(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 464 163**

(51) Int. CI.:

H04L 29/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **21.02.2006 E 06707225 (6)**

(97) Fecha y número de publicación de la concesión europea: **07.05.2014 EP 1891784**

(54) Título: **Sistema y procedimiento de comunicación en red segura**

(30) Prioridad:

03.06.2005 IE 20050376

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
30.05.2014

(73) Titular/es:

ASAVIE R&D LIMITED (100.0%)
24 Fleming Court
Dublin 4 , IE

(72) Inventor/es:

MAHER, THOMAS

(74) Agente/Representante:

CURELL AGUILÁ, Mireia

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

Sistema y procedimiento de comunicación en red segura.

La presente invención se refiere a un sistema y a un procedimiento de comunicación en red segura. Particularmente, se refiere a un sistema y a un procedimiento de comunicaciones para posibilitar que clientes seguros obtengan un acceso seguro y transparente a un servidor remoto a través de una red insegura.

Se ha reconocido ampliamente que la conexión de un ordenador para comunicarse con una red insegura, tal como Internet, representa un riesgo de seguridad. Para llevar a cabo tareas útiles, dicho ordenador debe reaccionar necesariamente a datos que son recibidos desde servidores remotos a través de la red. Los atacantes maliciosos pueden aprovechar deficiencias del sistema operativo y aplicaciones del ordenador para reaccionar de tal manera que lleven a cabo funciones que podrían comprometer la seguridad u operatividad del ordenador. Por consiguiente, es normal que un ordenador privado o red de ordenadores no se conecte directamente a una red pública. Por el contrario, se realiza una conexión a la red a través de alguna interfaz restrictiva que controla la naturaleza de los datos que se pueden intercambiar entre los ordenadores privados y la red no asegurada. La configuración típica de los anfitriones y redes es que los anfitriones de cliente sean “internos” o de confianza, mientras que los servidores son “externos” o no fiables.

Dos de los procedimientos existentes para posibilitar un acceso transparente y seguro por pasarela entre redes se realizan por medio de un proxy (por ejemplo, según se da a conocer en los documentos US-A-5.623.601 y US-A-5.781.550) y mediante la traducción de direcciones de red (NAT) (por ejemplo, según se da a conocer en el documento US-A-5.793.763). En muchos casos, estas disposiciones proporcionan un servicio adecuado a usuarios y ordenadores “internos”. No obstante, el servicio proporcionado no es el mismo que el de una conexión de red directa. En el caso de un proxy, normalmente se permiten solo conexiones a una gama configurable de puertos bien conocidos en servidores remotos. Esto puede permitir que un administrador de la red restrinja los tipos de servicio a los que puede acceder un usuario de la red, y las máquinas de cliente permanezcan ocultas para los servidores externos.

Estos sistemas conocidos permiten que un usuario de red acceda a servicios externos tales como la Red Informática Mundial (*Worldwide Web*), el correo electrónico de Internet, y otros, y por lo tanto son selectivamente transparentes a protocolos de la capa de aplicación de OSI (Capa 7). No obstante, no proporcionan conexiones transparentes para protocolos de red de nivel inferior. En particular, en la capa de transporte de OSI (Capa 4), en la que se usa comúnmente el Protocolo de Control de Transmisión (TCP), y en la capa de red de OSI (Capa 3), en la que se usa comúnmente el Protocolo de Internet (IP), estos sistemas carecen de transparencia. Esto puede constituir una fuente de problemas cuando la red no está funcionando normalmente, tal como, por ejemplo, cuando falla el establecimiento de una conexión TCP.

