(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 964 955**

(51) Int. Cl.:

H04L 9/40 (2012.01)
H04L 9/32 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **08.05.2018 PCT/US2018/031641**

(87) Fecha y número de publicación internacional: **15.11.2018 WO18208809**

(96) Fecha de presentación y número de la solicitud europea: **08.05.2018 E 18797606 (3)**

(97) Fecha y número de publicación de la concesión europea: **01.11.2023 EP 3622699**

(54) Título: **Métodos de intercambio de paquetes bidireccional a través de vías nodales**

(30) Prioridad:

09.05.2017 US 201762503808 P
26.06.2017 US 201762524705 P
21.07.2017 US 201715656454

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
10.04.2024

(73) Titular/es:

NETWORK NEXT, INC. (100.0%)
603 Arizona Avenue
Santa Monica, California 90401, US

(72) Inventor/es:

FIELDER, GLENN

(74) Agente/Representante:

ISERN JARA, Jorge

DESCRIPCIÓN

Métodos de intercambio de paquetes bidireccional a través de vías nodales

Campo de la divulgación

El campo de la divulgación son comunicaciones en red.

Antecedentes

La descripción de antecedentes incluye información que puede ser útil para entender la presente invención. No es una admisión de que cualquiera de la información proporcionada en esta solicitud sea técnica anterior o relevante para la invención actualmente reivindicada, o que cualquier publicación a la que se haga referencia específica o implícitamente sea técnica anterior.

Los juegos multijugador en tiempo real en general operan enviando paquetes desordenados poco fiables a través del Internet, por ejemplo, como paquetes UDP, en un patrón de flujo bidireccional donde los paquetes se envían en ambas direcciones, desde cliente a servidor y servidor a cliente, a alguna tasa como 10, 20, 30 o 60 paquetes por segundo.

Los paquetes intercambiados entre el cliente y servidor son extremadamente sensibles a la latencia, variación rápida, y/o pérdida de paquetes. Conocido colectivamente como calidad de servicio o "QoS".

En general, los clientes se conectan a servidores dedicados enviando y recibiendo paquetes directamente a la dirección de IP del servidor, pero esto deja a los servidores dedicados vulnerables a ataques DDoS debido a que la dirección de IP del servidor está expuesta.

También, cuando los paquetes se envían a través del Internet público, la ruta que toman los paquetes entre el cliente y servidor no está bajo el control directo del cliente o servidor. Los paquetes hacen tomar una ruta que es más barata, en lugar de una ruta que optimiza QoS.

De manera similar, mientras los paquetes se intercambian a través del Internet, si la ruta que toman los paquetes entre un cliente y servidor se degrada, o hay una ruta mejor disponible, el cliente o servidor no tienen forma de ajustar la ruta que toman los paquetes entre el cliente y el servidor.

De este modo, existe una necesidad de métodos mejorados de conexión de clientes con servidores dedicados que no expongan la dirección de IP del servidor y proporcionen cierto grado de control sobre la ruta tomada por los paquetes entre el cliente y servidor.

El documento US 2016/0248686 (Lee et al.) divulga un dispositivo que establece flujos asociados con una o más aplicaciones usando señalización de plano de control. Un dispositivo de puerta de acceso obtiene una solicitud de una señal de red durante la señalización de plano de control. El dispositivo de puerta de acceso deriva la señal de red y la envía al dispositivo y/o a un nodo de acceso durante la señalización de plano de control. El dispositivo y/o nodo de acceso obtienen la señal de red, donde la señal de red está asociada con un primer flujo del uno o más flujos, una primera aplicación de la una o más aplicaciones, y aprovisionado al dispositivo o nodo de acceso a través de la señalización de plano de control. La señal de red puede incluirse en un paquete enviado en el plano de usuario desde el dispositivo. La señal de red puede verificarse en el nodo de acceso y/o el dispositivo de puerta de acceso usando una función criptográfica y enviarse a su destino con base en los resultados de la verificación.

"PSB NSLP: Network Traffic Authorization; draft-hong-nsis-pbs-nslp-04.txt", IETF, 14 de octubre de 2013, páginas 1-28, describe un protocolo de capa de señalización NSIS (NSLP) para autorización de tráfico de red en el Internet, cuyo objetivo es evitar ataques DoS y otras formas de tráfico no autorizado. La señalización instala y mantiene el estado de permiso de enrutadores para un flujo de datos.

El documento US 2011/0185039 (Ueno et al.) divulga un dispositivo de control de acceso que hace que un dispositivo de procesamiento de información ejecute control de acceso entre un dispositivo cliente y dos o más dispositivos servidores.

El documento US 2015/0326542 (Serebrin) divulga la migración en vivo de un invitado en una máquina virtual de un servidor de anfitrión a un servidor de destino, incluyendo cómo un servidor de anfitrión puede utilizar una clave de flujo para encriptar y desencriptar comunicaciones con un servidor de destino.

El documento WO 2007/035655 (Columbia University et al.) divulga sistemas y métodos para inhibir ataques con una red reenviando paquetes a través de una pluralidad de nodos intermedios.

Resumen de la invención

Desde un primer aspecto, la invención proporciona un sistema de nodos como se reivindica en la reivindicación 1. Desde un segundo aspecto, la invención proporciona un medio legible por ordenador no transitorio como se reivindica en la reivindicación 11. Desde un tercer aspecto, la invención proporciona un sistema de servidor maestro como se reivindica en la reivindicación 13.

Breve descripción del dibujo

La figura 1 ilustra servidores dedicados que reportan información al adaptador.

La figura 2 ilustra relés que reportan información al servidor maestro.

La figura 3 ilustra un cliente que solicita conectarse a un servidor dedicado.

La figura 4 ilustra un servidor maestro que retorna un arreglo de rutas de flujo al cliente.

La figura 5A ilustra una ruta de flujo.

La figura 5B ilustra una señal de flujo.

La figura 6 ilustra un cliente que envía un paquete de solicitud a un servidor dedicado.

La figura 7 ilustra un paquete de respuesta enviado a un cliente en respuesta a un paquete de solicitud recibido desde ese cliente.

La figura 8A ilustra una caché del relé.

La figura 8B ilustra los datos de entrada en una caché del relé.

La figura 9A ilustra una caché del servidor.

La figura 9B ilustra los datos de señal en una caché del servidor.

La figura 10 ilustra un cliente que solicita una ruta de flujo actualizada.

La figura 11 ilustra un servidor maestro que envía una ruta de flujo actualizada a un cliente.

La figura 12 ilustra un paquete de solicitud para la ruta actualizada que pasa desde el cliente al servidor mientras se mantiene la ruta existente para paquetes de carga útil.

La figura 13 ilustra un paquete de respuesta que se envía a un cliente en respuesta a un paquete de solicitud actualizado que se recibe desde ese cliente.

La figura 14 ilustra una señal de sesión de acuerdo con aspectos de la presente divulgación.

La figura 15 ilustra un sistema que puede usarse para implementar un método de comunicación por relé de nodos de acuerdo con un aspecto de la presente divulgación.

Descripción detallada

La siguiente discusión proporciona realizaciones de ejemplo de la materia objeto inventiva. Aunque cada realización representa una única combinación de elementos inventivos, se considera que la materia objeto inventiva incluye todas las combinaciones posibles de los elementos divulgados. De este modo si una realización comprende elementos A, B, y C, y una segunda realización comprende elementos B y D, entonces también se considera que la materia objeto inventiva incluye otras combinaciones restantes de A, B, C, o D, incluso si no se divulgan explícitamente.

Como se usa en la descripción en esta solicitud y a lo largo de las reivindicaciones que siguen, el significado de "un", "uno, una" y "el, la" incluye referencia plural a menos que el contexto dicte claramente otra cosa. También, como se usa en la descripción en esta solicitud, el significado de "en" incluye "en" y "sobre" a menos que el contexto dicte claramente otra cosa.

También, como se usa en esta solicitud, y a menos que el contexto dicte otra cosa, el término "acoplado a" está previsto para incluir tanto acoplamiento directo (en el cual dos elementos que están acoplados entre sí entran en contacto entre sí) como acoplamiento indirecto (en el cual al menos un elemento adicional está ubicado entre los dos elementos). Por lo tanto, los términos "acoplado a" y "acoplado con" se usan como sinónimos.

En algunas realizaciones, los números que expresan cantidades de ingredientes, propiedades tales como concentración, condiciones de reacción, y así sucesivamente, usados para describir y reivindicar ciertas realizaciones de la invención deben entenderse como modificados en algunos casos por el término "aproximadamente". Por consiguiente, en algunas realizaciones, los parámetros numéricos establecidos en la descripción escrita y reivindicaciones adjuntas son aproximaciones que pueden variar dependiendo de las propiedades deseadas que se buscan obtener mediante una realización particular. En algunas realizaciones, los parámetros numéricos deben interpretarse a la luz del número de dígitos significativos reportados y aplicando técnicas de redondeo normales. A pesar de que los rangos numéricos y parámetros que establecen el amplio alcance de algunas realizaciones de la invención son aproximaciones, los valores numéricos establecidos en los ejemplos específicos se reportan tan precisamente como sea viable. Los valores numéricos presentados en algunas realizaciones de la invención pueden contener ciertos errores que necesariamente resultan de la desviación estándar encontrada en sus respectivas mediciones de prueba. Además, y a menos que el contexto dicte lo contrario, todos los rangos establecidos en esta solicitud deben interpretarse como inclusivos de sus puntos finales y los rangos abiertos deben interpretarse para incluir solo valores comercialmente prácticos. De manera similar, todas las listas de valores deben considerarse como inclusivas de valores intermedios a menos que el contexto indique lo contrario.

Debe anotarse que cualquier lenguaje dirigido a una ordenador debe leerse para incluir cualquier combinación adecuada de dispositivos informáticos, incluyendo servidores, interfaces, sistemas, bases de datos, agentes, pares, motores, controladores, u otros tipos de dispositivos informáticos que operen individual o colectivamente. Se debería apreciar que los dispositivos informáticos comprenden un procesador configurado para ejecutar instrucciones de software almacenadas en un medio de almacenamiento legible por ordenador tangible, no transitorio (por ejemplo, disco duro, unidad de estado sólido, RAM, flash, ROM, etc.). Las instrucciones de software configuran preferiblemente el dispositivo informático para proporcionar las funciones, responsabilidades, u otra funcionalidad como se discute a continuación con respecto al aparato divulgado. En realizaciones especialmente preferidas, los diversos servidores, sistemas, bases de datos, o interfaces intercambian datos usando protocolos o algoritmos estandarizados, posiblemente basados en HTTP, HTTPS, AES, intercambios de claves público-privadas, APIs de servicios web, protocolos de transacciones financieras conocidos, u otros métodos electrónicos de intercambio de información. Los intercambios de datos se llevan a cabo preferiblemente a través de una red conmutada por paquetes, el Internet, LAN, WAN, VPN, u otro tipo de red conmutada por paquetes. La siguiente descripción incluye información que puede ser útil para entender la presente invención. No es una admisión de que cualquiera de la información proporcionada en esta solicitud sea técnica anterior o relevante para la invención actualmente reivindicada, o que cualquier publicación a la que se haga referencia específica o implícitamente sea técnica anterior.

La materia objeto inventiva abarca sistemas y métodos de conexión de dos ordenadores a través de una ruta de flujo de tal manera que ninguno de los dos ordenadores tenga ninguna forma de conocer la dirección de IP de la otra. Se contempla que la materia objeto inventiva se pueda implementar en el área de juego en línea como una medida de protección para asegurar que ningún cliente (por ejemplo, un jugador) pueda conocer la dirección de IP del servidor dedicado (por ejemplo, el servidor alojado en juego).

Para asegurar que los clientes no puedan conocer la identidad o ubicación (por ejemplo, dirección de IP y puerto) de un servidor, se puede implementar al menos un relé como un intermediario para facilitar el intercambio de paquetes. Al tener un relé posicionado en medio del cliente y el servidor, el cliente solo necesita conocer que debe enviar paquetes al relé, y el relé a su vez conoce que recibe paquetes desde el cliente y envía paquetes al servidor. El servidor, de la misma manera, solo conoce que recibe paquetes desde el relé y a su vez envía paquetes al relé.

Puede ser ventajoso incluir relés adicionales. En sistemas que incluyen más de un relé, relés, el cliente, y el servidor pueden denominarse "nodos". El objetivo final es permitir el intercambio de paquetes entre un cliente y un servidor a través de una ruta de flujo de tal forma que el cliente nunca conozca la dirección de IP y puerto del servidor mientras que también optimiza la ruta de acuerdo con alguna métrica.

Más específicamente, realizaciones de la materia objeto inventiva proporcionan rutas optimizadas entre clientes y servidores dedicados, fijando la ruta a "relés" intermedios a través del Internet público. Las rutas se pueden optimizar para, por ejemplo, reducir latencia, reducir pérdida de paquetes, o mejorar cualquier otra métrica de QoS (calidad de servicio) según se desee. En tanto que existan múltiples rutas de relé entre cliente y servidor y cada ruta de relé tenga características diferentes, se puede seleccionar la mejor ruta. Esto es análogo al software de búsqueda de rutas tal como Google Maps, Apple Maps, Waze, etc. ya que el resultado final deseado es seleccionar y establecer la ruta más rápida a un destino.

Realizaciones de la materia objeto inventiva también proporcionan protección DDoS al ocultar la dirección de IP de un servidor dedicado de los clientes que están en comunicación con este. Esto hace imposible atacar el servidor dedicado en un ataque DDoS tradicional. Realizaciones también proporcionan la capacidad de cambiar rutas dinámicamente mientras un cliente continúa intercambiando paquetes con un servidor dedicado. Por ejemplo, si una ruta mejor está disponible, o si la ruta actual tiene un relé en el camino que está bajo ataque DDoS, entonces al cambiar dinámicamente la ruta sin cesar el intercambio de paquetes entre el cliente y el servidor a través de la ruta existente, una sesión del cliente (por ejemplo, una sesión de juego) puede continuar ininterrumpida en un servidor dedicado, incluso aunque haya ajustado dinámicamente su ruta.

Realizaciones de la materia objeto inventiva también mejoran la seguridad. Los terceros maliciosos son incapaces de secuestrar relés de la materia objeto inventiva para enviar paquetes a través de ellas. La materia objeto inventiva hace que sea trivial para un sistema rechazar paquetes que no se originan desde un cliente o servidor válido.

Las figuras 1 y 2 muestran varias operaciones de sondeo en segundo plano. Periódicamente (por ejemplo, a intervalos regulares o irregulares), los servidores dedicados $\{s_1,...,s_j\}$ (por ejemplo, un servidor de juego dedicado que es una versión sin cabezal del juego que se ejecuta en un centro de datos tal como una nube privada (por ejemplo, un centro de datos, o "*bare metal*"), o una nube pública tal como Google Compute, Amazon EC2, o Microsoft Azure) reportan sus direcciones de IP, puertos, y claves públicas al adaptador. La figura 1 muestra los servidores dedicados s_1 $s_2...s_j$ que reportan 104 sus direcciones de IP y puertos junto con sus claves públicas de vuelta al adaptador 101. El reporte se produce periódicamente (por ejemplo, a intervalos regulares o irregulares). Por ejemplo, cada servidor dedicado s_1 $s_2...s_j$ puede reportar 104 su dirección de IP y puerto al adaptador 101 una vez cada 1-5 minutos. También se contempla que los servidores dedicados s_1 $s_2...s_j$ puedan reportar al adaptador 101 en otros intervalos incluyendo cada 1-30 segundos, 30-59 segundos, o incluso múltiples veces por segundo (por ejemplo, 2-10 Hz). El reporte 104 periódico permite una arquitectura de microservicios optimizada basada alrededor de una cola para manejar un gran número de servidores dedicados.

