desencapsulación elimina esta cabecera y actualiza los campos de Protocolo, Longitud y Verificar-suma de la cabecera IP. Para esta operación no se necesita ningún dato de configuración (número de puerto, etc.)

La desencapsulación sólo ha de ejecutarse si coinciden todos los siguientes selectores:

- La dirección de destino es la dirección de destino de este anfitrión,

- la dirección de origen es la dirección de origen de un anfitrión con el que este anfitrión ha acordado usar esta entunelación,

- el campo de Protocolo indica UDP,

- el valor del campo de puerto de Destino es 500 y

- el valor del campo de puerto de Origen indica el puerto con el que este anfitrión ha acordado usar esta entunelación. (Obsérvese que puede haber muchas direcciones de origen y puertos a los que se aplique esta entunelación; cada uno de ellos se trata con un juego distinto de selectores).

Durante la desencapsulación se puede sustituir la dirección de origen del paquete recibido por la dirección de origen auténtica recibida durante la negociación IKE. Esto aplica la compensación para la verificación de AH MAC. En la fase post-tratamiento que sigue se vuelve a cambiar la dirección. Gracias a esta compensación se pueden usar las conversiones estándar AH y ESP sin modificación alguna.

En la Fig. 3 se muestra esquemáticamente como bloque 301 el procesamiento AH/ESP en el nodo emisor, el bloque 302 muestra esquemáticamente la encapsulación de datagramas a UDP, el bloque 303 muestra esquemáticamente la desencapsulación de datagramas desde UDP y el bloque 304 muestra esquemáticamente el procesamiento AH/ESP en el nodo receptor.

Después de que el paquete se haya desencapsulado desde AH o EPS se debe aplicar una compensación adicional. Esta desencapsulación adicional debe tratar el hecho de que el paquete saliente atravesó realmente una NAT (ilustrado esquemáticamente en el bloque 305 de la Fig. 3) y en consecuencia el paquete de texto común también debe sufrir una transformación similar. El receptor debe ver la dirección del dispositivo NAT como la dirección del anfitrión en vez de como la dirección interna original. Alternativamente, el emisor del paquete podría haber realizado esta compensación antes de encapsularlo dentro de AH o ESP.

Existen varias alternativas para esta compensación adicional según los diversos casos especiales (la mejor compensación depende de cada aplicación particular):

- Asignar un intervalo de direcciones de red para este procesamiento (es decir, usar el intervalo 169.254.x.x. para el enlace local. – no importan los valores reales, lo que se desea es fundamentalmente una red arbitraria que no esté usando nadie más). Se asigna una dirección de este intervalo para cada combinación <natip, ownip, natport, ownport>, donde natip significa dirección IP de la NAT, ownip la dirección IP propia del dispositivo de tratamiento, natport significa el número de puerto en la NAT y ownport quiere decir el número del puerto del propio equipo del procesamiento. La dirección distante del paquete se substituye por esta dirección antes de que el paquete sea enviado a las pilas de protocolo.

- El verificar-suma TCP para equipos internos se debe recalcular, como parte de la compensación, en caso de que hayan cambiado las direcciones del anfitrión o los números de puerto. Los cálculos de verificar-suma TCP pueden ser incrementales tal y como se conoce a partir de RFC1071. Puede ser necesario aplicar el puerto NAT para el puerto de origen.

- Cuando se utilice como VPN entre dos sitios que usen espacios de direcciones privadas incompatibles (posible solapamiento), se debe aplicar la conversión de direcciones para compatibilizar dichas direcciones con las direcciones locales.

- Cuando se utilice como VPN entre dos sitios que usen espacios de direcciones privada compatibles (no solapamiento), y se aplica un modo de túnel, puede no ser necesaria una compensación adicional.

- Puede resultar necesario realizar conversión de direcciones para los contenidos de paquetes de ciertos protocolos, tales como FTP (divulgado por RFC959) o H.323. También se analizan otros temas parecidos en la referencia dada como HoldregeSrisuresh99.

- También es posible usar en el servidor direcciones aleatorias para el cliente, y aplicar conversión de direcciones a esta dirección. Esto podría permitir al servidor distinguir entre múltiples clientes situados tras la misma NAT, y podría evitar la configuración manual del espacio de dirección local.

- La operación de compensación puede interactuar o no con la pila TCP/IP en la máquina local para reservar números de puerto UDP.

En general, esta invención no limita el método usado para compensar en los paquetes internos la NAT que tenga lugar en la cabecera externa. El procedimiento óptimo para realizar dicha compensación puede encontrase por experimentación entre las alternativas presentadas anteriormente, o podría presentarse otro procedimiento óptimo.