La implementación conocida de una pasarela proxy transparente no es totalmente transparente, particularmente con respecto a los paquetes de control TCP/IP. Por ejemplo, considérese el caso en el que un cliente completa el establecimiento de la conexión entre él mismo y un proxy, antes de que el proxy intente establecer una conexión con el anfitrión de destino. Si la conexión del proxy falla, entonces la conexión establecida con el anfitrión se desconecta. El cliente se hace por lo tanto una idea incorrecta de la red, lo cual para el descubrimiento de servicios por parte de clientes de servicios del servidor tiene una importancia crucial. En este ejemplo, el cliente percibe el puerto de servicio como abierto, lo cual no es así.

Los desarrollos más recientes en el acceso remoto han producido lo que se describe en ocasiones como proxys o pasarelas inversos, en donde clientes “externos” a la red son autenticados y se les proporciona un acceso seguro a servidores “internos”. No obstante, las pasarelas no son transparentes, lo cual requiere, en su lugar, software desplegado por el cliente, que negocia conexiones con el proxy inverso.

La solicitud de patente US 2002/0042875 da a conocer también cómo posibilitar que clientes obtengan un acceso seguro y transparente a un servidor remoto a través de una red insegura.

Uno de los objetivos de esta invención consiste en proporcionar una transparencia aumentada, en comparación con sistemas conocidos, al mismo tiempo que manteniendo la seguridad para los sistemas de cliente. Una finalidad particular de las formas de realización preferidas de la invención consiste en proporcionar una transparencia TCP completa en el cliente con el fin de permitir el descubrimiento automático de recursos de la red.

A partir de un primer aspecto, esta invención proporciona un sistema de transmisión de datos para un intercambio seguro de datos usando el Protocolo de Control de Transmisión entre un cliente y un servidor según se expone en la reivindicación 1.

Típicamente, el servidor se conecta al intermediario para el intercambio de datos a través de una red asegurada – es decir, una red “interna”. Para proteger la red interna, la red asegurada se conecta típicamente a la red no asegurada

por medio de uno o más de entre un cortafuegos, un proxy o un dispositivo de traducción de direcciones de red (NAT).

El protocolo seguro puede utilizar un protocolo de seguridad de la capa de transporte (TLS). El protocolo TLS se puede usar para acarrear el protocolo de transporte de hipertexto (HTTPS) seguro. Esto tiene la ventaja de que el puerto 443, puerto usado para HTTPS, se deja típicamente abierto en cortafuegos de la red y proxys. Alternativamente, se puede usar el protocolo HTTP CONNECT.

Los paquetes de control gestionados por el sistema incluyen paquetes TCP SYN y paquetes TCP ACK. Los paquetes de respuesta incluyen paquetes TCP SYN ACK y TCP RST ACK. Estos son los paquetes involucrados principalmente en el establecimiento de una sesión TCP. El paquete de control y el paquete de respuesta pueden ser paquetes del Protocolo de Mensajes de Control de Internet. Dichos paquetes están involucrados típicamente en el descubrimiento automático de recursos de la red.

Mediante la modificación adecuada de las direcciones de origen y de destino dentro de estos paquetes, los datos intercambiados por el cliente y el servidor son similares a aquellos que se intercambiarían si el cliente y el servidor estuvieran en comunicación directa.

Para controlar los servicios y servidores a los que puede acceder un usuario, el intermediario aplica reglas para determinar si el paquete de control se reenvía o no a la gente. Las reglas pueden depender de uno o más de entre el originador del paquete, la dirección del destino del paquete, y el número de puerto de destino del paquete.

Las formas de realización de la invención pueden ser operativas para prestar servicio a múltiples redes virtuales a través de una red de acceso de cliente. En tales formas de realización, la red de acceso de cliente es típicamente responsable de la gestión del acceso del cliente. La red de acceso de cliente puede usar uno o ambos de los servicios normalizados de AAA (Autorización, Autenticación y Contabilidad) y VPN (Red Privada Virtual).

A partir de un segundo aspecto, la invención proporciona un procedimiento de transmisión de datos según se expone en la reivindicación 11.