El adaptador 101 mantiene esta lista ,y la actualiza según sea necesario (por ejemplo, si un servidor dedicado s_1 $s_2...s_j$ deja de reportar, el adaptador 101 saca ese servidor dedicado de su lista mantenida, o si un nuevo servidor dedicado reporta con una nueva dirección de IP y puerto, el adaptador agrega esa información a su base de datos). Los datos por servidor dedicado contienen como mínimo, la dirección de IP, puerto, y clave pública para cada servidor dedicado, pero también pueden incluir otros criterios útiles para determinar cuáles servidores dedicados satisfacen mejor la solicitud de un cliente (por ejemplo, número de versión de juego, número de jugadores actualmente conectados al servidor, jugadores totales permitidos para conectarse al servidor, región en la que se ubica el servidor, modo de juego que el servidor está ejecutando actualmente, por ejemplo, "CTF" o "Deathmatch", el nivel de experiencia de los jugadores actualmente conectados al servidor , etc.).

El adaptador 101 puede ser operado, por ejemplo, por una empresa de videojuegos. El adaptador 101 tiene alguna autentificación que le permite comunicarse con el servidor 102 maestro, al cual de otro modo no es accesible públicamente. La función del adaptador 101 es tomar la solicitud de un cliente 103 para jugar el juego, y encontrar el conjunto de direcciones de IP y puertos de servidor dedicados para que el cliente se conecte y que satisfagan la solicitud de ese cliente. Esto podría ser por ejemplo, servidores que ejecutan el mismo modo de juego que el cliente solicitó o servidores en la misma región que el cliente 103, con el mismo número de versión de juego y un conjunto de jugadores de experiencia similar a la del jugador cliente, o cualquier otro criterio.

Para propósitos de esta solicitud, una "ruta de flujo" es una vía nodal que vincula un cliente a un servidor. Un "flujo" describe los paquetes que se intercambian a través de una "ruta de flujo" una vez se establece.

La figura 2 muestra los relés r_1, $r_2...r_i$ reportando sus direcciones de IP y puertos junto con sus claves públicas al servidor maestro. El servidor maestro realiza 102 las mismas funciones que el adaptador 101 en esta capacidad: almacena las direcciones de IP y puertos junto con las claves públicas para los relés r_1, $r_2...r_i$, y actualiza esta información según sea necesario. Al igual que con el adaptador 101, el reporte 201 se produce periódicamente (por ejemplo, a intervalos regulares o irregulares). Por ejemplo, cada relé r_1, $r_2...r_i$ puede reportar 201 su dirección de IP y puerto al servidor maestro una vez cada 1-5 minutos. También se contempla que los relés puedan reportar 201 al servidor 102 maestro en otros intervalos que incluyen cada 1-30 segundos, 30-59 segundos, o incluso múltiples veces por segundo (por ejemplo, 2-10 Hz). El reporte 201 periódico permite una arquitectura de microservicios optimizada basada alrededor de una cola para manejar un gran número de relés.

Se contempla adicionalmente que los relés r_1, $r_2...r_i$ puedan autentificarse con un servidor 102 maestro para garantizar que los relés no autorizados no sean capaces de registrarse por sí mismos con el servidor 102 maestro.

Como mínimo, los datos por relé incluyen la dirección de IP + puerto y clave pública de ese relé, pero también pueden contener información adicional que se puede usar en la creación de rutas de flujo que se optimizan con base en diferentes criterios (por ejemplo, longitud/latitud de cada relé, relés cercanos, tiempos de viaje de ida y vuelta medidos actuales a relés cercanos, etc.).

La función del servidor 102 maestro es generar rutas de flujo entre dos puntos finales (por ejemplo, una vía desde un cliente a un servidor dedicado a través de una serie de relés). Las vías nodales se pueden identificar algorítmicamente en un esfuerzo por identificar una ruta de flujo que se optimice con base en uno o más factores (por ejemplo, para minimizar latencia, minimizar pérdida de paquetes, minimizar variación rápida, o cualquier combinación de las mismas). El servidor 102 maestro está disponible para que el adaptador 101 lo consulte usando, por ejemplo, una REST API.

Antes de discutir el proceso de establecimiento de un flujo, es importante introducir los diferentes tipos de paquetes que incorporan realizaciones de la materia objeto inventiva. Los paquetes enviados a través de la red en realizaciones

de la materia objeto inventiva tienen el prefijo de un byte que identifica el tipo de paquete. Hay cuatro tipos de paquetes: 0, 1, 2, y 3. El tipo de paquete 0 indica un paquete de solicitud. El tipo de paquete 0 tiene la forma [0][señal de flujo 0, señal de flujo 1,..., señal de flujo n-1] y corresponde a la estructura de datos de ruta de flujo prefijada por un byte cero. El tipo de paquete 1 indica un paquete de respuesta. El tipo de paquete 2 indica un paquete de carga útil que pasa desde el cliente al servidor. El tipo de paquete 3 indica un paquete de carga útil que pasa desde el servidor al cliente. Los números de secuencia de paquetes solo se aplican a los paquetes de respuesta y a paquetes de carga útil. El tipo de paquete 1 tiene la forma: [1][secuencia de paquete][ID de flujo][versión de flujo][hmac], mientras que los tipos de paquete 2 y 3 tienen la forma: [1,2 o 3][secuencia de paquete][ID de flujo][versión de flujo][hmac](datos de carga útil).

Estados de cliente

Para comenzar, un cliente puede existir en varios estados:

FLOW_CLIENT_STATE_INVALID_FLOW_ ROUTE	2
FLOW_CLIENT_STATE_TIMED_OUT	1
FLOW_CLIENT_STATE_STOPPED	
FLOW_CLIENT_STATE_REQUESTED	
FLOW_CLIENT_STATE_ESTABLISHED	

Los clientes comienzan en el estado "detenido" (estado 0), y cuando un usuario necesita establecer un flujo, el usuario pasa la ruta de flujo al cliente. Luego el cliente intenta desencriptar la primera señal de flujo en la ruta de flujo con su clave privada, y la clave pública del servidor maestro (que conoce). Si la señal de flujo falla en desencriptar, ha expirado, o no es válida por alguna razón, el cliente pasa al estado de ruta de flujo no válido (estado -2). De lo contrario, el cliente pasa al estado "solicitante" (estado 1). Mientras está en este estado, el cliente envía paquetes de solicitud a alguna frecuencia (por ejemplo, 10 Hz) al primer relé. Si mientras está en el estado "solicitante", el cliente recibe un "paquete de respuesta" desde el primer relé, el cliente hace transición al estado "establecido" (estado 2). Mientras está en el "estado establecido" el cliente deja de enviar "paquetes de solicitud". Si el cliente mientras está en los estados "solicitante" o "establecido" no recibe un paquete desde el primer relé durante alguna cantidad de tiempo (por ejemplo, 1-10 segundos), se expira y pasa al estado "expirado".

Si el cliente está en el estado "solicitado" o el estado "establecido", un usuario puede enviar paquetes de carga útil desde el cliente al servidor y potencialmente recibir cualquier paquete de carga útil enviado desde el servidor al cliente. Esto permite al cliente enviar paquetes de carga útil de manera optimista antes de que el flujo se haya confirmado como completamente establecido. Además, cuando el cliente envía paquetes de carga útil hacia el servidor, para cada paquete genera un "encabezado de flujo" con el número de secuencia de paquete, el ID de flujo, versión de flujo, y HMAC (por ejemplo, firmado con una clave privada de flujo desde la señal de flujo), y luego pasa ese paquete al primer relé. Luego, el cliente aumenta el número de secuencia de paquete, que inicia en 0 y aumenta en 1 con cada paquete enviado hacia el servidor. La clave privada de flujo es una clave simétrica independiente que se usa para asegurar el flujo contra paquetes no autorizados. La clave privada de flujo puede generarse aleatoriamente para cada flujo concedido por el servidor 102 maestro.

Cuando el cliente recibe paquetes de carga útil desde el servidor, compara su número de secuencia de paquete contra el búfer de protección de reproducción. Los paquetes se descartan si ya se han recibido o son demasiado antiguos. Esto evita una clase de ataques a nivel de protocolo llamados ataques de "reproducción", donde los paquetes válidos ya intercambiados a través del sistema son reproducidos por un atacante en un intento de romper el protocolo. Muchos de estos conceptos se describen más completamente a continuación.

Comportamientos de relé

Los relés a través de realizaciones pueden tener algunos comportamientos comunes. Por ejemplo, cuando los paquetes se reciben a través de la red mediante un relé, si el primer byte en el paquete es 0, indicando un "paquete de solicitud", el relé en algunas realizaciones tomará varias acciones: (1) el relé desencriptará la primera señal de flujo en el paquete (por ejemplo, la señal que corresponde a ese relé en la ruta de flujo) usando la clave privada de relé y la clave pública de servidor maestro; (2) si la señal de flujo falla en desencriptar, el relé ignora el paquete; (3) el relé prueba si la señal de flujo ha expirado, e ignora el paquete si la señal ha expirado; (4) el relé busca una entrada de flujo que corresponde al ID de flujo y versión de flujo (por ejemplo, la tupla {ID de flujo, versión de flujo}) en la señal de flujo; (5) si la entrada ya existe, el relé actualiza la marca de tiempo en que fue recibido el último paquete desde el nodo previo a la marca de tiempo actual; (6) si la entrada aún no existe en la caché del relé, el relé crea una nueva entrada para este flujo, indexada por ID de flujo y versión de flujo (por ejemplo, la tupla {ID de flujo, versión de flujo}) con la marca de tiempo de los últimos paquetes recibidos desde los nodos previos y actuales establecidos en la marca de tiempo actual, (6a) si la dirección previa en la señal de flujo está marcada como "ninguna", entonces la dirección y puerto previos en la nueva entrada de flujo se establecen en la dirección de IP + puerto desde donde fue enviado el paquete de solicitud, lo que permite a los clientes sin dirección de IP pública fija y puerto (por ejemplo, clientes detrás de NAT) participar en rutas de flujo; (7) en ambos casos 5 y 6, el relé toma el paquete de solicitud y retira tanto el byte de prefijo (que es 0 para este tipo de paquete) como la primera señal de flujo; (8) el relé luego agrega un nuevo byte

de prefijo de 0 delante del resto del paquete de solicitud, y pasa este paquete modificado al siguiente nodo en la ruta de flujo (por ejemplo, el siguiente relé, o si el servidor es el siguiente, el servidor).

La clave privada de relé mencionada anteriormente puede generarse aleatoriamente para cada relé. Cada clave privada de relé tiene una clave pública correspondiente. La clave privada de relé permite que el servidor 102 maestro comunique señales de flujo a ese relé de manera segura, conociendo solo la clave pública de ese relé. En algunas implementaciones los nodos en los extremos de cada flujo, por ejemplo, clientes y servidores, también pueden tener sus propias claves privadas generadas aleatoriamente. El término "clave privada de nodo" se usa a veces en este documento para referirse en general a claves privadas para relés y otros tipos de nodos, tales como clientes y servidores.

Si el primer byte de paquete es 1, indicando un "paquete de respuesta", en algunas realizaciones el relé tomará varias acciones: (1) busca la entrada de flujo que corresponde al ID de flujo y versión de flujo (por ejemplo, la tupla {ID de flujo, Versión de flujo}) en el paquete; (2) si no existe ninguna entrada de flujo, el relé ignora el paquete; (3) el relé verifica que el HMAC del paquete indique que los datos de paquete (número de secuencia, ID de flujo, versión de flujo) fueron firmados con la clave privada de flujo (que fue enviada al relé en la señal de flujo, en el paquete de solicitud); (4) si la firma no coincide, el relé ignora el paquete; (5) el relé prueba el número de secuencia de paquete contra el búfer de protección de reproducción para paquetes recibidos desde el siguiente nodo, y si el paquete ya se ha recibido, o es antiguo (por ejemplo, el paquete está fuera del búfer de protección de reproducción), el relé ignora el paquete; (6) de lo contrario, el paquete es válido, y el relé reenvía el paquete, sin modificación, al nodo previo (por ejemplo, el relé previo o el cliente si el relé es el primer relé en la ruta de flujo); (7) actualiza la marca de tiempo del último paquete recibido desde el siguiente nodo en la entrada a la marca de tiempo actual.

Si el primer byte de paquete es 2, indicando un "paquete de cliente a servidor", en algunas realizaciones el relé tomará varias acciones: (1) el relé busca una entrada de flujo que corresponde al ID de flujo y a versión de flujo (por ejemplo, la tupla {ID de flujo, versión de flujo}) en el paquete; (2) si no existe ninguna entrada de flujo, el relé ignora el paquete; (3) el relé verifica que el HMAC del paquete indique que los datos de paquete (número de secuencia, ID de flujo, versión de flujo) fueron firmados con la clave privada de flujo (que fue enviada al relé en la señal de flujo, en el paquete de solicitud); (4) si la firma no coincide, el relé ignora el paquete; (5) probar el número de secuencia de paquete contra el búfer de protección de reproducción para paquetes recibidos desde el nodo previo, y si el paquete ya ha sido recibido o es antiguo (por ejemplo, fuera del búfer de protección de reproducción), el relé ignora el paquete; (6) de lo contrario, el paquete es válido, y el relé reenvía el paquete, sin modificación, al siguiente nodo (por ejemplo, el siguiente relé, o el servidor, si este relé es el último relé antes del servidor); y (7) actualiza la marca de tiempo del último paquete recibido desde el nodo previo a la marca de tiempo actual.

Si el primer byte de paquete es 3, indicando un "paquete de servidor a cliente", en algunas realizaciones el relé tomará varias acciones: (1) el relé busca una entrada de flujo que corresponde al ID de flujo y a versión de flujo (por ejemplo, la tupla {ID de flujo, versión de flujo}) en el paquete; (2) si no existe ninguna entrada de flujo, el relé ignora el paquete; (3) el relé verifica que el HMAC del paquete indique que los datos de paquete (número de secuencia, ID de flujo, versión de flujo) fueron firmados con la clave privada de flujo (que fue enviada al relé en la señal de flujo, en el paquete de solicitud); (4) si la firma no coincide, el relé ignora el paquete; (5) el relé prueba el número de secuencia de paquete contra el búfer de protección de reproducción para paquetes del siguiente nodo, y si el paquete ya ha sido recibido, o es antiguo (por ejemplo, fuera del búfer de protección de reproducción), ignora el paquete; (6) de lo contrario, el paquete es válido, y el relé reenvía el paquete, sin modificación, al nodo previo en la ruta de flujo (que es el relé previo, o el cliente, para el primer nodo de relé en el flujo); y (7) actualiza la marca de tiempo del último paquete recibido desde el siguiente nodo a la marca de tiempo actual.

Si en cualquier momento una entrada en la caché de relé no ha recibido paquetes desde el nodo previo durante algún período de tiempo (por ejemplo, 5 segundos), o, no ha recibido paquetes desde el siguiente nodo durante algún período de tiempo (por ejemplo, 5 segundos), esa entrada de flujo indexada por la tupla {ID de flujo, versión de flujo} expira, y se retira de la caché de relé. En este punto el relé cesa de reenviar paquetes para el flujo identificado por {ID de flujo, versión de flujo} en cualquier dirección.

Comportamientos de servidor

Al igual que los relés, los servidores a través de realizaciones pueden tener comportamientos comunes. Los servidores escuchan paquetes y crean entradas para sesiones de clientes. Las entradas se indexan mediante ID de flujo para que el servidor pueda "actualizar" una sesión de cliente cuando llega con una versión de flujo más reciente (por ejemplo, se ha establecido una ruta de flujo actualizada para un cliente). Esto permite una transición sin interrupciones desde una ruta de flujo a otra.