A continuación se hará referencia al aspecto “mantener-activo” del invento, es decir asegurar que las conversiones de direcciones de red realizadas en la red no se modifican después de que se hayan determinado las conversiones

que tienen lugar. Los conversores de direcciones de red guardan en memoria caché la información de puesta en correspondencia de las direcciones, de modo que pueden invertir la puesta en correspondencia para los paquetes de respuesta. Si se usa TCP, el conversor de direcciones puede mirar en el bit FIN de la cabecera TCP para determinar cuándo puede desaparecer una determinada puesta en correspondencia. Sin embargo, para UDP no hay indicación explicita de finalización de flujos. Por esta razón, muchas NAT limitan bastante el tiempo de espera de puestas en correspondencia para UDP (incluso tanto como 30 segundos). Por tanto, se hace necesario forzar el mantenimiento de la puesta en correspondencia.

Una forma posible de asegurar el mantenimiento de las puestas en correspondencia consiste en enviar paquetes de activación con frecuencia suficiente para que la conversión de direcciones permanezca en la memoria caché. Al calcular la frecuencia necesaria debe tenerse en cuenta que los paquetes pueden perderse en la red, y en consecuencia deberán enviarse muchos paquetes de mantenimiento en activo dentro del intervalo más corto en que se haya estimado que las NAT pueden olvidar la puesta en correspondencia. La frecuencia apropiada depende tanto del período de tiempo en el que la puesta en correspondencia permanece memorizada en caché como de la probabilidad de pérdida de paquetes en la red; se puede llegar a los valores óptimos de frecuencia para cada situación mediante ensayos en cada una de ellas.

Los paquetes de mantenimiento en activo no necesitan contener más información significativa que las cabeceras necesarias que son iguales a las cabeceras de paquetes de datos para asegurar que los paquetes de mantenimiento en activo sean gestionados exactamente de la misma manera que los paquetes de datos reales. Un paquete de mantenimiento en activo puede contener un indicador que lo identifique como paquete de mantenimiento en activo y no como paquete de datos; sin embargo también se puede determinar que todos los paquetes que no contengan una información significativa de carga neta se interpreten como paquetes de mantenimiento en activo. En la FIG. 3 se ilustra esquemáticamente en el bloque 306 la transmisión de paquetes de mantenimiento en activo y en el bloque 307 se ilustra esquemáticamente la recepción y supresión de los mismos. Debe advertirse que el uso de paquetes de mantenimiento en activo no es en absoluto necesario si los paquetes de datos reales se transmiten con la frecuencia necesaria y/o la conexión se mantiene válida sólo durante un espacio de tiempo tan corto (por ejemplo unos pocos segundos) que haga improbable que un equipo intermedio pueda borrar la información de puesta en correspondencia de su memoria caché. Sólo es necesario transmitir los paquetes de mantenimiento en activo en una única dirección, si bien también se pueden transmitir bidireccionalmente; el inconveniente derivado de la transmisión bidireccional es el aumento de tráfico innecesario en la red. La invención no limita la dirección o direcciones en que los paquetes de mantenimiento en activo (si los hay) son transmitidos.

La Fig. 5 es un diagrama de bloques simplificado de un dispositivo de red 500 que puede actuar como Iniciador o Respondedor de acuerdo con el procedimiento de proporcionar comunicaciones seguras sobre conversiones de direcciones de red según la invención. La interfaz de red 501 conecta físicamente el dispositivo de red 500 a la red. El bloque 502 de gestión de direcciones mantiene bajo control las correctas direcciones de red, número de puerto y otra información de identificación pública esencial tanto del dispositivo de red 500 en sí mismo como de su pareja (no mostrada). El bloque IKE 503 es responsable del proceso de gestión de claves y de otras actividades relacionadas con el intercambio de información secreta. El bloque 504 encriptación/desencriptación ejecuta la encriptación y la desencriptación de datos una vez que el bloque IKE 503 ha obtenido la clave secreta.

El bloque 505 de compensación se usa para compensar las transformaciones permisibles en los paquetes transmitidos y/o recibidos de acuerdo con la invención. Cualquiera de los bloques 504 y 505 puede usarse para transmitir, recibir y desechar paquetes de mantenimiento en activo. El bloque 506 ensamblador/desensamblador es el que interviene entre los bloques 502 a 505 y la interfaz física 501 de la red. Todos los bloques operan bajo la supervisión de un bloque de control 507 que también se encarga de encaminar la información entre los otros bloques y el resto del dispositivo de la red, por ejemplo para mostrar la información al usuario a través de una unidad de visualización (no mostrada) y para obtener órdenes del usuario a través de un teclado (no mostrado). Los bloques de la Fig. 5 se implementan preferiblemente como procedimientos operativos previamente programados de un microprocesador, cuya ejecución práctica es conocida como tal por un experto en la materia. También se pueden usar en la práctica de esta invención otras disposiciones distintas a las mostradas en la Fig. 5.