A continuación se describirá con mayor detalle, a título de ejemplo, una forma de realización de la invención, haciendo referencia a los dibujos adjuntos, en los que:

la figura 1 es un diagrama que ilustra la comunicación entre un cliente y un servidor en un sistema que materializa la invención;

la figura 2 es un diagrama de bloques de un intermediario como componente de un sistema que materializa la invención; y

la figura 3 es un diagrama de bloques de un agente como componente de un sistema que materializa la invención.

Haciendo referencia inicialmente a la figura 1, un cliente 10 que está funcionando dentro de una red segura se conecta a un servidor remoto externo 12 por medio de un intermediario 14 y un agente 16. El intermediario 14 y el agente 16 se comunican a través de una red insegura 18, que lo más habitual es que incluya un enlace de Internet.

En el sistema de la figura 1, el cliente 10 está intentando establecer una comunicación con el servidor 12. El intermediario 14 sirve como pasarela de Internet intermedia, que “engaña” al servidor 12. El intermediario 14 se coordina con el agente remoto 16, usando un protocolo de capa de aplicación segura. El agente 16 se conecta al servidor 12 y actúa como cliente para el intermediario, y por lo tanto debe seguir los protocolos del intermediario, cuya naturaleza se pondrá de manifiesto a partir de la siguiente descripción. Por lo tanto, los datos desde el cliente 10 al servidor 12 fluyen en conexiones desde el cliente 10 al intermediario 14, donde son encaminados y transportados de forma segura al agente 16, en el que los datos se transportan en conexiones hacia delante desde el agente 16 al servidor 12. Los agentes 16 se conectan al intermediario 14 usando un transporte seguro, tal como el TLS, de una manera que facilita que se atraviesen cortafuegos, proxys y dispositivos de NAT intermedios. El planteamiento preferido consiste en usar el puerto 443, permitido típicamente para el tráfico de HTTPS con el fin de proteger servidores web de Internet.

Se describirá a continuación el procedimiento usado por el intermediario 14 y el agente 16 para establecer una conexión transparente en la capa de transporte usando el TCP entre el cliente 10 y el servidor 12. En esta forma de realización (en la medida en la que es la más típica) el protocolo de la capa de transporte es el Protocolo de Control de Transporte (TCP).

Como es bien conocido por los expertos en la tecnología de redes, una sesión TCP entre un cliente 10 y un servidor 12 se establece mediante el envío, por parte de un cliente, de un paquete de sincronización SYN al servidor 12 y la espera de un paquete de acuse de recibo ACK a cambio desde el servidor 12. Cuando el cliente 10 recibe el

paquete ACK, envía un paquete ACK como respuesta al servidor 12. El procedimiento mediante el cual funciona el intermediario 14 de esta forma de realización se desarrolla capturando paquetes TCP SYN en la capa de IP y reteniéndolos mientras se toman decisiones de encaminamiento y política. Si se encuentra una ruta de agente para la conexión TCP, se ofrece la conexión a ese agente 16, el cual a continuación puede intentar materializar la conexión, generando su propio paquete TCP SYN. El agente 16 obtendrá como respuesta del servidor uno de: TCP SYN ACK, lo cual significa que la conexión se ha establecido; TCP RST ACK, lo cual significa que la conexión se rechaza; o un paquete ICMP, lo cual significa que la red, el anfitrión o el puerto es inaccesible.

Si la conexión tiene éxito, el agente 16 establece una sesión de datos para el tráfico de la conexión a transferir entre el intermediario y el agente, y a continuación el agente informa al intermediario 14 de que acepte la conexión TCP solicitada por el cliente 10. A continuación, el intermediario 14 libera el paquete TCP SYN original, donde el sistema utiliza un proxy transparente convencional para terminar la conexión TCP en el intermediario. Todos los datos del cliente 10 se trasladan al agente 16 a través de la sesión de datos establecida por el agente, donde el agente 16 traslada los datos al otro lado del proxy, hacia el servidor. Un proceso similar funciona en el trayecto inverso.