Si el primer byte de paquete es 0, es un "paquete de solicitud". En algunas realizaciones los servidores tomarán las siguientes acciones: (1) el servidor desencriptará la primera señal de flujo en el paquete usando la clave privada de servidor y la clave pública del servidor maestro; (2) si la señal de flujo falla en desencriptar, el servidor ignora el paquete; (3) si la señal de flujo ha expirado, el servidor ignora el paquete; (4) de lo contrario, el servidor busca una entrada con el ID de flujo en la señal; (5) si ya existe una entrada, y el número de versión de flujo es el mismo, el

servidor actualiza la marca de tiempo en la que fue recibido el último paquete en esa entrada a la marca de tiempo actual; (6) si ya existe una entrada, pero el número de versión de flujo es mayor en el paquete de solicitud que el valor de entrada, el servidor copia a través de los nuevos datos de flujo (por ejemplo, como si fuera una nueva sesión de cliente) y actualiza la marca de tiempo en la que fue recibido el último paquete en esa entrada de la marca de tiempo actual; (7) de lo contrario, el servidor agrega una nueva entrada de flujo, indexada por ID de flujo, con la marca de tiempo en la que fue recibido el último paquete en esa entrada establecida en la marca de tiempo actual; (8) en todos los casos anteriores (5, 6, y 7), el servidor responde con un "paquete de respuesta" al relé previo con el número de secuencia de paquete para esta entrada con el bit alto establecido en 1 (por ejemplo, para evitar repetición del mismo valor *nonce* entre paquetes de cliente a servidor y paquetes de servidor a cliente); y (9) después de que el servidor envía el paquete de respuesta al nodo previo, se incrementa el número de secuencia de paquete para esta entrada de flujo.

Si el primer byte de paquete es 2, es un "paquete de cliente a servidor". En algunas realizaciones los servidores tomarán las siguientes acciones: (1) el servidor busca la entrada de flujo correspondiente para el ID de flujo en el paquete; (2) si no existe ninguna entrada de flujo, el servidor ignora el paquete; (3) el servidor verifica el HMAC en el paquete para asegurarse de que la verificación de firma pase de acuerdo con la clave privada de flujo que corresponde a esta entrada de flujo, y si no es así, el servidor ignora el paquete; (4) el servidor suministra la carga útil de paquete al usuario. Esto permite al cliente enviar datos a través de la ruta de flujo al servidor, como si estuviera conectado directamente al servidor.

Un usuario puede enviar paquetes de carga útil desde el servidor al cliente especificando el ID de flujo al que se deben enviar los paquetes. Cuando el servidor envía paquetes de carga útil a un cliente, el servidor busca la entrada en su caché para ese ID de flujo. Luego el servidor genera un "encabezado de flujo" con el número de secuencia de paquete desde esa entrada de flujo, el ID de flujo, versión de flujo, y HMAC (por ejemplo, firmado con una clave privada de flujo desde la señal de flujo), y pasa ese paquete al relé previo en la ruta de flujo para esa entrada de flujo. Al igual que con los paquetes de respuesta, el servidor establece el bit alto del número de secuencia de paquete en 1, para asegurar que los valores de números de secuencia de paquete (*nonce*) sean únicos a través de los paquetes de cliente a servidor y de servidor a cliente para un flujo particular. Luego el servidor aumenta el número de secuencia de paquete para esa entrada de flujo, que inicia en 0 y aumenta en 1 con cada paquete enviado al cliente en ese flujo.

Si en cualquier momento una entrada de flujo en la caché del servidor no ha recibido un paquete durante alguna cantidad de tiempo (por ejemplo, 5 segundos), esa entrada indexada por ID de flujo expira y se retira de la caché. En este punto el servidor deja de ser capaz de recibir paquetes enviados desde el cliente que corresponde a ese ID de flujo, y de ser capaz de enviar paquetes al cliente que corresponde a ese ID de flujo.

La siguiente discusión describe cómo operan estos comportamientos de cliente, servidor, y relé en el contexto del establecimiento de un flujo entre un servidor y cliente. Aunque algunas de las descripciones a continuación pueden incluir detalles diferentes, se contempla que cualquiera de los comportamientos descritos anteriormente se puede implementar donde sea necesario en los procesos o etapas que se describen a continuación.

La figura 3 muestra las primeras etapas para establecer una ruta de flujo. En el contexto de un juego, por ejemplo, el adaptador 101 es un servidor propiedad de una empresa de juegos que hace un seguimiento de todos los servidores dedicados s_1 s_2...s_j que están operando para alojar el juego. La solicitud del cliente al adaptador incluye un conjunto de parámetros (por ejemplo, tipo de juego, número de jugadores, mapa de juego, etc.) junto con la clave pública del cliente, como se muestra en 301. La solicitud 301 del cliente al adaptador 101 puede llevarse a cabo sobre, por ejemplo, una REST API. Esta solicitud 301 incluye pasar al adaptador la clave pública del cliente.

Debido a que el adaptador 101 conoce los parámetros de servidor deseados del cliente, puede identificar los servidores s_1 s_2...s_j que satisfacen la solicitud 101 del cliente. Con un conjunto de servidores dedicados s_1 s_2...s_j identificados, el adaptador 101 puede crear una solicitud de ruta y enviarla al servidor 102 maestro, como se muestra en 302.

Una solicitud 302 de ruta incluye la clave pública del cliente, así como las claves públicas y direcciones de IP y puertos de los servidores dedicados s_1s_2...s_j que satisfacen la solicitud 301 inicial del cliente (teniendo en cuenta que no es necesaria una pluralidad de servidores dedicados en una solicitud de ruta). No es necesario conocer la dirección de IP del cliente en este proceso. En cambio, el servidor 102 maestro necesita conocer la clave pública del cliente como mínimo, debido a que la dirección del cliente en la ruta de flujo está establecida en "ninguna" y puede ser determinada por el primer relé r_1 como la dirección desde la cual fue enviado el paquete de solicitud.

El servidor 102 maestro recibe las solicitudes 302 de ruta desde el adaptador 101 (por ejemplo, a través de REST API), y el servidor 102 maestro identifica las mejores rutas para cada uno de los servidores dedicados s_1 s_2...s_j identificados por el adaptador 101 de acuerdo con algunos criterios (por ejemplo, latencia mínima, pérdida de paquetes, variación rápida, etc.). El servidor 102 maestro luego responde al adaptador 101 con un arreglo de rutas desde el cliente a los servidores 401, correspondiendo cada ruta a un servidor en la lista de servidores dedicados en la solicitud de ruta, como se muestra en la figura 4. En realizaciones alternativas de la presente invención el servidor 102 maestro puede responder a la solicitud del adaptador enviando un ID de sesión y un arreglo de señales de sesión al adaptador 101. Cada señal de sesión corresponde a un servidor dedicado identificado, y el ID de sesión identifica

la sesión del cliente. En algunas realizaciones, el ID de sesión es un número (por ejemplo, un número de 64 bits, un número de 128 bits, etc.). Aunque no es un requisito, es preferible que cada ID de sesión sea único.

Cada ruta de flujo tiene señales de flujo. La primera señal de flujo corresponde al cliente 103. Está encriptada con la clave pública del cliente y la clave privada del servidor maestro. Las señales que vienen después de la señal de cliente pero antes de la señal de servidor (la última señal) corresponden a relés, y cada una está encriptada con la clave privada del servidor maestro y la clave pública del relé correspondiente. La última señal de flujo en cada ruta de flujo se encripta con la clave pública del servidor y la clave privada del servidor maestro. Las señales de flujo encriptadas son luego transmitidas 402 al cliente 103 por el adaptador 101.

Al hacer que el servidor maestro envíe el arreglo de rutas al servidor al adaptador en lugar de directamente al cliente, el cliente nunca obtiene acceso a información sobre el servidor maestro (por ejemplo, la dirección de IP). Esto ayuda a proteger el servidor maestro (que puede ser propiedad/operado mediante, por ejemplo, una entidad separada de la entidad que posee/opera el adaptador) de ataques.

En realizaciones alternativas las señales de sesión se usan para mantener una conexión segura. El contenido de una señal de sesión se ve en la figura 14. Las señales de sesión incluyen dos subseñales: una señal de inicio de sesión y una señal de continuación de sesión. La señal de inicio de sesión incluye tanto información pública como privada. La información privada se encripta de manera asimétrica, de tal manera que solo puede ser creada por el servidor maestro, y solo leída por el relé correspondiente. La información pública es fácilmente legible, pero está firmada de tal manera que su autenticidad pueda ser verificada por un receptor. La información privada en la señal de inicio de sesión incluye, por ejemplo, una dirección de IP y puerto de servidor dedicado, un número de secuencia de sesión, el ID de sesión, un límite superior de ancho de banda, y un límite inferior de ancho de banda. La información pública en la señal de continuación de sesión incluye, por ejemplo, una dirección de IP de un relé y una marca de tiempo de expiración. La señal de continuación de sesión tiene información privada que incluye un número de secuencia de sesión y un ID de sesión.

La figura 5A muestra una realización de una ruta de flujo. Dentro de cada ruta de flujo hay una serie de señales de flujo, correspondiendo cada señal de flujo con un nodo particular. El nodo 0 siempre corresponde al cliente, y el último nodo (por ejemplo, nodo n-1) siempre corresponde al servidor dedicado. Todos los nodos en medio (por ejemplo, nodos 1 hasta n-2) corresponden a relés, y están ordenados en una secuencia que indica una ruta de flujo deseada. La figura 5B muestra una realización de una señal de flujo, que incluye: ID de flujo, versión de flujo, marca de tiempo de expiración, dirección de IP + puerto de nodo previo, dirección de IP + puerto de siguiente nodo, y una clave privada de flujo. En algunas realizaciones, la IP + dirección + puerto de nodo previo en la señal de flujo, se puede sustituir por una entrada "ninguno", indicando que el relé que corresponde a esa señal debe usar la dirección desde la que fue enviado el paquete de solicitud como la dirección de IP + puerto previo para esa entrada de flujo.

Las figuras 6 y 7 ilustran cómo se puede establecer un flujo entre un cliente 103 y un servidor s_j a través de cualquier número de relés r_1, r_2...r_i. Aunque la realización mostrada en las figuras incorpora tres relés, se contempla que se pueda implementar cualquier número de relés usando realizaciones de la materia objeto inventiva.

Las figuras 6 y 7 demuestran las acciones emprendidas para una ruta de flujo única. En realizaciones donde el arreglo de rutas al servidor s_j incluye más de una ruta, el cliente itera a través de cada ruta de flujo hasta que se establece un flujo. Por ejemplo, si el cliente 103 y servidor s_j no son capaces de establecer un flujo usando la primera ruta de flujo en alguna cantidad de tiempo (por ejemplo, 1 segundo), el cliente 103 se mueve a la segunda ruta de flujo para intentar establecer el flujo que contiene la segunda ruta de flujo, y hace lo mismo para la tercera ruta, etc. En algunas realizaciones, el cliente 103 intenta establecer un flujo usando todas las rutas de flujo simultáneamente, y acepta el primer flujo que se establece. En otras realizaciones, un cliente 103 puede intentar establecer un flujo usando subgrupos de rutas al servidor s_j. De manera similar en realizaciones alternativas que usan señales de sesión el cliente 103 recibe las señales de sesión desde el adaptador 101 y puede iterar a través del conjunto de señales de sesión, intentando usar cada señal de sesión para establecer una conexión con un servidor dedicado a través de uno o más relés. El cliente cesa de iterar a través de señales de sesión después de que establece exitosamente una conexión con un servidor dedicado a través de uno o más relés.

Igual que cada nodo (por ejemplo, cliente, relé, o servidor) de la materia objeto inventiva tiene un par de clave pública y clave privada, el servidor 102 maestro también tiene un par de clave pública y clave privada. Cada vez que se crea una ruta de flujo, cada señal de flujo dentro de esa ruta se encripta usando la clave privada del servidor maestro y la clave pública del nodo correspondiente, sea ese nodo un cliente, un relé o un servidor). De este modo, cada señal de flujo solo puede ser generada por el servidor maestro y no puede ser modificada por ningún tercero, y solo puede ser desencriptada por el nodo particular para el que fue generada.

De este modo, 601 muestra que el cliente 103 recibe una ruta de flujo y desencripta la primera señal, reemplazando la primera señal con un indicador de tipo de paquete de solicitud, un prefijo de un único byte "0", creando un paquete de solicitud. Para el contexto, la figura 6 muestra que hay relés "i". Esta notación se usa para indicar que, en el contexto de la figura 6, i es cualquier número entre 4 y un número alto arbitrario que está limitado solamente por un número de

relés que pueden desplegarse razonablemente en el mundo real (por ejemplo, como dispositivos físicos o virtuales). De este modo se contempla que pueda haber cualquier número de relés entre 1 y ese número alto arbitrario.

El cliente 103 es capaz de desencriptar la primera señal de flujo en la ruta de flujo debido a que fue encriptada usando la clave pública del cliente y la clave privada del servidor maestro. Con la primera señal (por ejemplo, la señal que corresponde al nodo 0 como se ilustra en la figura 5A) reemplazada por un indicador de tipo de paquete (por ejemplo, 0), la ruta de flujo se convierte en un paquete de solicitud. De este modo el paquete de solicitud incluye una señal menos, y la primera señal en el paquete de solicitud corresponde ahora al nodo 1, que es el primer relé r_1 en la ruta de flujo. El cliente luego envía una serie de este paquete de solicitud al primer relé r_1 indicado en la ruta de flujo (y cuya dirección está contenida en la señal de flujo del cliente como la dirección de IP + puerto de siguiente nodo) con el objetivo final de establecer un flujo. En algunas realizaciones, la serie de paquetes de solicitud se envía a alguna frecuencia (por ejemplo, 10 Hz) durante un período de tiempo (por ejemplo, 5 segundos), mientras que en otras realizaciones, se envía una cantidad de paquetes de solicitud (por ejemplo, 100) independientemente del tiempo. Esto se aplica cada vez que un nodo envía "una serie" de paquetes.

En realizaciones que utilizan una señal de sesión las conexiones se establecen en fases. Una vez que el cliente ha comenzado a enviar paquetes a un relé (por ejemplo, el relé indicado en la señal de sesión), el cliente pasa a través de dos fases de envío de paquetes. En una primera fase de envío de paquetes, el cliente envía paquetes al relé identificado que tiene el prefijo de la señal de inicio de sesión. Estos paquetes se envían durante una cantidad de tiempo (por ejemplo, 1-2 segundos, 2-3 segundos, 3-4 segundos, 4-5 segundos, 5-10 segundos, 10-15 segundos). Después de que expira esa cantidad de tiempo, en la segunda fase de envío de paquetes, los paquetes tienen el prefijo de una señal de continuación de sesión en lugar de una señal de inicio de sesión.

El primer relé r_1 (que corresponde al nodo 1 en la ruta de flujo) recibe al menos uno de los paquetes de solicitud enviados desde el cliente 103, como muestra 602. El primer relé r_1 desencripta la primera señal del paquete de solicitud antes de reemplazar tanto la primera señal como el indicador de tipo de paquete existente con un indicador de tipo de paquete de solicitud (por ejemplo, 0 en este caso dado que el paquete es un paquete de solicitud). Debido a que el cliente 103 ya desencriptó la "primera" señal original y la reemplazó con un indicador de tipo de paquete de solicitud, la nueva "primera" señal es una señal que el primer relé r_1 (y solo el primer relé) puede desencriptar como fue encriptada usando la clave pública del primer relé y la clave privada del servidor maestro.

Siempre que una señal de flujo se refiera al cliente 103 como el nodo previo (por ejemplo, la señal de flujo que corresponde al primer relé en una ruta de flujo), el cliente tendrá un tipo de dirección de 0 (donde el tipo 0 indica una dirección desconocida o "ninguna", tipo 1 indica una dirección IPv4, y tipo 2 indica una dirección IPv6). Cada vez que llega un paquete de solicitud y la señal correspondiente tiene una dirección previa de tipo 0, se reemplaza con la dirección + puerto desde donde fue enviado el paquete de solicitud. De este modo, nunca hay una necesidad de que se incluya la dirección del cliente en la señal del relé. Esto se usa principalmente para manejar situaciones donde el cliente 103 está detrás de la traducción de direcciones de red (NAT) (por ejemplo, la dirección de IP pública + puerto de ese cliente es generado dinámicamente por un enrutador), pero el concepto se puede extender y usar en relación con cualquier nodo donde un nodo previo tiene un tipo de dirección de 0. Esto puede ser útil para situaciones donde algunos nodos a lo largo de la ruta están en una red privada y no exponen o no necesariamente conocen sus direcciones de IP públicas con anticipación.