Incluso aunque la presente invención se ha expuesto en el contexto de IKE, y entunelación usando puerto IKE, debe entenderse que puede aplicarse a otros casos análogos que usen diferentes métodos de formateado de paquetes, diferentes detalles de negociación, diferente protocolo de intercambio de claves, o diferente protocolo de seguridad. La invención también puede aplicarse a protocolos no IP que tengan las características adecuadas. La invención es igualmente aplicable a ambos protocolos IPv4 e IPv6. También se pretende que la invención sea aplicable a futuras revisiones de los protocolos IPSEC e IKE.

Del mismo modo se ha de entender que la invención es también aplicable a las conversiones de protocolos, además de a las conversiones de direcciones. Para un experto ha de ser fácil adaptar la presente invención a las conversiones de protocolo a partir de la presente descripción y las explicaciones sobre conversión de protocolo contenidas en solicitudes de patentes presentadas anteriormente por este mismo solicitante.

1. LISTA DE REFERENCIAS

BorellaLo99

M. Borella, J. Lo: Realm Specific IP: Protocol Specification, draft-ietf-nat-rsip-protocol-00.txt, Work in Progress, Internet Engineering Task Force, 1999

HoldregeSrisuresh99

M. Holdrege, P. Srisuresh: Protocol Complications with the IP Network Address Translator (NAT), draft-ietf-nat-protocol-complications-00.txt, Work in Progress, Internet Engineering Task Force, 1999.

LoBorella99

J. Lo, M. Borella: Real Specif IP: A Framework, draft-ietf-nat-rsip-framework-00.txt, Work in Progress, Internet Engineering Task Force, 1999

Rekhter99

Y. Rekhter: Implications of NATs on the TCP/IP architecture, draft-ietf-arch-implications-00.txt, Internet Engineering Task Force, 1999.

RFC768

J. Postel: User Datagram Protocol, RFC 768, Internet Engineering Task Force, 1980.

RFC791

J. Postel: Internet Protocol, RFC 791, Internet Engineering Task Force, 1981

RFC793

J. Postel: Transmission Control Protocol, RFC 793, Internet Engineering Task Force, 1981.

RFC959

J. Postel, J.Reynolds: File Transfer Protocol, RFC 959, Internet Engineering Task, 1985.

RFC1071

R. Braden, D. Borman, C. Partridge: Computing the Internet checksum, RFC 1071, Internet Engineering Task Force, 1988.

RFC1226

B. Kantor: Internet protocol encapsulation of AX.25 frames, RFC 1226, Internet Engineering Task Force, 1991.

RFC1234

D. Provan: Tunneling IPX traffic through IP networks, RFC 1234, Internet Engineering Task Force, 1991.

RFC1241

R. Woodburn, D. Mills: Scheme for an Internet encapsulation protocol: Version 1, RFC 1241 Internet Engineering Task Force, 1991.

RFC1321

R. Rivet: The MD5 message-digest algorithm, RFC 1321, Internet Engineering Task Force, 1992.

RFC1236

P. Tsuchiya: Mutual Encapsulation Considered Dangerous, RFC 1326, Internet Engineering Task Force, 1992.

RFC1631

K. Egevang, P. Francis: The IP Network Address Translator (NAT), RFC 1631, Internet Engineering Task Force, 1994.

RFC1701

S. Hanks, T. Li, D. Farinacci, P. Traina: Generic Routing Encapsulation, RFC 1701, Internet Engineering Task Force, 1994.

RFC1702

S. Hanks, T. Li, D. Farinacci, P. Traina: Generic Routing Encapsulation over IPv4 networks, RFC 1702, Internet Engineering Task Force, 1994.

RFC1853

W. Simpson: IP in IP Tunneling, RFC 1853, Internet Engineering Task Force, 1995.

RFC2003

C. Perkins: IP Encapsulation within IP, RFC 2003, Internet Engineering Task Force, 1996.

RFC2004

C. Perkins: IP Encapsulation within IP, RFC 2004, Internet Engineering Task Force, 1996.

RFC2107

K. Hamzeh: Ascend Tunnel Management Protocol, RFC 2107, Internet Engineering Task Force, 1997.

RFC2344

G. Montenegro: Reverse Tunneling for Mobile IP, RFC 2344, Internet Engineering Task Force, 1998.

RFC2391

P. Srisuresh, D. Gan: Load Sahring using IP Network Address Translation (LSNAT), RFC 2391, Internet Engineering Task Force, 1998.