Si la conexión falla, entonces el agente 16 informa al intermediario 14 con respecto al tipo de fallo. El intermediario consume el paquete TCP SYN original y genera un paquete IP nuevo que se enviará al cliente 10. El paquete nuevo 10 se construye de manera que es una réplica de la condición detectada por el agente 16: es decir, TCP RST ACK, o un paquete ICMP. Por lo tanto, la información correcta referente al fallo de la conexión se devuelve al cliente 10. En el caso de un paquete TCP RST ACK, el paquete se direcciona de manera que al cliente 10 le parece que se ha originado en el anfitrión de destino. En el caso de un paquete ICMP, el paquete se direcciona de manera que al cliente 10 le parece que se ha originado en la dirección de pasarela del intermediario 14 y transporta los primeros 64 bytes del paquete TCP SYN original, según requiere el protocolo ICMP definido en RFC 792. De esta manera, la pasarela de intermediario es completamente transparente para el cliente 10 en el nivel TCP.

Se describirá a continuación el procedimiento usado por el intermediario 14 y el agente 16 para establecer una conexión transparente en la capa de red para el Protocolo de Mensajes de Control de Internet (ICMP) entre el cliente 10 y el servidor 12.

El ICMP es un requisito obligatorio para dispositivos de pasarela. La forma de realización soporta el ICMP consumiendo paquetes ICMP, interpretándolos en la capa de aplicación, y, si fuera necesario, creando un intercambio de mensajes del protocolo de la capa de aplicación desde el intermediario 14 a un agente 16.

Por ejemplo, supóngase que el cliente 10 desea efectuar un *ping* al servidor 12. Lo hace enviando un mensaje de solicitud de eco ICMP, con carga útil, desde el cliente 10 al servidor 12. El mensaje ICMP es consumido por el intermediario 14, donde es analizado sintácticamente para determinar su tipo. En este caso, se identificará como una solicitud PING. Se busca la dirección de destino (el servidor 12) y se toma una decisión de encaminamiento para generar un mensaje PING hacia el agente 16, a través del protocolo de capa de transporte segura. El agente 16, tras la recepción del mensaje PING, completa la solicitud generando un mensaje de solicitud de eco ICMP desde él mismo al servidor 12. Al producirse la recepción de una respuesta de eco ICMP, el agente 16 responde con una respuesta “OK” al PING. En el intermediario 14, la respuesta de la capa de aplicación al mensaje PING determina la respuesta ICMP al mensaje de solicitud ICMP original: respuesta de eco ICMP (servidor 12 al cliente 10); o ICMP inaccesible (intermediario 12 al cliente 10). El intermediario 14 gestiona una solicitud de eco ICMP a la dirección de pasarela, sin la necesidad de un encaminamiento a un agente 16. Otros mensajes ICMP son gestionados de una manera similar, consiguiendo así que la forma de realización sea transparente al ICMP.

A continuación se describirá el procedimiento usado por el intermediario 14 y el agente 16 para establecer una conexión transparente en la capa de red para el Protocolo de Datagrama de Usuario (UDP) entre el cliente 10 y el servidor 12.

El UDP, Protocolo de Datagrama de Usuario, es un protocolo muy sencillo que permite el envío de un paquete de datos, conocido como datagrama, desde un anfitrión a otro. La forma de realización soporta el UDP consumiendo paquetes UDP desfragmentados, en el intermediario 14, y realizando apropiadamente un encaminamiento a los agentes 16.

Supóngase que el cliente 10 envía un datagrama UDP al servidor 12. El intermediario 14 consume el datagrama y lo retransmite al agente 16, lugar desde el cual el datagrama se retransmite al servidor 12, que tiene la dirección del agente 16 como su dirección de origen y un puerto de retransmisión de origen seleccionado por el agente. Las respuestas del datagrama siguen el trayecto inverso desde el servidor 12 al puerto de retransmisión seleccionado por el agente, en el agente 16. El agente 16 recibe la respuesta y transporta el datagrama al intermediario 14 usando el transporte seguro. La respuesta del datagrama se envía al cliente 10 que tiene como su dirección de origen la correspondiente del servidor 12.