El primer relé r_1 luego verifica si el ID de flujo y versión de flujo en esa señal ya existen en la caché del relé y dónde enviar el paquete a continuación. Si el ID de flujo y versión de flujo son nuevos, entonces el ID de flujo y otros contenidos de la señal se almacenan en la caché del relé. El paquete de solicitud modificado luego se pasa al siguiente nodo en la ruta de flujo.

En realizaciones que utilizan señales de sesión. Las señales de inicio de sesión tienen el prefijo en paquetes durante una cantidad de tiempo limitada para asegurar que el relé reciba la señal de inicio de sesión. Cuando un relé recibe un paquete con prefijo de una señal de inicio de sesión, el relé primero verifica la marca de tiempo de expiración (que se almacena como datos públicos). Si la señal de inicio de sesión ha expirado, el paquete se ignora. A continuación, el relé ejecuta una verificación de firma y autentificación para asegurarse de que la señal de inicio de sesión sea válida y fuera generada por el servidor maestro. Después de esto desencripta la señal de inicio de sesión.

Una vez desencriptada, el relé luego verifica para ver si el ID de sesión en la señal de inicio de sesión ya existe en la caché del relé. El relé también puede verificar los límites superior/inferior de ancho de banda para la sesión (como se indica en la señal de inicio de sesión), y finaliza la sesión si el ancho de banda excede los límites en cualquier dirección durante algún período de tiempo (por ejemplo, 1-5 segundos, 5-10 segundos, 10-15 segundos, u otro tiempo especificado). De esta forma incluso si un cliente tiene una señal de sesión válida, ese cliente aún no puede DDoS en el servidor dedicado.

Si el ID de sesión es nuevo, entonces el ID de sesión y otros contenidos de la señal de inicio de sesión (por ejemplo, toda o alguna de la información pública y privada en la señal de inicio de sesión) se almacenan en la caché del relé antes de enviar el paquete al servidor dedicado u otro relé. Antes de enviar el paquete al servidor dedicado u otro relé, el relé elimina el prefijo (por ejemplo, la señal de inicio de sesión) del paquete y lo reemplaza con el ID de sesión y

número de secuencia de sesión antes de pasar el paquete al servidor dedicado s_j o relé r_i que fue identificado en la señal de sesión.

Si el relé encuentra que el ID de sesión en una señal de inicio de sesión ya existe en su caché (indicando que ya se ha recibido y registrado un paquete con prefijo de una señal de inicio de sesión), entonces el paquete se pasa al servidor dedicado u otro relé. De nuevo, antes de enviar el paquete al servidor dedicado u otro, el relé elimina el prefijo (por ejemplo, la señal de inicio de sesión) del paquete y lo reemplaza con el ID de sesión y número de secuencia de sesión antes de pasar el paquete al servidor dedicado s_j o relé r_i que fue identificado en la señal de sesión.

Si un relé recibe paquetes que tienen el prefijo de una señal de continuación de sesión, el relé solo verifica si el ID de sesión ya existe en la caché. Si el ID de sesión existe en la caché del relé, el relé procede a eliminar el prefijo y lo reemplaza con el ID de sesión y número de secuencia de sesión antes de pasar el paquete al servidor dedicado. Si el ID de sesión no existe en la caché, el paquete se ignora.

La figura 8A ilustra contenido en una caché del relé según sea necesario para algunas realizaciones de la materia objeto inventiva. La caché para cada relé incluye una tabla que tiene claves y valores, donde una clave incluye la tupla {ID de flujo, versión de flujo}, y un valor que corresponde a cada clave incluye una entrada. La figura 8B muestra datos de entrada de ejemplo divididos en datos de señal y datos de tiempo de ejecución. Los datos de señal incluyen: marca de tiempo de expiración, dirección de nodo previo (por ejemplo, dirección de IP y puerto), dirección de siguiente nodo (por ejemplo, dirección de IP y puerto), y clave privada de flujo. Los datos de tiempo de ejecución incluyen: el tiempo en que fue recibido por última vez un paquete desde un nodo previo, el tiempo en que fue recibido por última vez un paquete desde el siguiente nodo, protección de reproducción de nodo previo, y protección de reproducción de siguiente nodo. La protección de reproducción se discute con más detalle a continuación.

En algunas realizaciones de la presente divulgación el segundo relé r_4 (que corresponde al nodo 2 de la ruta de flujo) recibe al menos uno de los paquetes de solicitud enviados desde el nodo 1 (es decir, relé r_1), como se muestra en 603. El segundo relé r_4 desencripta la primera señal del paquete de solicitud antes de reemplazar una vez más tanto la primera señal como el indicador de tipo de paquete existente con un indicador de tipo de paquete de solicitud (por ejemplo, 0 en este caso dado que el paquete es un paquete de solicitud). Debido a que el primer relé r_1 ya desencriptó la "primera" señal previa y la reemplazó con un indicador de tipo de paquete de solicitud, la nueva "primera" señal es una señal que el segundo relé r_4 (y solo el segundo relé) puede desencriptar como fue encriptada usando la clave pública del segundo relé y la clave privada del servidor maestro.

El segundo relé r_4 luego verifica para ver si el ID de flujo y versión de flujo en esa señal ya existen en la caché del relé y dónde enviar el paquete a continuación. Si el ID de flujo y versión de flujo son nuevos, entonces el ID de flujo y otros contenidos de la señal se almacenan en la caché del relé. El paquete de solicitud modificado luego se pasa al siguiente nodo en la ruta de flujo.

El tercer relé r_3 (que corresponde secuencialmente al siguiente nodo después del nodo 2 en la ruta de flujo original) recibe al menos uno de los paquetes de solicitud enviados desde el nodo 2 (es decir, relé r_4), como se muestra en 604. El tercer relé r_3 desencripta la primera señal del paquete de solicitud antes de reemplazar una vez más tanto la primera señal como el indicador de tipo de paquete existente con un indicador de tipo de paquete de solicitud (por ejemplo, 0 en este caso dado que el paquete es un paquete de solicitud). Debido a que el segundo relé r_4 ya desencriptó la "primera" señal previa y la reemplazó con un indicador de tipo de paquete de solicitud, la nueva "primera" señal será una señal que el tercer relé r_3 (y solo el tercer relé) puede desencriptar como fue encriptada usando la clave pública del tercer relé y la clave privada del servidor maestro.

El tercer relé r_3 luego verifica para ver si el ID de flujo y versión de flujo en esa señal ya existen en la caché del relé y dónde enviar el paquete a continuación. Si el ID de flujo y versión de flujo son nuevos, entonces el ID de flujo y otros contenidos de la señal se almacenan en la caché del relé. El paquete de solicitud modificado luego se pasa al siguiente nodo en la ruta de flujo. Aunque en las figuras el tercer relé r_3 es el relé final, se contempla que puedan usarse tantos o tan pocos relés como sea necesario para encontrar una ruta de flujo óptima.

Finalmente, un servidor dedicado s_j (que corresponde al nodo final en el paquete de solicitud) recibe al menos uno de los paquetes de solicitud enviados desde el nodo 3 (es decir, relé r_3), como se muestra en 605. El servidor dedicado s_j desencripta la primera señal del paquete de solicitud (que ahora corresponde a una señal de flujo que solo el servidor puede desencriptar, dado que está encriptada con la clave privada del servidor maestro y la clave pública del servidor) y verifica para ver si el ID de flujo en esa señal ya existe en la caché del servidor dedicado. Si el ID de flujo es nuevo, entonces el ID de flujo y otros contenidos de la señal se almacenan en la caché del servidor dedicado. El servidor responde a cada paquete de solicitud válido con un paquete de respuesta enviado al nodo previo en la ruta de flujo.

Por razones de seguridad, los tipos de paquetes 1, 2, y 3 están "firmados" con la clave privada de flujo, que se incluye en cada señal de flujo, y es la misma para cada entrada de flujo que corresponde a este flujo en cada nodo involucrado (por ejemplo el cliente, los relés y el servidor). Esto permite que cada nodo rechace trivialmente paquetes enviados por partes no autorizadas (por ejemplo, partes que no conocen la clave privada de flujo). Es importante anotar que los paquetes de respuesta y carga útil (por ejemplo, tipos 1, 2, y 3) no están encriptados, solo están firmados. De este

modo, los contenidos son legibles por cualquiera, pero un tercero no puede generar ni modificar el ID de flujo o número de versión de flujo en el encabezado de flujo para estos tipos de paquetes. Para que estas realizaciones funcionen, los tipos de paquetes 1, 2, y 3 deben tener un número de secuencia de paquete (por ejemplo, un número "*nonce*" que se usa solo una vez), y un código de autentificación de mensaje de direccionamiento con clave (HMAC). Para evitar que el número de secuencia de paquete se use más de una vez, los paquetes de tipo 1, 2 y 3 enviados en la dirección de cliente a servidor tienen el bit alto del número de secuencia de 64 bits establecido en 0, y los paquetes de tipo 1, 2 y 3 enviados en la dirección de servidor a cliente tienen el bit alto del número de secuencia de 64 bits establecido en 1.

Se puede obtener un proceso de conexiones más seguras en la realización alternativa utilizando señales de sesión. En la realización que usa señales de sesión se contempla que cada relé tenga un par de clave pública/clave privada correspondiente. Esto facilita la encriptación de señales por el servidor maestro para asegurar que las señales (por ejemplo, señales de sesión) solo se puedan leer por el relé para el que el servidor maestro generó las señales, a través de encriptación asimétrica. Esto asegura que si un relé se compromete, no comprometerá a todos los otros relés en el sistema. En algunas realizaciones, los relés necesitan un certificado para registrarse con el servidor maestro, permitiendo que se revoque el certificado de relés comprometidos. En algunas realizaciones, los relés generan automáticamente nuevos pares de clave pública/clave privada (por ejemplo, a intervalos regulares o irregulares tales como 5-10 minutos, o cada hora o cualquier combinación de intervalos dentro de ese rango). Cada vez que un relé genera un nuevo par de clave pública/clave privada, el relé comunica su nueva clave pública al servidor maestro.

Adicionalmente en realizaciones que utilizan señales de sesión de acuerdo con aspectos de la presente divulgación, se contempla que el servidor maestro también tenga su propio par de clave pública/clave privada. El servidor maestro da su clave pública a los relés. De este modo, los relés pueden desencriptar, pero no escribir, señales de sesión. También se contempla que un relé pueda firmar, o encriptar y firmar paquetes que intercambia con un servidor dedicado.

Al igual que los relés, los servidores dedicados tienen una caché para almacenar datos relacionados con diferentes flujos. Esto permite que un servidor haga un seguimiento de que un flujo particular se relaciona con un cliente particular, y así sucesivamente. La figura 9A muestra un ejemplo de cómo un servidor puede almacenar claves y valores relacionados con flujos. Una clave incluye el ID de flujo, y un valor que corresponde a cada clave incluye una entrada. La entrada tiene datos de señal y datos de tiempo de ejecución, se muestra en la figura 9B. Los datos de señal incluyen: una marca de tiempo de expiración, una dirección de IP + puerto de nodo previo, una clave privada de flujo, y una versión de flujo. Los datos de tiempo de ejecución incluyen el tiempo en que fue recibido el último paquete, un número de secuencia de paquete, y un búfer de protección de reproducción.

Cuando se recibe un paquete de solicitud válido en el servidor dedicado r_j, el servidor dedicado r_j responde con paquetes de respuesta al nodo previo r_3, como se muestra en 701. Finalmente, los paquetes de respuesta se reenvían de vuelta al cliente 103 a través de la misma ruta de flujo definida por el paquete de solicitud, pero al revés. Debe entenderse que la ruta de flujo inverso no necesita limitarse a la misma ruta exacta que el flujo directo y el flujo inverso puede tomar cualquier número de otras

Como se muestra en 702, el relé r_3 recibe un paquete de respuesta desde el servidor dedicado s_j que está firmado usando una clave privada de flujo (por ejemplo, la clave privada de flujo contenida dentro de los datos de señal que el servidor dedicado desencriptó del paquete de solicitud que finalmente llegó al servidor dedicado s_j pasó a lo largo del conjunto de relés en la ruta de flujo). El relé r_3 busca la entrada de flujo en su caché por ID de flujo y versión de flujo, luego verifica ver si la firma es válida. Si la firma es válida, el relé reenvía el paquete de respuesta al relé previo r_4.

Como se muestra en 703, el relé r_4 recibe un paquete de respuesta desde el relé r_3 que está firmado usando la clave privada de flujo. El relé r_4 busca la entrada de flujo por ID de flujo y versión de flujo, luego ve si la firma es válida. Si la firma es válida, el relé reenvía el paquete de respuesta al relé previo r_1.

Como se muestra en 704, el relé r_1 recibe un paquete de respuesta desde el relé r_4 que está firmado usando la clave privada de flujo. El relé r_1 busca la entrada de flujo por ID de flujo y versión de flujo, luego ve si la firma es válida. Si la firma es válida, el relé reenvía los paquetes de respuesta al nodo previo (en este caso, el cliente o nodo 0).

Cuando el cliente recibe el paquete de respuesta desde el primer relé, el cliente realiza la misma verificación de firma que completaron todos los nodos previos, y si el paquete pasa, el cliente considera que el flujo está "establecido". Una vez que se establece un flujo, el cliente tiene confirmación de que los paquetes de carga útil (por ejemplo, tipos de paquetes 2 y 3) se pueden intercambiar entre el cliente y el servidor a través de la ruta de flujo. Los paquetes de carga útil, al igual que paquetes de respuesta, tienen su encabezado de flujo firmado por la clave privada de flujo, y se pueden estructurar como sigue: [2 o 3][secuencia de paquetes][ID de flujo][versión de flujo][hmac](datos de carga útil).

En algunas realizaciones, el cliente puede comenzar a enviar paquetes de carga útil al servidor dedicado antes de recibir un paquete de respuesta desde el servidor dedicado. Esto puede ayudar a minimizar la latencia para establecer un flujo dado que, en la mayoría de los casos los paquetes de solicitud llegarán a cada relé y al servidor antes que el

paquete de carga útil, "perforando" de este modo la ruta de flujo de tal manera que los paquetes que se mueven desde el cliente al servidor puedan reenviarse inmediatamente al siguiente nodo en el caso común.

En algunos casos, el flujo puede volverse poco fiable o lento por cualquier número de razones (por ejemplo, un nodo es atacado, un nodo comienza a ralentizarse inesperadamente, pérdida de paquetes se vuelve inaceptablemente alta, etc.). En otros casos, la ruta de flujo existente puede todavía ser de alta calidad, pero ya está disponible una ruta de flujo mejor. En estos casos, puede resultar necesario que se actualice la ruta de flujo.

Mientras los paquetes de carga útil continúan intercambiándose entre el cliente y el servidor dedicado a través de la ruta 1003 de flujo existente (usando el ID de flujo original y número de versión de flujo que corresponde a esa ruta de flujo), el cliente 103 puede solicitar una ruta de flujo actualizada desde el adaptador 101, como se muestra en 1001. La solicitud pasa el ID de flujo del cliente (por ejemplo, el ID que corresponde al flujo existente que vincula al cliente con el servidor) y versión de flujo al adaptador 101 de tal manera que el servidor reconocerá la nueva ruta de flujo como que pertenece a la misma sesión de cliente lógico, pero siendo más reciente (por ejemplo, una versión actualizada del flujo existente). En realizaciones que usan señales de sesión la solicitud de reubicación pasa la señal de sesión previa del cliente al adaptador. Es importante usar la misma señal de sesión de tal manera que se pueda mantener una conexión entre el cliente y el mismo servidor dedicado al que el cliente ya está conectado. El único cambio que finalmente se produce es un cambio en relé.