RFC2401 1

S. Kent, R. Atkinson: Security Architecture for the Internet Protocol, RFC 2401, Internet Engineering Task Force, 1998.

RFC2402

S. Kent, R. Atkinson: IP Authentication Header, RFC 2401, Internet Engineering Task Force, 1998.

RFC2406

S. Kent, R. Atkinson: IP Encapsulating Security Carga neta, RFC 2406, Internet Engineering Task Force, 1998.

RFC2407

D. Piper: The Internet IP Security Domain of Interpretation for ISAKMP, RFC 2407, Internet Engineering Task Force, 1998.

RFC2408

D. Maughan, M. Schertler, M. Schneider, J. Turner: Internet Security Association and Key Management Protocol (ISAKMP), RFC 2408, Internet Engineering Task Force, 1998.

RFC2409

D. Hakins, D. Carrel: The Internet Key Exchange (IKE), RFC 2409, Internet Engineering Task Force, 1998.

RFC2473

A. Conta, S. Deering: Generic Packet Tunneling in IPv6 Specification, RFC 2473, Internet Engineering Task Force, 1998.

RFC2529

B. Carpenter, C. Jung: Transmission of IPv6 over IPv4 Domains without Explicit Tunnels, RFC 2529, Internet Engineering Task Force, 1999.

Srisuresh98Terminology

P. Srisuresh: IP Network Address Translator (NAT) Terminology and Considerations, draft-ietf-nat-terminology-01.txt, Work in Progress, Internet Engineering Task Force, 1998.

Srisuresh98Security

P. Srisuresh: Security Model for Network Address Translator (NAT) Domains, draft-ietf-nat-security-01.txt, Work in Progress, Internet Engineering Task Force, 1998.

SrisureshEgevang98

P. Srisuresh, K. Egevang: Traditional IP Network Address Translator (Traditional NAT), draft-ietf-nat-traditional-01.txt, Work in Progress, Internet Engineering Task Force, 1998.

TYS99

W. Teo, S. Yeow, R. Singh: IP Relocation through twice Network Address Translators (RAT), draft-ietf-nat-rnat-00.txt, Work in Progress, Internet Engineering Task Force, 1999.

REIVINDICACIONES

1. Procedimiento para mantener la comunicación de datagramas en un sistema de comunicación en el que la conversión de dirección es proporcionada por un conversor (305) de dirección de red para comunicación de datagramas entre un primer dispositivo y un segundo dispositivo, **caracterizado** por mantener una conversión de dirección de red determinada para comunicación de datagramas entre el primer dispositivo y el segundo dispositivo enviando (306) desde el primer dispositivo o el segundo dispositivo por lo menos un paquete de mantenimiento en activo antes de la finalización del intervalo de retardo de la conversión de dirección de red determinada.

2. Procedimiento para proporcionar conversiones de direcciones por un conversor (305) de dirección de red, que comprende determinar una conversión de dirección para comunicación de datagramas entre un primer dispositivo y un segundo dispositivo; **caracterizado** por recibir al menos un paquete de conservación en activo desde el primer dispositivo y/o el segundo dispositivo antes de la finalización del intervalo de retardo de la conversión de dirección determinada para la comunicación de datagramas; y en respuesta a la recepción de al menos un paquete de mantenimiento en activo, mantener la conversión de dirección de red determinada para la comunicación de datagramas entre el primer dispositivo y el segundo dispositivo.

3. Un procedimiento según la reivindicación 1 ó 2, en el que al menos un paquete de mantenimiento en activo comprende una cabecera que es igual a las cabeceras de los datagramas.

4. Un procedimiento según cualquier reivindicación precedente, en el que al menos un paquete de mantenimiento en activo contiene un indicador que lo identifica como un paquete de mantenimiento en activo.

5. Un procedimiento según cualquier reivindicación precedente, en el que un paquete es interpretado como un paquete de mantenimiento en activo si no contiene ninguna carga neta significativa.

6. Un procedimiento según la reivindicación 1 o cualquier reivindicación dependiente de la reivindicación 1, que comprende la determinación de un período de tiempo más corto para la finalización del intervalo de retardo, y basado en la determinación, enviando al menos el paquete de mantenimiento en activo lo bastante frecuentemente para mantener la conversión de dirección determinada en el conversor (305) de dirección de red.