Cuando un intermediario identifica el agente al que se debería encaminar un datagrama, solicita el establecimiento de una retransmisión de datagrama. Para establecer una retransmisión de datagrama, en primer lugar un agente establece una sesión de datagrama denominada, entre él mismo y el intermediario de servicio. El agente, si acepta

la solicitud de retransmisión, responde con el nombre de la sesión de datagrama asignada a la retransmisión de datagramas. A continuación, el intermediario envía el datagrama, en forma de un paquete IP completo, a través de la sesión de datagrama denominada, y almacena los detalles de la retransmisión (protocolo = UDP, dirección de origen, puerto de origen, dirección de destino, puerto de destino). El agente envía el datagrama al servidor de destino final usando la retransmisión de datagramas establecida.

Una retransmisión de datagramas establecida soporta respuestas en la dirección de retorno, donde el agente recibe la respuesta y envía el datagrama, como un paquete IP completo, a través de la sesión de datagramas denominada. El intermediario es responsable de la fragmentación de los paquetes y de la transmisión en bruto del datagrama al anfitrión.

La sesión de datagrama entre el intermediario y el agente es un conducto seguro bidireccional. El conducto es fiable. No obstante, los transportes de datagramas deberían ser, por definición, no fiables. Por lo tanto, dentro de la forma de realización, se utilizan estrategias en las sesiones de datagrama para mimetizar la falta de fiabilidad. Sobre la sesión de datagrama se implementa un protocolo de control de flujo para proporcionar una información de extremo-a-extremo sobre el número de bytes pendientes entre las entidades pares de la sesión de datagrama. Si, por algún motivo, se produce una congestión en las sesiones de datagrama, las entidades pares perderán datagramas. Si una sesión de datagrama subyacente se desconecta, se pierden los datagramas pendientes. En el intermediario y el agente se pueden implementar otras estrategias conocidas de gestión de tráfico. Por ejemplo, la forma de realización puede implementar el “envejecimiento” de datagramas: es decir, se pierden datagramas si el intervalo de tiempo que los mismos han estado en cola de espera supera un umbral configurable.

Aunque la forma de realización descrita anteriormente soporta el UDP, el principio de funcionamiento se puede extender fácilmente a cualquier servicio basado en datagramas.

La forma de realización soporta la capacidad de que una pasarela de red, a título de intermediaria, preste servicio a múltiples redes virtuales a través de una red de acceso de cliente. La red de acceso de cliente es responsable de gestionar el acceso de clientes, por ejemplo a través de una combinación de servicios convencionales de AAA (Autorización, Autenticación y Contabilidad), tales como RADIUS, y servicios de VPN, tales como IPSec o MPLS.

Cada red virtual consta de varios clientes 10, agentes 14 y una dirección IP de pasarela virtual. El intermediario restringe el acceso del cliente a dentro de la red virtual del mismo. La identidad del cliente se determina mediante una dirección de origen IP (intervalo de direcciones IP del lado del cliente sin solapamiento). Cada red virtual trabaja sobre un intervalo de direcciones IP del lado del servidor con solapamiento, en la medida en la que la comunicación entre clientes y servidores se sitúa dentro del contexto de la red virtual. Las relaciones de confianza de los clientes con una red virtual se pueden establecer dinámicamente a través de servicios existentes de AAA, usando una asignación de direcciones IP o bien estática o bien dinámica.

A continuación se describirá haciendo referencia a la figura 2 la estructura del intermediario 14.

El intermediario 14 se implementa actualmente en un ordenador servidor que ejecuta el sistema operativo Linux.