El adaptador 101, como se muestra en 1002, luego envía la solicitud de reubicación del cliente, junto con el ID de flujo y versión de flujo, al servidor 102 maestro. El servidor 102 maestro, como se muestra en 1101 de la figura 11, luego responde con una nueva ruta de flujo que tiene el mismo ID de flujo, pero con una nueva ruta de flujo (por ejemplo, un conjunto diferente de relés) que lleva al mismo servidor dedicado s_j y una versión de flujo incrementada, por lo que se puede determinar que esta ruta es una versión más reciente de la ruta de flujo existente. De manera similar en realizaciones que usan señales de sesión el adaptador envía la solicitud de reubicación, incluyendo la señal de sesión previa, al servidor maestro. El servidor maestro responde con una nueva señal de sesión que tiene las mismas propiedades que la señal de sesión previa, excepto que la nueva señal de sesión apunta a uno o más relés nuevos que han sido seleccionados por el servidor maestro, y el número de secuencia de sesión se incrementa.

A continuación, como se muestra en 1102, la ruta de flujo actualizada (o en algunas realizaciones la señal de sesión) se envía desde el adaptador 101 al cliente 103. Desde allí, como se muestra en la figura 12, el cliente 103 usa la ruta de flujo actualizada para crear un paquete 1201 de solicitud, que usa para someterse al mismo proceso como se describe con respecto a la figura 6 para establecer un nuevo flujo. Una vez que el cliente 103 recibe un paquete de respuesta desde el servidor s_j a través de la ruta 1302 de flujo actualizada, como se muestra en la figura 13 (y como se describe más completamente con respecto a la figura 7), se establece el flujo actualizado. Mientras tanto, el intercambio de paquetes de carga útil continúa a través de la ruta de flujo existente solo hasta que el servidor dedicado s_j recibe el paquete de solicitud actualizado que corresponde a la ruta de flujo actualizada transmitida a través del nuevo conjunto de relés r_2, r_3, r_4, en la ruta 1301 actualizada. Después de recibir el paquete de solicitud, el servidor dedicado s_j comienza a enviar paquetes de carga útil al cliente 103 a través de la ruta 1302 de flujo actualizada, y recibe paquetes para esa sesión de cliente solo desde la ruta de flujo actualizada, ignorando cualquier paquete enviado desde el cliente 103 a lo largo de la ruta de flujo previa. El cliente 103 por su parte inicia inmediatamente a enviar paquetes de carga útil de cliente a servidor a lo largo de la ruta de flujo actualizada, mientras que en el período de transición acepta paquetes de carga útil de servidor a cliente desde ya sea la ruta existente o actualizada. Una vez que se establece la ruta actualizada, el cliente expira la ruta previa y deja de aceptar paquetes que se le envían a lo largo de esa ruta, y la ruta actualizada se establece completamente y se completa la transición a la ruta de flujo actualizada.

En el caso de los métodos de flujo de señales de sesión, las solicitudes de reubicación se manejan como inicio de conexión con señales de sesión como se describió anteriormente. Excepto que los paquetes se envían al relé identificado por la señal de inicio de sesión durante una cantidad de tiempo limitada. Una vez que expira esa cantidad de tiempo, en la segunda fase de envío de paquetes, los paquetes tienen el prefijo de una nueva señal de continuación de sesión en lugar de la nueva señal de inicio de sesión. Si un relé recibe un paquete que tiene el prefijo de una señal de continuación de sesión, el relé ejecuta una verificación de firma y autentificación para asegurarse de que la señal de continuación de sesión sea válida y fuera generada por el servidor maestro, después de esto desencripta la señal de continuación de sesión. Una vez desencriptada, el relé verifica para ver si el ID de sesión en la señal de continuación de sesión ya existe en la caché del relé.

En el caso del método de conexión de señal de sesión, si el ID de sesión existe en una caché del relé, el relé procede a eliminar el prefijo (por ejemplo, la señal de continuación de sesión) y lo reemplaza con el ID de sesión y número de secuencia de sesión antes de pasar el paquete al servidor dedicado o a uno o más relés. Si el ID de sesión no existe en la caché, el paquete se ignora.

De acuerdo con aspectos del método de conexión de sesión de señal de la presente divulgación los paquetes enviados a través del nuevo relé tendrán todos un número de secuencia de sesión incrementado, de tal manera que el servidor dedicado pueda comparar el número de Secuencia de sesión previo y el nuevo número de secuencia de sesión para determinar que el nuevo relé es más reciente que el antiguo relé. En otras palabras, el número de secuencia de sesión

cambia (por ejemplo, aumenta) cada vez que se produce una reubicación de relé. Por ejemplo, el número de secuencia de sesión podría contar desde 1 a 2 después de que se produzca una reubicación de relé. Esto permite que el servidor dedicado compare los números de secuencia de sesión y solo acepte paquetes que tengan el número de secuencia de sesión más reciente (es decir, 2, en este caso).

El servidor dedicado al que el cliente se ha conectado para ejecutar el método de conexión de señal de sesión de acuerdo con aspectos alternativos de la presente divulgación, siempre verificará el ID de sesión (por ejemplo, para conocer cuál cliente es cuál) y también el número de secuencia de sesión. Es importante que el servidor dedicado verifique el número de secuencia de sesión de tal manera que, en caso de una reubicación de sesión, el servidor dedicado conocerá cuáles paquetes escuchar y cuáles ignorar. Por ejemplo, aunque los paquetes todavía serán enviados desde el cliente al servidor dedicado durante el proceso de reubicación de una conexión a un servidor a través de un nuevo relé, tan pronto como los paquetes comiencen a aparecer desde el nuevo relé, los paquetes recibidos desde el antiguo relé se pueden ignorar. El servidor dedicado conocerá cuáles paquetes conservar con base en el número de secuencia de sesión con base en lo reciente de ese número. Por ejemplo, si un número de secuencia de sesión inicia en 1 y aumenta a 2 después de una reubicación de sesión, el servidor dedicado (aunque por un tiempo puede recibir paquetes desde ambos relés) solo escuchará paquetes con el número de secuencia de sesión más reciente de 2.

Se contempla que puedan producirse tiempos de espera en cualquier nodo a lo largo de una ruta de flujo. Por ejemplo: si un cliente no recibe ningún paquete desde el primer relé durante alguna duración de tiempo (por ejemplo, 1-10 y preferiblemente 5 segundos) se expira; si el servidor no recibe ningún paquete desde el relé anterior para un flujo particular durante alguna duración de tiempo (por ejemplo, 1-10 y preferiblemente 5 segundos), expira y retira esa entrada de flujo; y si un relé no recibe ningún paquete desde el nodo previo durante alguna duración de tiempo (por ejemplo, 1-10 y preferiblemente 5 segundos), o no recibe ningún paquete previo desde el nodo siguiente durante alguna duración de tiempo (por ejemplo, 1-10 y preferiblemente 5 segundos), expira y retira esa entrada de flujo.

La protección de reproducción, mencionada brevemente en los párrafos precedentes, detiene que un atacante registre un paquete válido y que lo reproduzca de vuelta más tarde en un ataque en un nodo (por ejemplo, un cliente, un relé, o un servidor). Para habilitar la protección de reproducción, se pueden implementar varias medidas. Por ejemplo, los paquetes encriptados y/o firmados se pueden enviar con números de secuencia de 64 bits que inician en cero y aumentan con cada paquete enviado. Los números de secuencia pueden incluirse en el encabezado de paquete y pueden ser leídos por el nodo que recibe el paquete (por ejemplo, antes de la desencriptación o verificación de firma). Además, los números de secuencia se pueden usar como el *nonce* para la encriptación de paquetes, por lo que cualquier modificación del número de secuencia falla la verificación de firma de encriptación.

La protección de reproducción de este modo opera como sigue. Primero, los paquetes se reciben y almacenan en un búfer de reproducción del nodo que tiene un tamaño de búfer de reproducción establecido. El tamaño de búfer de reproducción determina una cantidad de paquetes que se pueden almacenar en el búfer de reproducción (por ejemplo, 64-128, 128-256, 256-512, 512-1028 paquetes). El tamaño de búfer de reproducción es específico de implementación. En algunas realizaciones, se soportan paquetes de unos pocos segundos a una tasa de envío típica (20-60 Hz). Por ejemplo, un tamaño de búfer de reproducción de 256 entradas por cliente debería ser suficiente para la mayoría de aplicaciones. Cada paquete recibido tiene un número de secuencia asociado. Cualquier paquete que tenga un número de secuencia que sea más antiguo que el número de secuencia más reciente recibido (por ejemplo, recibido con un paquete), menos el tamaño de búfer de reproducción, se descarta en el lado de receptor.

Por ejemplo, si un tamaño de búfer de reproducción es 100, y el paquete más reciente recibido tiene un número de secuencia de 600, un paquete que tenga un número de secuencia de 599 o menos (es decir, 1 menor que 600 menos 100) se descartaría. Cuando llega un nuevo paquete que tiene un número de secuencia más reciente que el número de secuencia más reciente recibido previamente, el número de secuencia asociado con el nuevo paquete se actualiza en el lado de receptor y se acepta el nuevo paquete. Si llega un paquete que está dentro del tamaño de búfer de reproducción del número de secuencia más reciente, se acepta solo si su número de secuencia aún no se ha recibido. Si se recibe un paquete que tiene un número de secuencia que ya se ha recibido, ese paquete se ignora.

Implementación

Aspectos de la presente divulgación se pueden implementar en un aparato de ordenador configurado adecuadamente tal como un servidor (por ejemplo, servidor de adaptación, servidor maestro, etc.), ordenador personal, relé de red y similares. La figura 15 ilustra un diagrama de bloques de un sistema 1500 que puede usarse para implementar un método de comunicación por relé de nodos de acuerdo con un aspecto de la presente divulgación. El aparato 1500 generalmente puede incluir un módulo 1501 de procesador y una memoria 1505.

El módulo 1501 de procesador puede incluir uno o más núcleos de procesador. El módulo 1501 de procesador puede incluir múltiples núcleos de procesador, por ejemplo, si va a ser implementado el procesamiento en paralelo. Ejemplos de procesadores multinúcleo adecuados, incluyen, pero no se limitan a procesadores de doble núcleo, procesadores de cuatro núcleos, arquitecturas de procesador que tienen un procesador principal y uno o más coprocesadores,

arquitecturas de procesador celular, y similares. La capacidad de procesar datos en paralelo ahorra valioso tiempo de procesamiento, llevando a un sistema más eficiente y simplificado para reconocimiento de emociones.

La memoria 1505 puede ser en la forma de un circuito integrado, por ejemplo, RAM, DRAM, ROM, y similares. La memoria 1505 también puede ser una memoria principal que es accesible por todos los módulos de procesador. En algunas realizaciones, el módulo 1501 de procesador puede tener memorias locales asociadas con cada núcleo. Se puede almacenar un programa 1503 en la memoria 1505 principal en la forma de instrucciones legibles por procesador que se pueden ejecutar en los módulos de procesador. El programa 1503 puede configurarse para implementar el método para comunicación entre nodos usando señales como se describió anteriormente y en las figuras 1, hasta 13. El programa 1503 puede escribirse en cualquier lenguaje legible por procesador adecuado, por ejemplo, C, C++, JAVA, Assembly, MATLAB , FORTRAN, y un número de otros lenguajes. Durante la operación del programa los paquetes 1507 de datos pueden almacenarse en la memoria antes de que se transmitan a otro nodo. El programa puede hacer que una base 1508 de datos almacene paquetes de datos y se indexe de acuerdo con el ID de flujo y/o una versión de flujo en la memoria 1505. Adicionalmente la base de datos puede almacenar claves públicas o claves privadas para desencriptación de señales. En algunas realizaciones donde el sistema 1500 es un servidor maestro la memoria 1505 puede almacenar una clave privada para la encriptación de unos datos de señal. Adicionalmente en el caso de un servidor maestro, el programa 1503 puede hacer que el sistema trace una ruta de flujo a través de la red 1550 para un dispositivo cliente usando la interfaz 1523 de red y proporcione una serie de señales de flujo que definen esa ruta de flujo. Durante la ejecución del programa 1503, se pueden cargar porciones de código y/o datos de programa en la memoria o en los almacenes locales de núcleos de procesador para el procesamiento paralelo mediante múltiples núcleos de procesador.

El aparato 1500 también puede incluir funciones 1509 de soporte bien conocidas, tales como elementos 1511 de entrada/salida (E/S), fuentes de alimentación (P/S) 1513, un reloj (CLK) 1515, y una caché 1517. El aparato 1500 puede incluir opcionalmente un dispositivo 5119 de almacenamiento masivo tal como una unidad de disco, unidad de CD-ROM, unidad de cinta, o similar para almacenar programas y/o datos. El aparato 1500 puede incluir opcionalmente una unidad 1521 de visualización para facilitar la interacción entre el aparato y un usuario. La unidad 1521 de visualización puede ser en la forma de un tubo de rayos catódicos (CRT) o una pantalla de panel plano que muestra texto, números, símbolos gráficos o imágenes. La interfaz 1525 de usuario puede incluir un teclado, ratón, palanca de mando, pantalla táctil, almohadilla táctil, u otro dispositivo que puede usarse en conjunto con una interfaz gráfica de usuario (GUI).

Los componentes del aparato 1500, incluyendo el procesador 1501, memoria 1505, funciones 1509 de soporte, dispositivo 1519 de almacenamiento masivo, interfaz 1525 de usuario, interfaz 1523 de red, y pantalla 1521 pueden conectarse operativamente entre sí a través de uno o más buses 1527 de datos. Estos componentes pueden implementarse en hardware, software o firmware o alguna combinación de dos o más de estos.

De este modo, se han divulgado composiciones y métodos específicos para establecer flujos para intercambio de paquetes bidireccional. Además, al interpretar la divulgación todos los términos deben interpretarse de la manera más amplia posible consistente con el contexto. En particular los términos "comprende" y "que comprende" deben interpretarse como que se refieren a los elementos, componentes, o etapas de una manera no exclusiva, indicando que los elementos, componentes, o etapas a los que se hace referencia pueden estar presentes, o usarse, o combinarse con otros elementos, componentes, o etapas a los que no se hace referencia expresamente.

REIVINDICACIONES

1. Un sistema de nodos que comprende:

un procesador (1501); y

una memoria (1505);

en donde el sistema de nodos está configurado para implementar un método para comunicación por relé de nodos que comprende:

a) recibir una descripción de una entrada de flujo en un paquete desde otro nodo, caracterizado por:

la descripción de la entrada de flujo que incluye una dirección en un flujo, un identificador (ID) de flujo de la entrada de flujo, una versión de flujo, una información de dirección y puerto para uno u otros más nodos en el flujo, y una clave privada, en donde el paquete desde el otro nodo incluye una marca de tiempo de expiración, y en donde la información de dirección y puerto para uno u otros más nodos en el flujo incluye una dirección de protocolo de internet, IP, y puerto para un siguiente nodo en el flujo;

b) almacenar la entrada de flujo y la clave privada en una base de datos indexada por el ID de flujo;

c) recibir un paquete, en donde el paquete comprende un código de autentificación y datos de paquete que incluyen información de secuencia de paquete y el ID de flujo del paquete;

d) realizar una búsqueda en la base de datos de una entrada de flujo que corresponde al ID de flujo del paquete; e

e) ignorar el paquete o reenviar el paquete a la dirección de IP del siguiente nodo en el flujo, dependiendo del resultado de la búsqueda.

2. El sistema de la reivindicación 1 en donde la etapa e) comprende ignorar el paquete si no existe ninguna entrada de flujo.

3. El sistema de la reivindicación 1 en donde realizar la búsqueda en la base de datos en la etapa d) comprende además verificar que el código de autentificación del paquete indica que los datos de paquete fueron firmados con una clave privada de flujo que coincide con la clave privada en la entrada de flujo en la base de datos.

4. El sistema de la reivindicación 3 en donde la etapa e) comprende ignorar el paquete si los datos de paquete no fueron firmados con una clave privada de flujo que coincida con la clave privada en la entrada de flujo.