7. Un procedimiento según la reivindicación 1 o las reivindicaciones dependientes de la reivindicación 1, que comprende tener en cuenta la posibilidad de pérdida de paquete para determinar la frecuencia de envío de al menos el paquete de mantenimiento en activo.

8. Un dispositivo (500) para comunicación de datagramas en un sistema de comunicación en el que el la conversión de dirección es proporcionada por un conversor (305) de dirección de red para comunicación de datagramas entre el dispositivo y un segundo dispositivo, **caracterizado** por medios (504 o 505) para mantener una conversión de dirección de red determinada para la comunicación de datagramas entre el dispositivo y el segundo dispositivo provocando el envío de al menos un paquete de mantenimiento en activo antes de la finalización del intervalo de retardo de la conversión de direcciones de red determinada.

9. Un dispositivo para conversiones de direcciones de red (305), que comprende medios para determinar una conversión de dirección para comunicación de datagramas entre un primer dispositivo y un segundo dispositivo; **caracterizado** por medios para mantener la conversión de dirección de red determinada para la comunicación de datagramas entre el primer dispositivo y el segundo dispositivo en respuesta a la recepción de al menos un paquete de mantenimiento en activo desde el primer dispositivo y/o el segundo dispositivo antes de la finalización del intervalo de retardo de la conversión de dirección determinada para la comunicación de datagramas.

10. Un dispositivo según la reivindicación 8 ó 9, en el que al menos un paquete de mantenimiento en activo comprende una cabecera que es igual a las cabeceras de los datagramas.

11. Un dispositivo según cualquiera de las reivindicaciones 8 a 10, en el que al menos un paquete de mantenimiento en activo contiene un identificador que lo identifica como un paquete de mantenimiento en activo.

12. Un dispositivo según cualquiera de las reivindicaciones 8 a 11, en el que el dispositivo está configurado para interpretar un paquete como un paquete de mantenimiento en activo si el paquete no contiene ninguna carga neta significativa.

13. Un dispositivo (500) según la reivindicación 8 o cualquier reivindicación dependiente de la reivindicación 8, en el que los medios para mantenimiento están configurados para provocar el envío de al menos un paquete de mantenimiento en activo lo bastante frecuentemente para mantener la conversión de dirección determinada.

14. Un dispositivo (500) según la reivindicación 8 o cualquier reivindicación dependiente de la reivindicación 8, en el que los medios para mantenimiento están configurados para tener en cuenta la posibilidad de la pérdida de un paquete en la determinación de la frecuencia de envío.

15. Un programa de ordenador que comprende medios de código de programa adaptados para realizar cualquiera de las etapas de cualquiera de las reivindicaciones 1 a 7 cuando el programa se ejecutado en un procesador.