Los agentes 16 establecen y mantienen una sesión de control sobre un transporte seguro (TLS sobre TCP, en esta forma de realización) y una serie de sesiones de datos dinámicas. Un módulo de transporte seguro 30 es responsable de proporcionar un transporte seguro, interaccionando con el módulo de TCP 32 del núcleo *kernel* Linux, el cual a su vez recibe y transmite paquetes IP por medio, respectivamente, de “ENTRADA DE IP” y “SALIDA DE IP”.

En un intermediario de servicio, un módulo de control 34 mantiene sesiones de control. Las sesiones de datos son de dos formas, sesiones STREAM y sesiones DGRAM. En un intermediario operativo, un módulo de gestión de STREAM 36 mantiene sesiones STREAM y un módulo de gestión de DGRAM 38 mantiene sesiones DGRAM. Los agentes son responsables del establecimiento de todas las sesiones, permitiéndose por lo tanto que se atraviesen cortafuegos y proxys intermedios. Las sesiones de datos se pueden establecer bajo demanda o pueden estar pre-conectadas.

Todo el tráfico IP pasa a través del núcleo *kernel*; el tráfico de IP hacia dentro pasa a través de un módulo de ENTRADA DE IP 40, donde el mismo se traslada a un módulo de Filtro de IP (PREENCAMINAMIENTO) (PREROUTING)) 42. Usando *iptables*, módulo de filtrado de paquetes del núcleo *kernel* de Linux, se establecen reglas para situar en cola paquetes para el módulo de la capa de aplicación denominado Demultiplexor de Paquetes 44.

El módulo Demultiplexor de Paquetes 44, sobre la base de información mantenida por un módulo de base de datos de intermediario de servicio 46 se usa para determinar cómo procesar cada paquete que pasa a través del intermediario, basándose en el protocolo IP, el cliente de origen, el servidor de destino y en función del protocolo, los puertos de los protocolos de origen y destino.

El módulo Demultiplexor de Paquetes 44 implementa también un filtro de paquetes configurable, de cada de aplicación, (cortafuegos) basado en reglas de la Lista de Control de Acceso (ACL) almacenadas en la base de datos de intermediario de servicio 46. La presente forma de realización soporta ACL definidas por cada cliente y ACL por cada red.

Cuando el intermediario está gestionando paquetes TCP, en la entrada del módulo Demultiplexor de Paquetes 44 se sitúan en cola de espera únicamente paquetes SYN. Si un paquete TCP SYN representa una conexión TCP nueva válida para un cliente válido (según define el contenido de la base de datos de intermediario de servicio 46), al módulo de control 34 se traslada un mensaje SYN, con los detalles del paquete. El módulo de control 34 localiza una sesión válida de control de agente, y ofrece la nueva conexión TCP a través de la sesión de control establecida. Si no se encuentra ninguna ruta de agente, entonces el módulo de control 34 genera un paquete en bruto de ICMP Inaccesible para su transmisión al originador de la solicitud TCP a través de un módulo de salida de IP 48.

Si la respuesta de un agente a una conexión TCP nueva es “OK”, entonces la sesión de STREAM asignada se localiza en la base de datos de intermediario de servicio 46 y se fija para situarse en un estado de “estacionamiento”, guardando los detalles de la conexión TCP. A continuación, se envía un mensaje al módulo Demultiplexor de Paquetes 44 para darle instrucciones de manera que responda al paquete TCP SYN original con un “ACEPTACIÓN”, permitiendo que el paquete TCP SYN llegue al módulo TCP de núcleo *kernel* 32. El módulo de Filtro de IP (PREENCAMINAMIENTO) 42 está configurado para funcionar como un proxy transparente, usando NAT de destino, para un puerto proxy transparente, en el que está escuchando el módulo de gestión de flujos continuos (*stream*) 36. El módulo de gestión de flujos continuos 36 acepta la conexión TCP nueva y le pregunta al módulo de Filtro de IP (PREENCAMINAMIENTO) 42 cuál era la dirección de destino original de la conexión. Basándose en las direcciones y los puertos de origen y de destino, la sesión de STREAM estacionada se localiza y se deja de estar estacionada. El proxy establecido se envía a un proxy de flujos continuos 50 para gestionar hasta que finalice la conexión.