5. El sistema de la reivindicación 3 en donde la etapa e) comprende además probar la información de secuencia de paquetes contra un búfer de protección de reproducción para paquetes recibidos desde el siguiente nodo, si los datos de paquete fueron firmados con la clave privada de flujo que coincide con la clave privada en la entrada de flujo en la base de datos y si el paquete ya se ha recibido, o es antiguo, ignorar el paquete, en donde la etapa comprende además reenviar el paquete sin modificación a un nodo previo y/o siguiente nodo en el flujo, si los datos de paquete fueron firmados con la clave privada de flujo que coincide con la clave privada en la entrada de flujo en la base de datos y el paquete aún no se ha recibido y el paquete no es antiguo.

6. El sistema de la reivindicación 5 que comprende además actualizar la marca de tiempo en la entrada de flujo del último paquete recibido a la marca de tiempo actual, o retirar la entrada de flujo desde la base de datos si los paquetes con el ID de flujo que corresponde a la entrada de flujo no se han recibido durante un período de tiempo predeterminado desde el nodo previo y/o el siguiente nodo, y dejar de reenviar paquetes con el ID de flujo que corresponde a la entrada de flujo retirada.

7. El sistema de la reivindicación 1 en donde la base de datos también está indexada por una versión de flujo en donde el paquete incluye la versión de flujo, y en donde realizar la búsqueda en la base de datos incluye buscar usando la versión de flujo en el paquete.

8. El sistema de la reivindicación 1 en donde la etapa a) comprende recibir la descripción de la entrada de flujo desde un servidor (102) maestro.

9. El sistema de la reivindicación 1, en donde el paquete desde el otro nodo incluye una dirección de IP y puerto de nodo previo.

10. El sistema de la reivindicación 1, que comprende además intentar desencriptar una primera señal de flujo en el paquete desde el otro nodo usando una clave privada de nodo y una clave pública de servidor maestro; y modificar el

paquete retirando la primera señal de flujo y reenviando el paquete modificado resultante a una dirección de IP y puerto de siguiente nodo en la primera señal de flujo cuando el intento de desencriptar la primera señal de flujo tiene éxito.

11. Un medio legible por ordenador no transitorio que tiene instrucciones legibles por ordenador incorporadas en el mismo, estando las instrucciones legibles por ordenador configuradas para implementar un método de comunicación por relé de nodos, cuando se ejecuta el método de comunicación por relé de nodos que comprende:

a) recibir una descripción de una entrada de flujo en un paquete desde otro nodo, caracterizado por:

la descripción de la entrada de flujo que incluye una dirección en un flujo, un identificador (ID) de flujo de la entrada de flujo, una versión de flujo, una información de dirección y puerto para uno u otros más nodos en el flujo, y una clave privada, en donde el paquete desde el otro nodo incluye una marca de tiempo de expiración, y en donde la información de dirección y puerto para uno u otros más nodos en el flujo incluye una dirección de protocolo de internet, IP, y puerto para un siguiente nodo en el flujo;

b) almacenar la entrada de flujo y la clave privada en una base de datos indexada por el ID de flujo;

c) recibir un paquete, en donde el paquete comprende un código de autentificación y datos de paquete que incluyen una secuencia de paquetes y el ID de flujo del paquete;

d) realizar una búsqueda en la base de datos de una entrada de flujo que corresponde al ID de flujo del paquete; e

e) ignorar el paquete o reenviar el paquete a la dirección de IP del siguiente nodo en el flujo, dependiendo del resultado de la búsqueda.

12. El medio legible por ordenador no transitorio de la reivindicación 11 en donde realizar la

búsqueda en la base de datos de acuerdo con la etapa d) comprende además verificar que el código de autentificación del paquete indica que los datos de paquete fueron firmados con una clave privada de flujo que coincide con la clave privada en la entrada de flujo en la base de datos.

13. Un sistema que comprende un sistema de nodos configurado para realizar el método de la reivindicación 1 a 10 y un sistema (102) de servidor maestro que comprende:

un procesador (1501); y

una memoria (1505);

caracterizado porque el sistema de servidor maestro está configurado para implementar un método para comunicación por relé de nodos con dicho sistema de nodos que comprende:

a) recibir información de nodo desde nodos en una red (1550);

b) determinar una o más rutas (1003) de flujo entre un nodo inicial y un nodo final a partir de información de nodo en donde cada ruta de flujo de la una o más rutas de flujo incluye uno o más nodos en la red distintos del nodo inicial y el nodo final;

c) enviar información de ruta de flujo a uno o más nodos, en donde la información de ruta de flujo incluye una o más señales de flujo que corresponden a cada nodo de uno o más nodos en una ruta de flujo de la una o más rutas de flujo, y una señal de flujo para el servidor y en donde cada señal de flujo incluye un identificador (ID) de flujo, una versión de flujo, una marca de tiempo de expiración, una clave privada de flujo, una dirección de protocolo de internet, IP, y un puerto para un siguiente nodo y una dirección de IP y un puerto para un nodo previo.