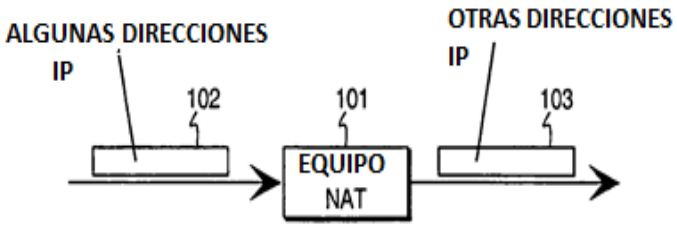

Fig. 1a
ESTADO DE LA TÉCNICA

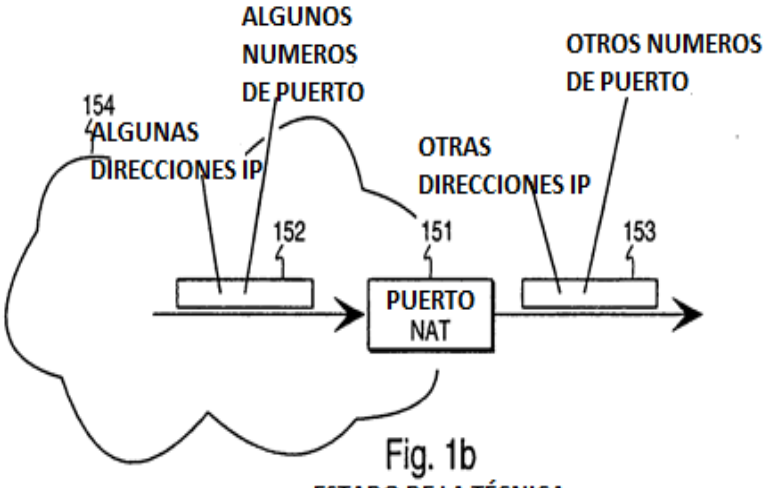

Fig. 1b
ESTADO DE LA TÉCNICA

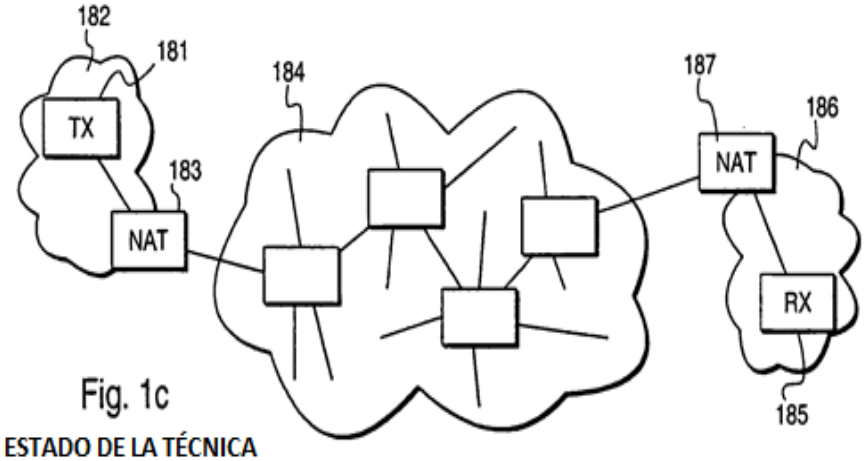

Fig. 1c
ESTADO DE LA TÉCNICA

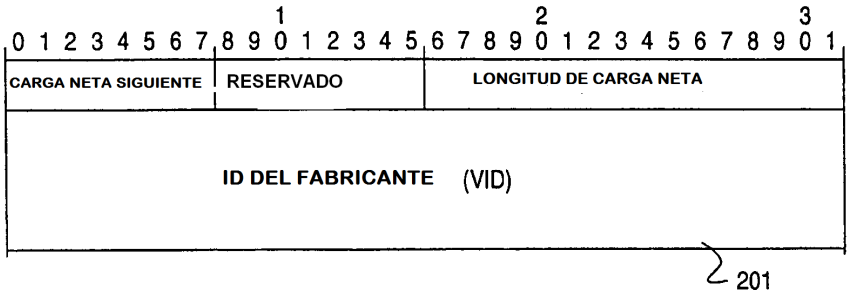

Fig. 2a

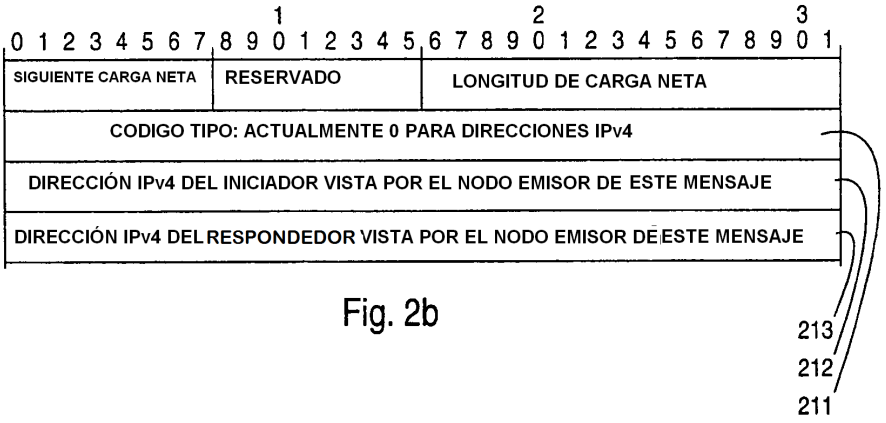

Fig. 2b

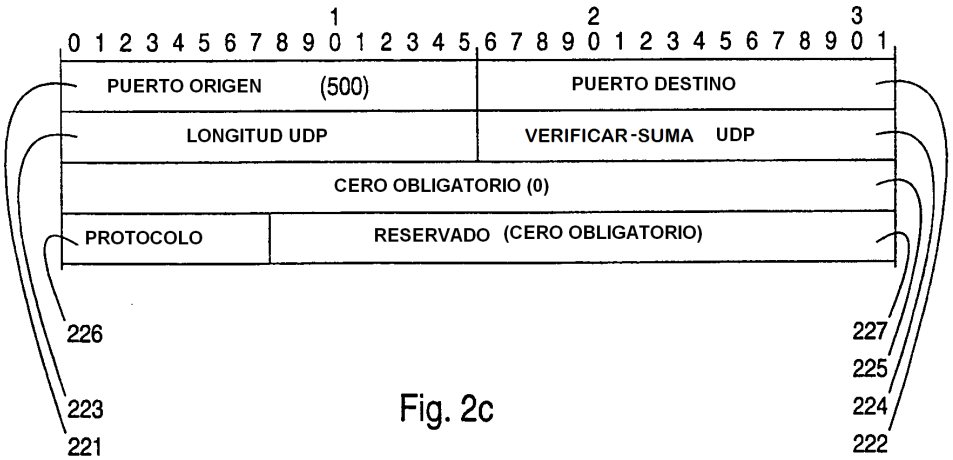

Fig. 2c