Si la respuesta del agente a una conexión TCP nueva es “RECHAZO” (“REFUSE”), entonces el módulo de control 34 envía un mensaje al módulo Demultiplexor de Paquetes 44 para rechazar la conexión TCP. En este caso, el paquete TCP SYN original se modifica a un paquete TCP RST y el origen y el destino se intercambian para reflejar el paquete hacia el cliente. A continuación, el paquete original se consume (DROP) y el paquete modificado se envía al módulo de SALIDA de IP 48.

Si la respuesta del agente a una conexión TCP nueva es “INACCESIBLE” (“UNREACHABLE”), entonces el módulo de control 34 envía un mensaje al módulo Demultiplexor de Paquetes 44 para responder al TCP SYN con un paquete ICMP INACCESIBLE (ICMP UNREACH). En este caso, el paquete TCP SYN original se modifica a un paquete ICMP INACCESIBLE y el origen y el destino se intercambian para reflejar el paquete hacia el cliente. Los primeros 64 bytes del paquete TCP SYN original se copian en el paquete modificado ICMP INACCESIBLE de acuerdo con la especificación del ICMP. A continuación, el paquete original se consume (DROP) y el paquete modificado se envía al módulo de SALIDA DE IP 48.

A continuación se describirá la gestión del ICMP. Los paquetes ICMP son puestos en cola de espera por el módulo de Filtro de IP (PREENCAMINAMIENTO) 42 para el módulo Demultiplexor de Paquetes 44. En el caso de una Solicitud de Eco ICMP, se envía un mensaje “PING” al módulo de control 34, donde se envía un mensaje “PING”, a través de una sesión de CONTROL establecida, a un agente seleccionado (encaminado), basándose en información de la base de datos de intermediario de servicio 46. La Solicitud de Eco ICMP original se pierde.

En función de la respuesta del agente al PING, se crea un mensaje ICMP nuevo y el mismo se envía al cliente en forma de un paquete IP en bruto a través del módulo de SALIDA de IP 48.

A continuación se describirá la gestión del UDP. Los paquetes UDP son puestos en cola de espera por el módulo de Filtro de IP (PREENCAMINAMIENTO) 42 para el módulo Demultiplexor de Paquetes 44. El módulo de Filtro de IP (PREENCAMINAMIENTO) **44** desfragmenta los paquetes UDP. No obstante, una implementación alternativa podría permitir que la desfragmentación se produjese en el módulo Demultiplexor de Paquetes 44. La información en los encabezamientos IP y UDP del datagrama UDP se usa para seleccionar/buscar una asociación de datagrama (DA) establecida, en la base de datos de intermediario de servicio 46. Si se encuentra una DA, entonces la información de la DA determina que el datagrama o bien debería descartarse, o bien debería reenviarse directamente a una RETRANSMISIÓN DE DGRAM 52 asignada. Si no se encuentra ninguna DA, entonces se crea una DA nueva, y el paquete de datagrama se reenvía al módulo de control 34. El módulo de control 34, usando información de la base de datos de intermediario de servicio 46, determina a qué agente solicitar el establecimiento de una nueva retransmisión de datagrama.

Si el agente responde con “OK”, entonces la sesión de DGRAM asignada se localiza en la base de datos de intermediario de servicio 46 y la DA está vinculada a una RETRANSMISIÓN DE DGRAM 52. A continuación, el paquete original se reenvía a la RETRANSMISIÓN DE DGRAM 52.