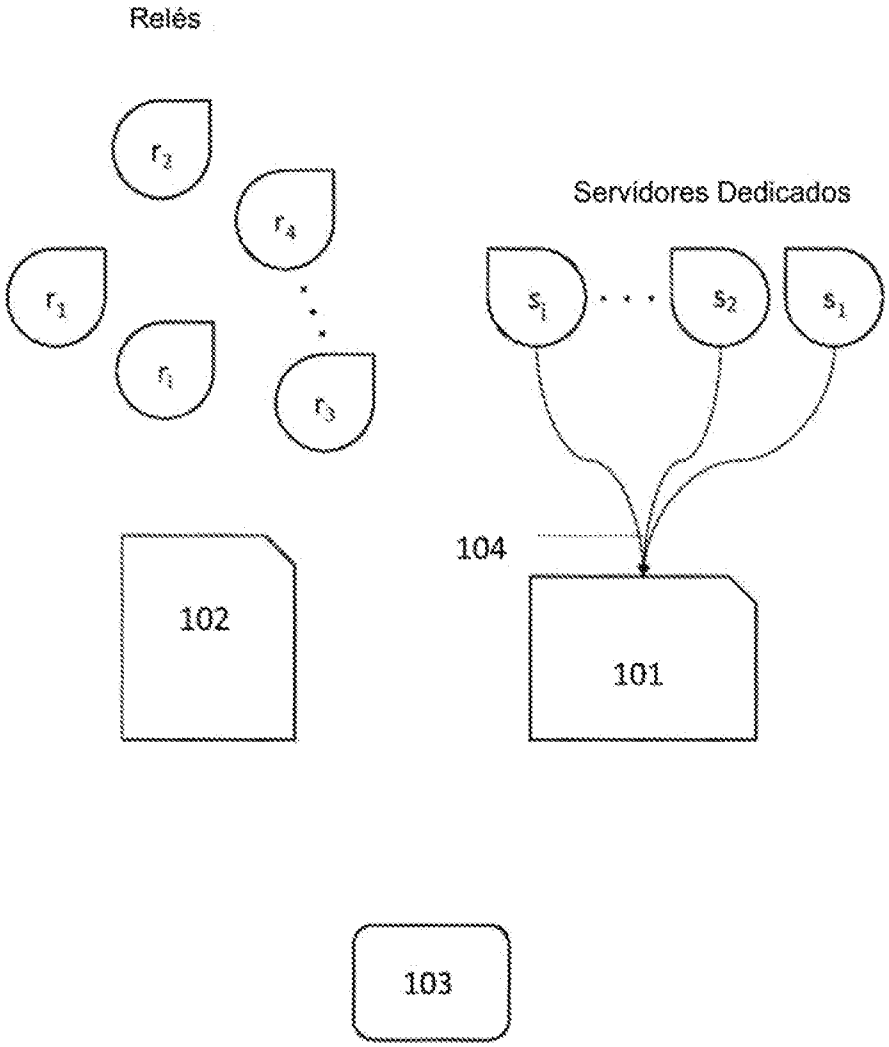

Figura 1

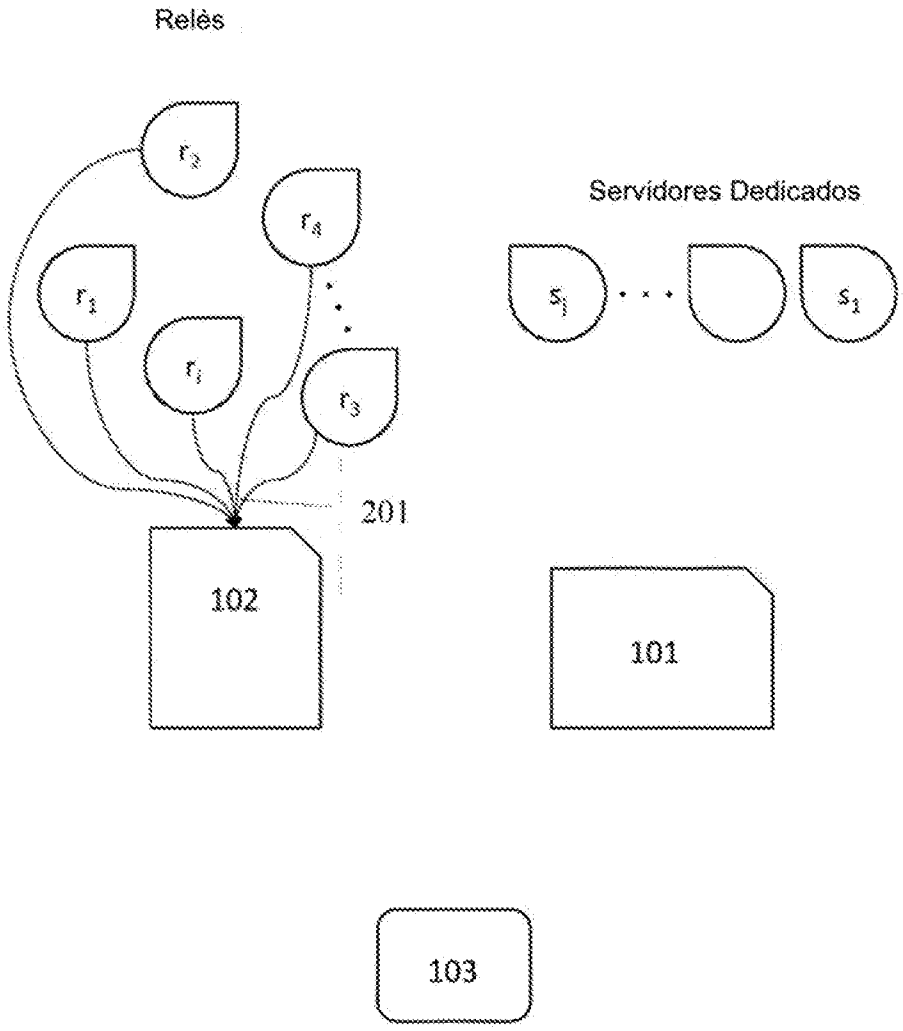

Figura 2

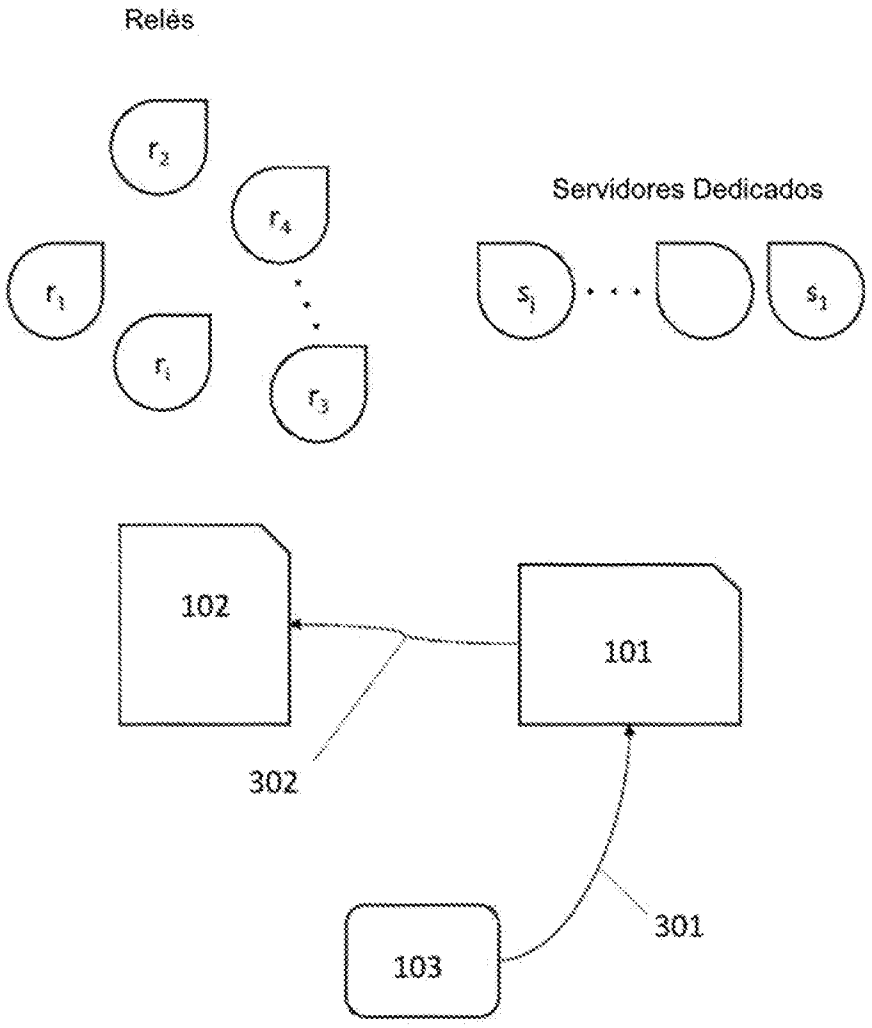

Figura 3

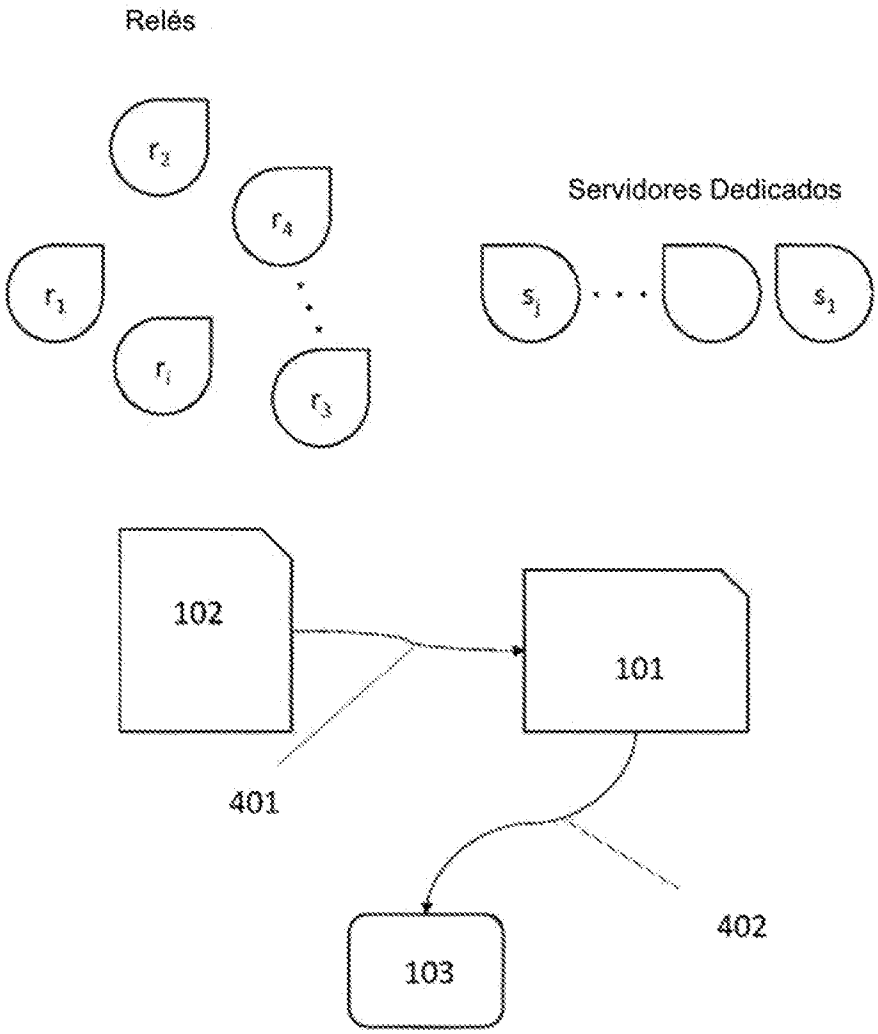

Figura 4

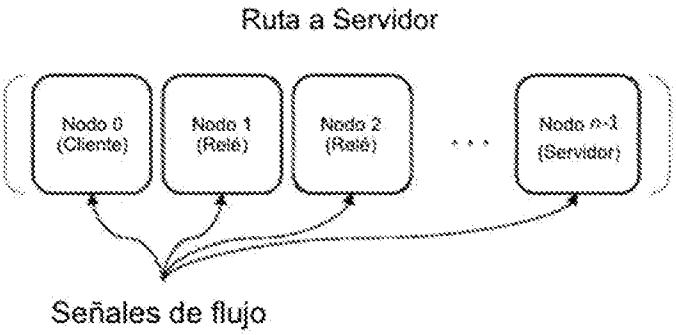

Figura 5A

Figura 5B

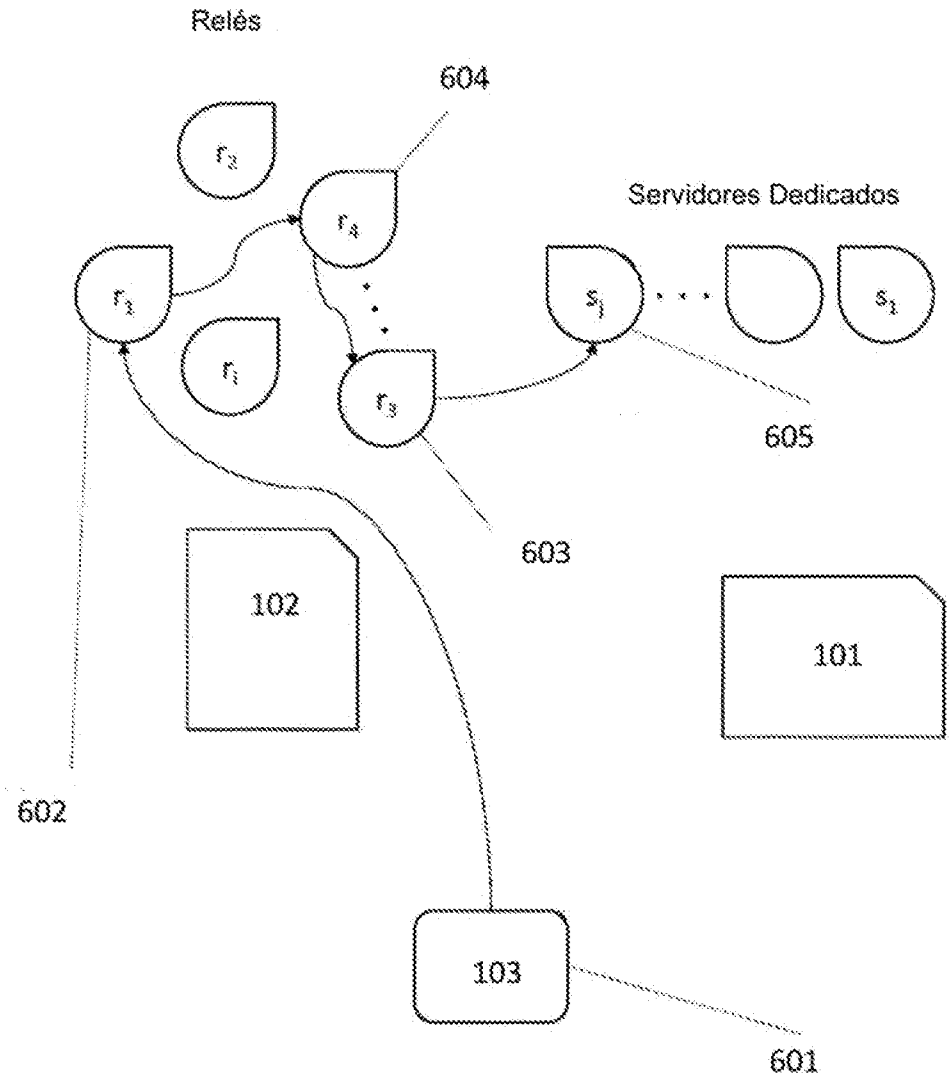

Figura 6

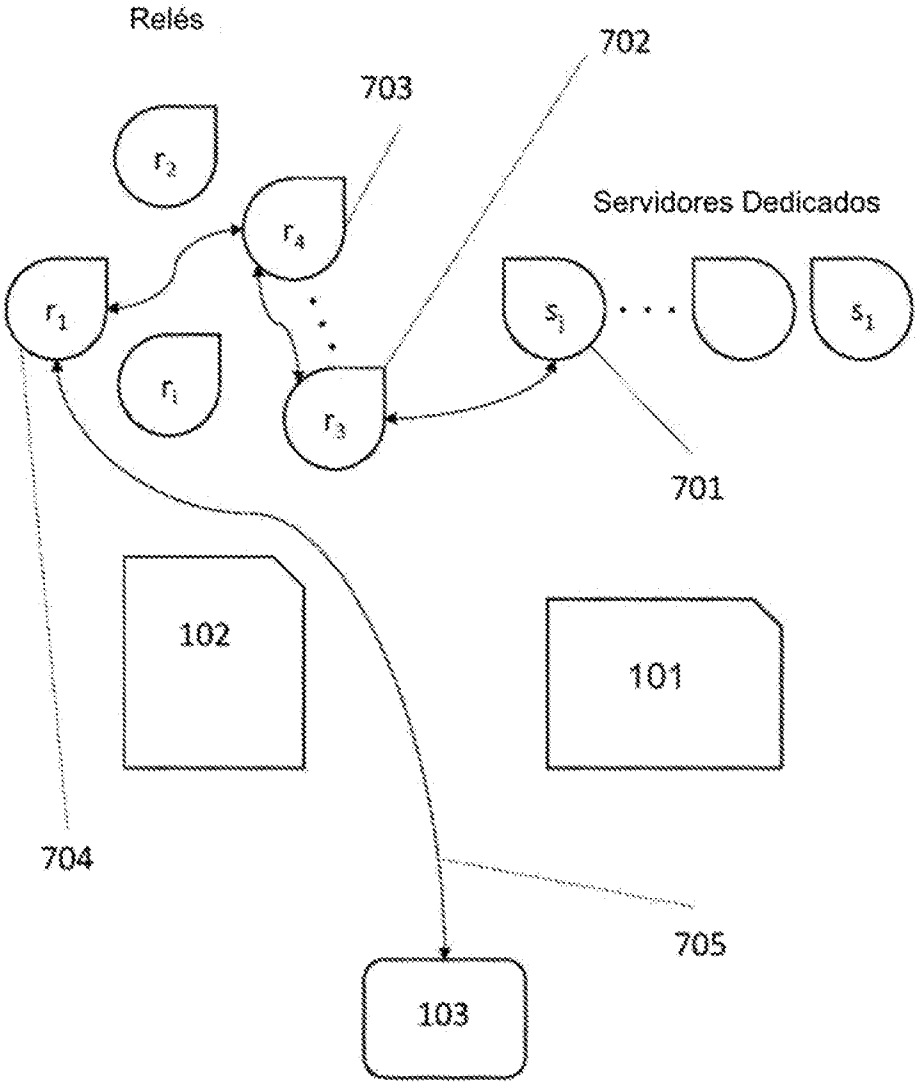

Figura 7

Clave	Valor
{ID de Flujo, Versión de Flujo}	Entrada
~	~
~	~
~	~

Figura 8A

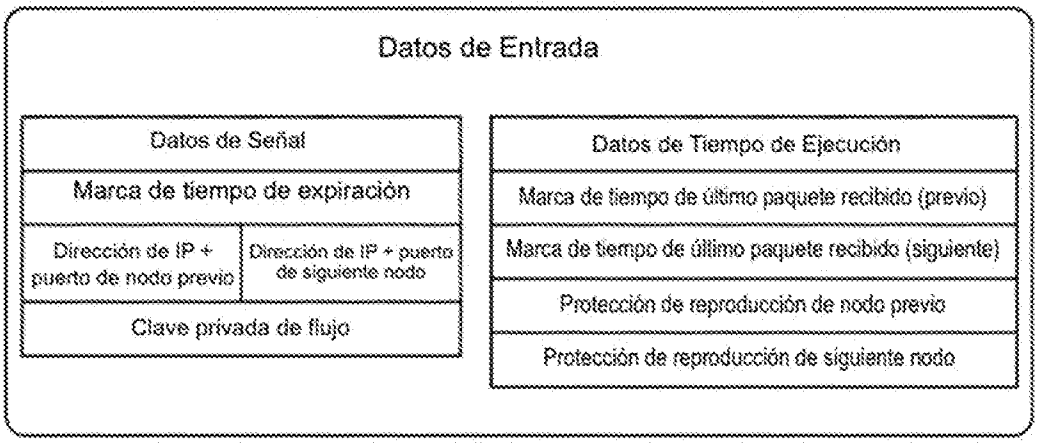

Figura 8B

Clave	Valor
ID de Flujo	Entrada
ˇ	ˇ
ˇ	ˇ
ˇ	ˇ

Figura 9A

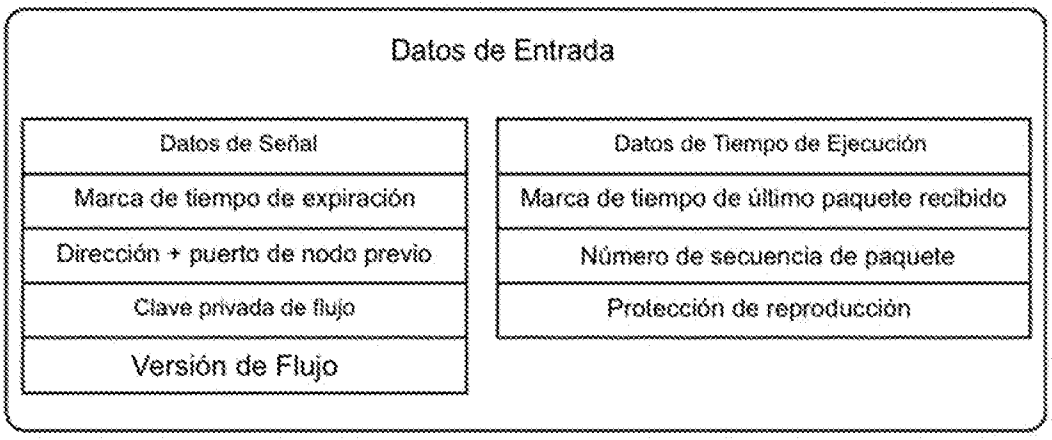

Figura 9B

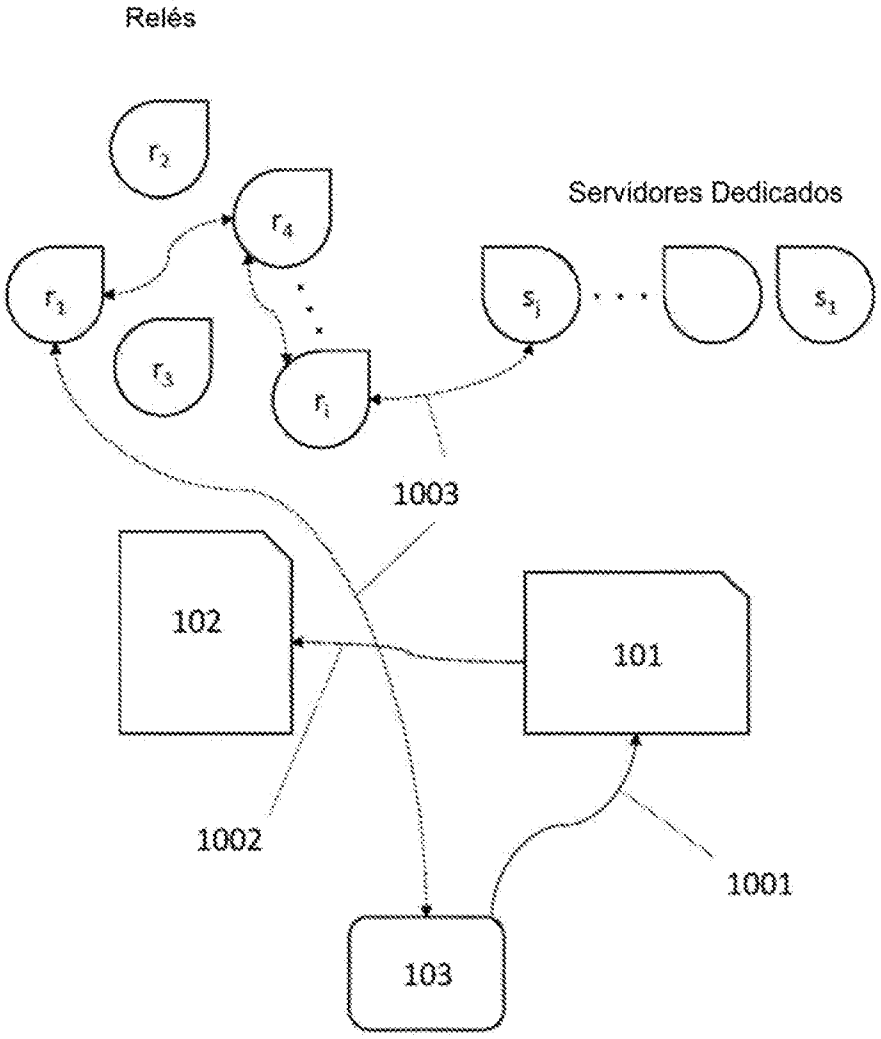

Figura 10

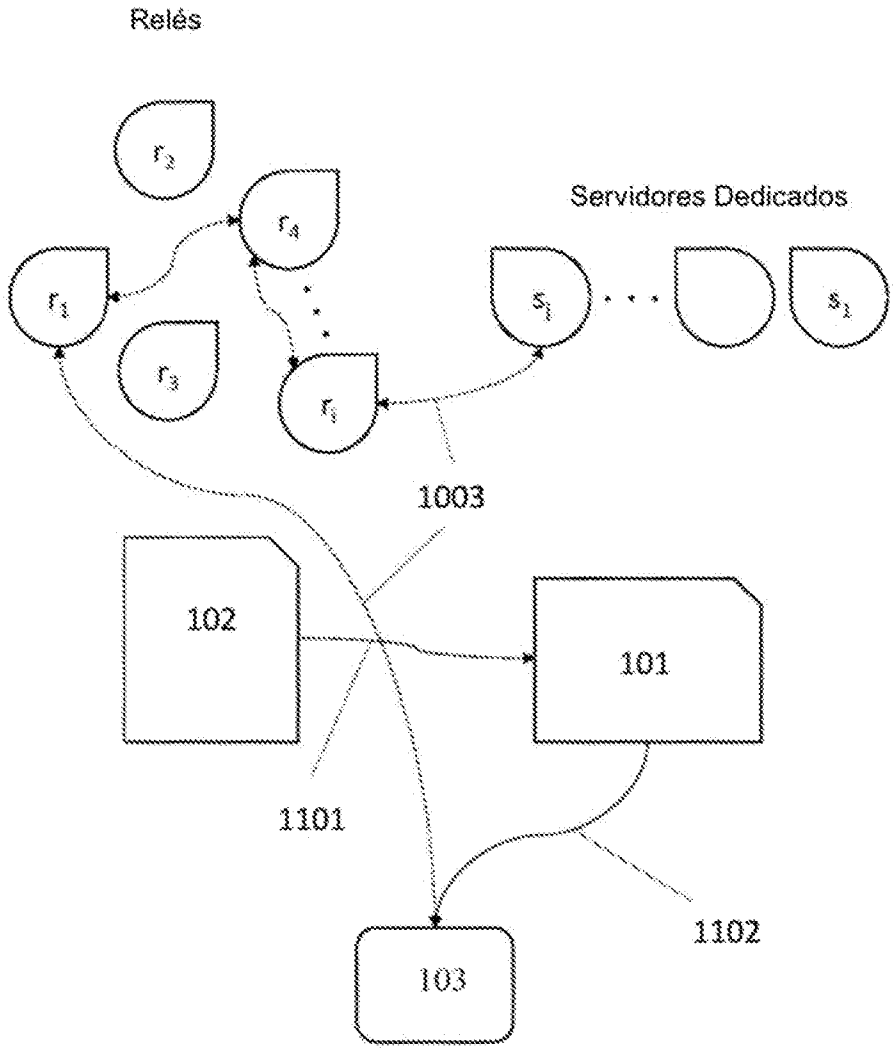

Figura 11

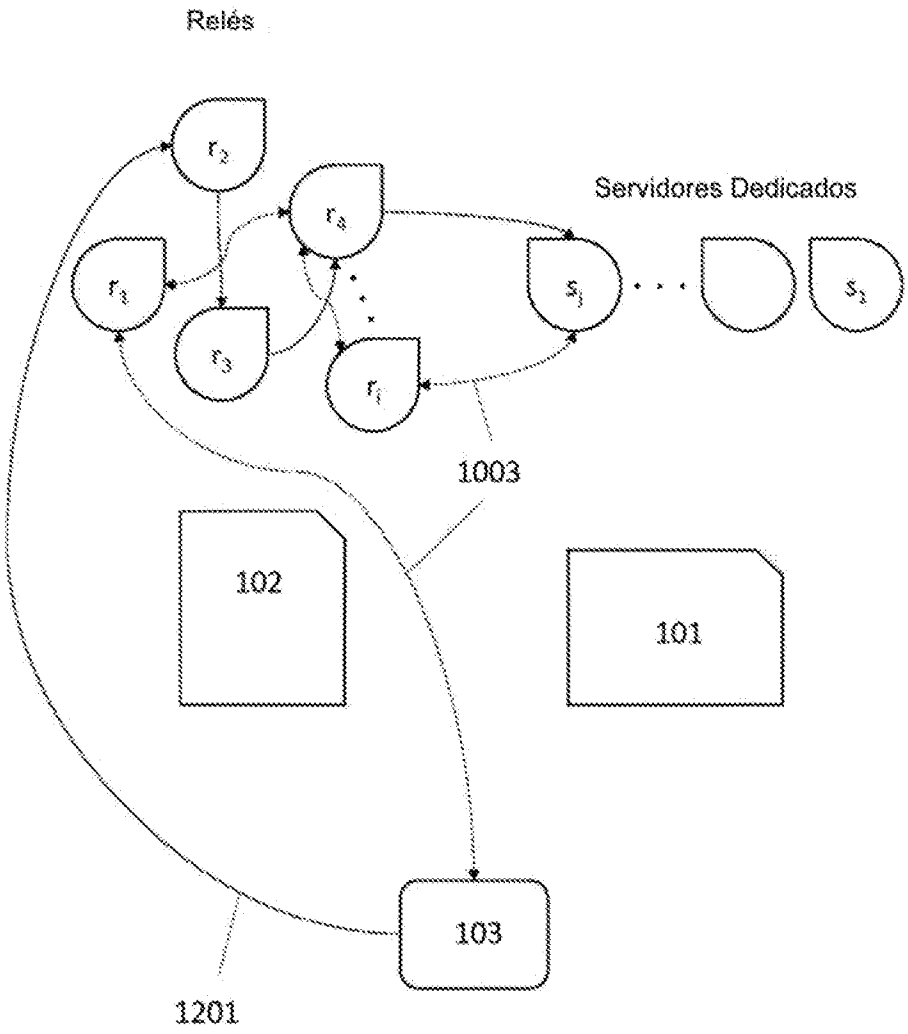

Figura 12

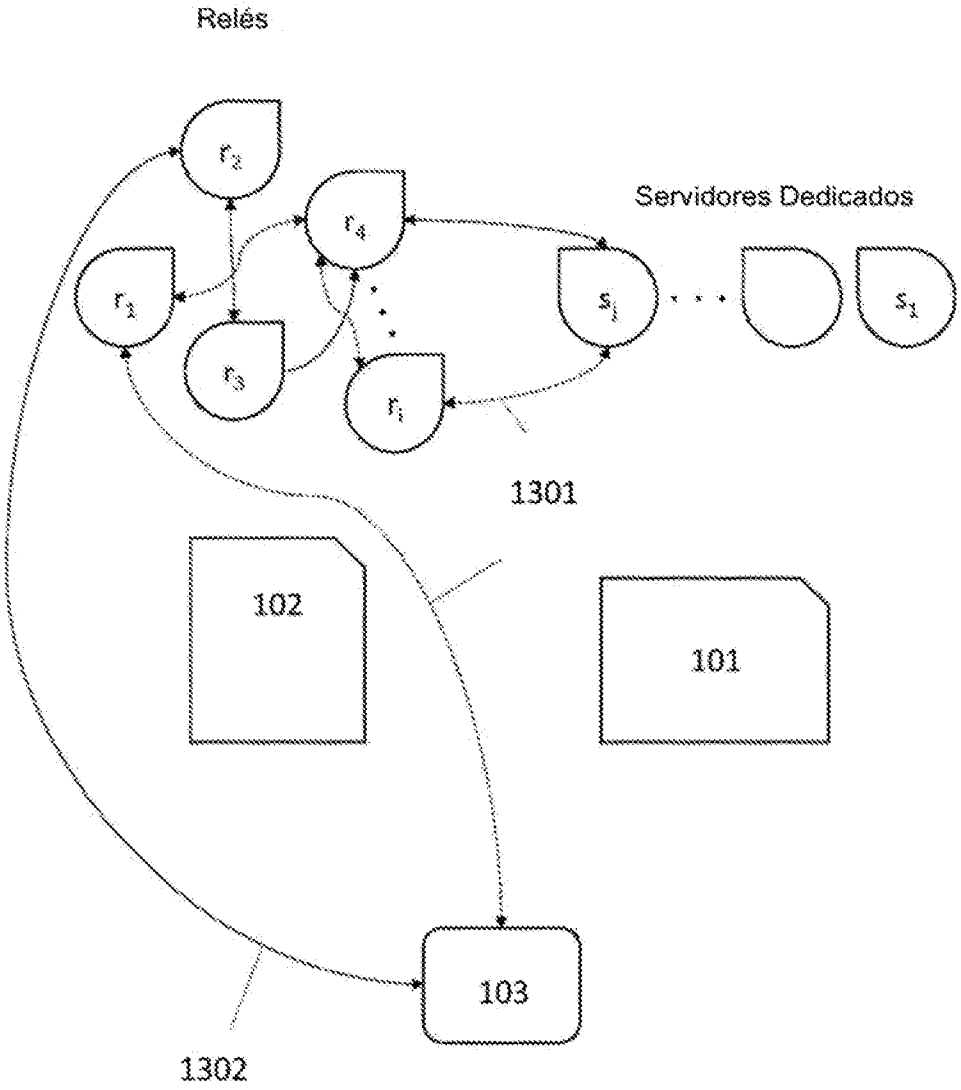