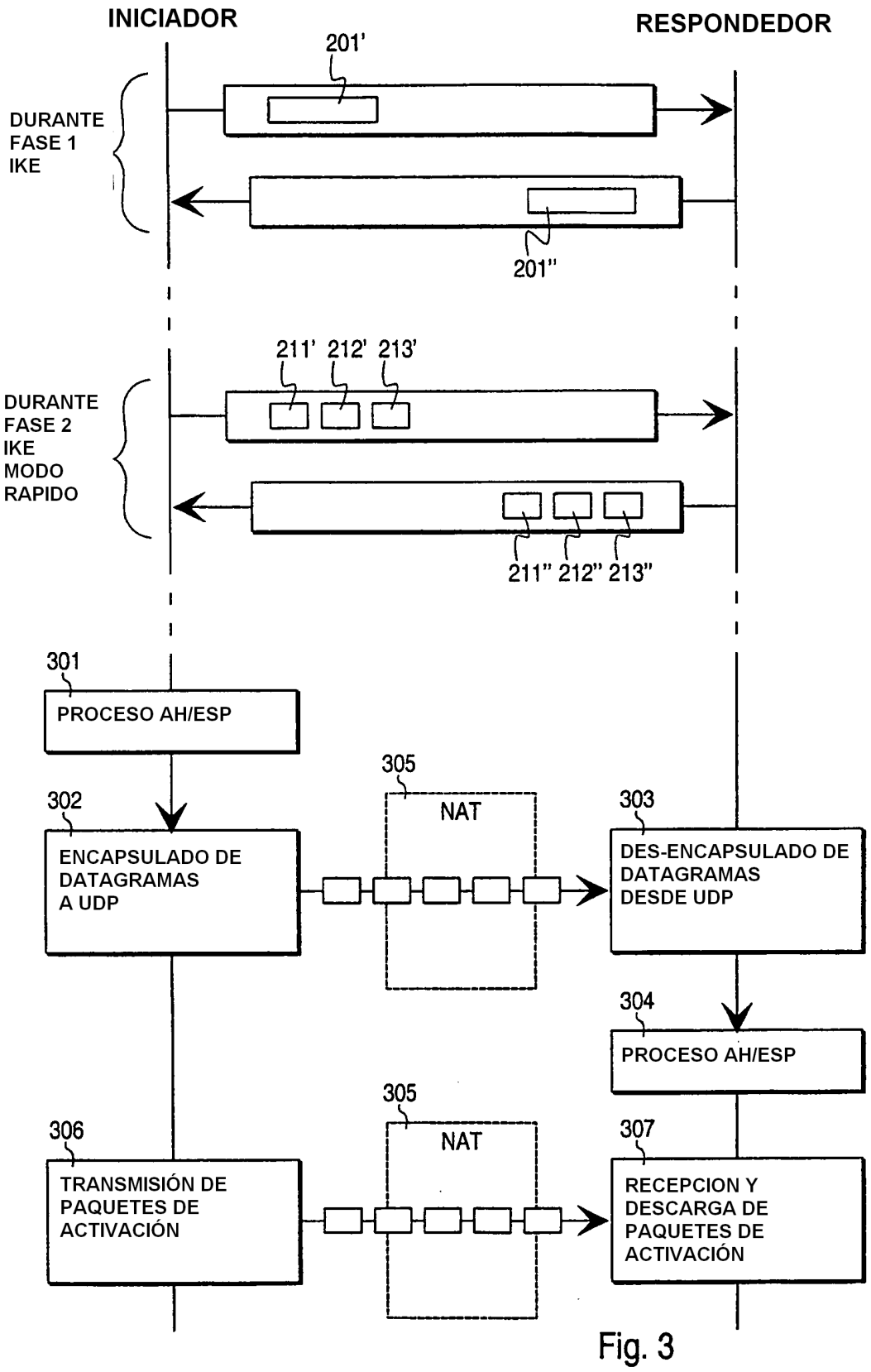
INICIADOR
RESPONDEDOR
201'
DURANTE FASE 1 IKE
201"
211' 212' 213'
DURANTE FASE 2 IKE MODO RAPIDO
211" 212" 213"
301
PROCESO AH/ESP
305
NAT
302
ENCAPSULADO DE DATAGRAMAS A UDP
303
DES-ENCAPSULADO DE DATAGRAMAS DESDE UDP
304
PROCESO AH/ESP
305
NAT
306
TRANSMISIÓN DE PAQUETES DE ACTIVACIÓN
307
RECEPCION Y DESCARGA DE PAQUETES DE ACTIVACIÓN