Figura 13

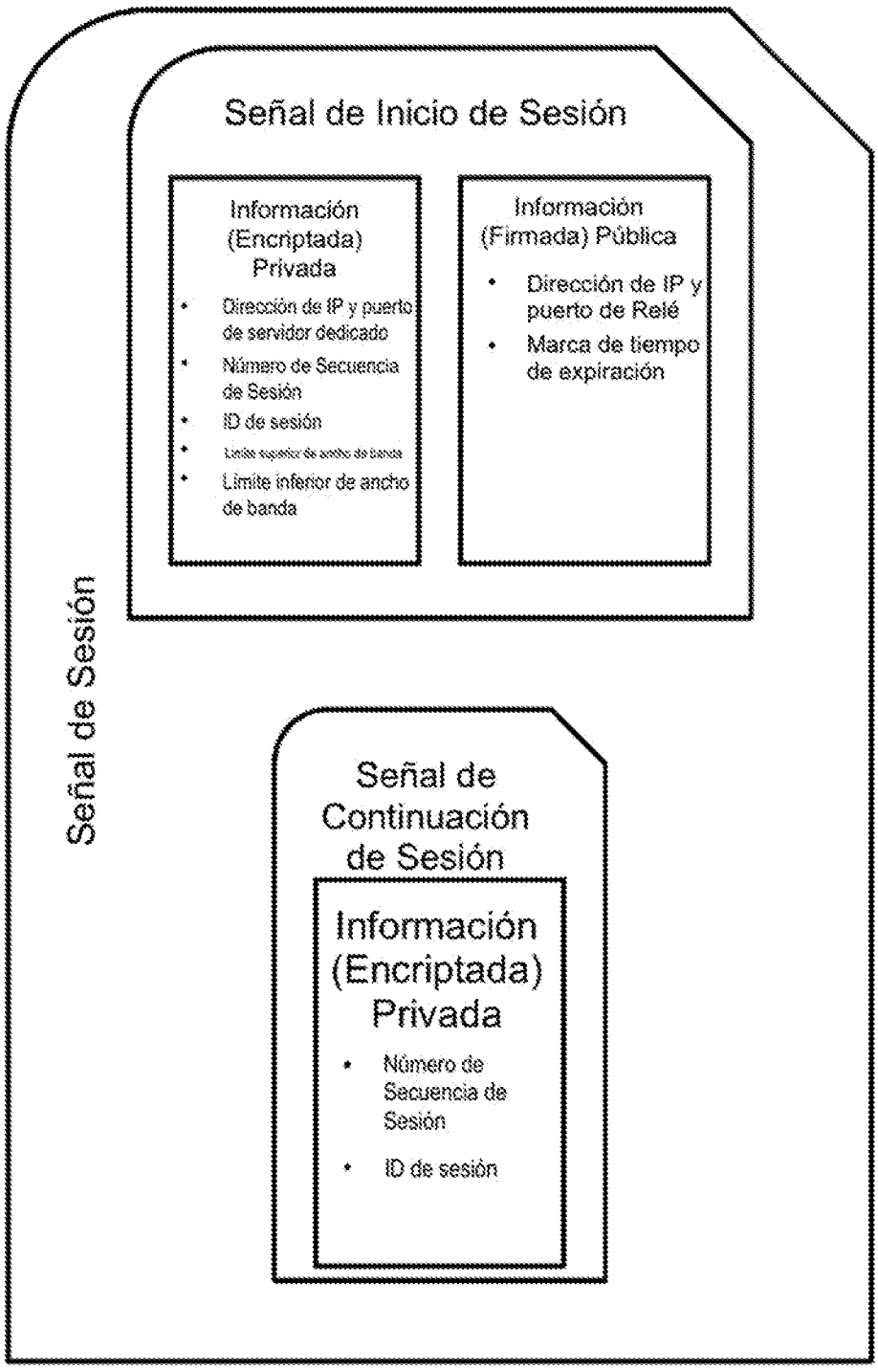
Señal de Sesión
Señal de Inicio de Sesión
Información (Encriptada) Privada
Dirección de IP y puerto de servidor dedicado
Número de Secuencia de Sesión
ID de sesión
Límite superior de ancho de banda
Límite inferior de ancho de banda
Información (Firmada) Pública
Dirección de IP y puerto de Relé
Marca de tiempo de expiración
Señal de Continuación de Sesión
Información (Encriptada) Privada
Número de Secuencia de Sesión
ID de sesión

Figura 14

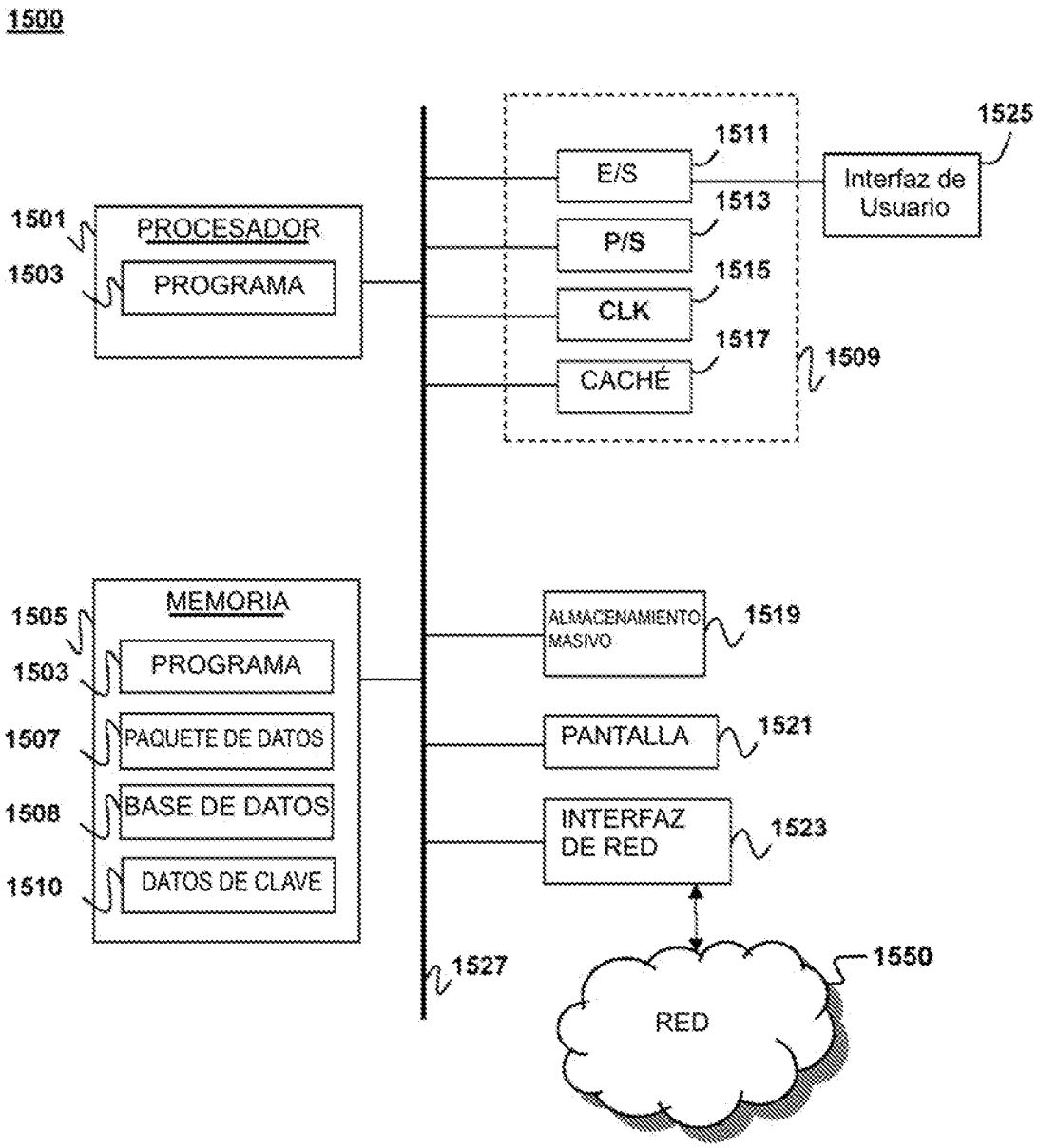
1500
1501
PROCESADOR
1503
PROGRAMA
1511
E/S
1513
P/S
1515
CLK
1517
CACHÉ
1509
1525
Interfaz de Usuario
1505
MEMORIA
1503
PROGRAMA
1507
PAQUETE DE DATOS
1508
BASE DE DATOS
1510
DATOS DE CLAVE
ALMACENAMIENTO MASIVO
1519
PANTALLA
1521
INTERFAZ DE RED
1523
1527
1550
RED

FIG. 15