Fig. 3

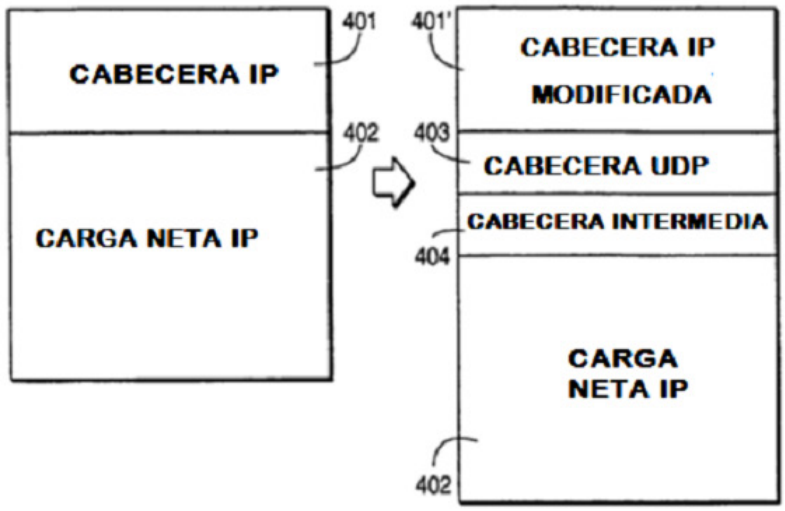
CABECERA IP
CARGA NETA IP
401
402
401'
CABECERA IP
MODIFICADA
403
CABECERA UDP
CABECERA INTERMEDIA
404
CARGA
NETA IP
402

Fig. 4

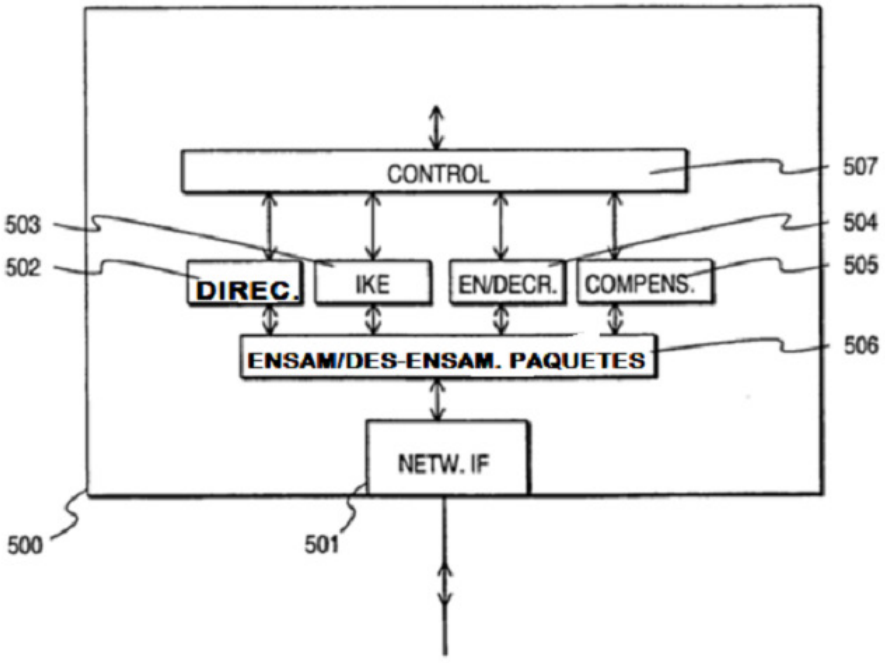
CONTROL
507
503
504
502
DIREC.
IKE
EN/DECR.
COMPENS.
505
ENSAM/DES-ENSAM. PAQUETES
506
NETW. IF
500
501

Fig. 5