Si el agente responde con “RECHAZO”, entonces el paquete de datagrama se descarta y la DA se marca con “descarte”. La condición de “descarte” está gestionada por un temporizador, provocando que la DA se destruya automáticamente después de un periodo de tiempo configurable.

La RETRANSMISIÓN DE DGRAM 52 es responsable de gestionar el tráfico sobre sesiones de DGRAM. Cada RETRANSMISIÓN DE DGRAM 52 implementa el control del flujo, el mantenimiento de DA, y estrategias de gestión de tráfico. Cualquier datagrama de IP se puede transportar sobre una sesión de DGRAM. Los datagramas en el trayecto inverso (agente al cliente) son recibidos por una RETRANSMISIÓN DE DGRAM 52 y transmitidos, después de la fragmentación, por medio del módulo de SALIDA de IP 48.

Se asignan redes virtuales a uno de entre una serie de intermediarios de servicio desplegados. Los agentes pueden usar un servicio de registro para descubrir el intermediario de servicio asignado como pasarela a la red virtual del agente. De esta manera, una forma de realización de la invención se puede escalar para soportar un número elevado de clientes, agentes y redes virtuales. En la forma de realización preferida, el servicio de registro se implementa usando servicios web seguros (usando procedimientos HTTPS).

El agente se puede implementar sobre cualquier plataforma que proporcione una pila de red TCP/IP, por ejemplo, un ordenador que ejecute un sistema operativo Windows (rtm), Linux (rtm), Unix (rtm), Apple (rtm) o una máquina virtual Java (rtm). En la figura 3 se muestra la estructura genérica de un agente. La plataforma debe proporcionar interfaces para TCP/IP. En la figura 3, la interfaz de TCP 60, la interfaz de UDP 62 y la interfaz de ICMP 64 de la pila TCP/IP nativa se muestran por separado por motivos de claridad.

Todo el tráfico entre el agente 16 y el intermediario de servicio 14 se establece como conexiones seguras salientes (del agente al intermediario de servicio) usando el módulo de TRANSPORTE SEGURO 66, el cual proporciona TLS sobre TCP en la forma de realización preferida. Puesto que los agentes son responsables de atravesar proxys intermedios entre el agente 16 y el intermediario de servicio 14, un agente debe incluir una funcionalidad de conexión proxy, la cual se puede incluir en el módulo de TRANSPORTE SEGURO 66, o, como alternativa, se puede incluir en un módulo proxy independiente. Un agente debería implementar una funcionalidad proxy de CONEXIÓN HTTP (HTTP CONNECT) como mínimo.

En un agente 16, un módulo de CONTROL 68 mantiene sesiones de control. Las sesiones de datos presentan dos formas, sesiones de FLUJO CONTINUO, gestionadas por un módulo GESTOR DE FLUJOS CONTINUOS 70, y sesiones de DGRAM, gestionadas por un GESTOR DE DGRAM 72.

El módulo de CONTROL 68 del agente 16 establece y mantiene una única sesión de control segura por medio del módulo de TRANSPORTE SEGURO 66. El módulo CONTROL 68 está a la escucha de mensajes de control provenientes del intermediario 14 sobre la sesión de control. El módulo de CONTROL 68 debería tener la capacidad de gestionar al mismo tiempo múltiples mensajes de intermediarios de servicio.

Si un intermediario de servicio 14 envía un mensaje PING, entonces el módulo de CONTROL 68 iniciará un ping ICMP desde el agente 16 al destino a través de la interfaz de TCP/IP ICMP 60. Un experto en la materia apreciará que un ping ICMP es permitido típicamente solo por procesos con derechos avanzados o de administrador. Como consecuencia, un agente 16 se debe ejecutar típicamente en un contexto con derechos avanzados o de administrador. La respuesta al ICMP se remitirá al intermediario de servicio 14 en un mensaje de respuesta.

Si un intermediario de servicio 14 envía un mensaje CONEXIÓN TCP (“TCP CONNECT”) al agente 16, entonces el módulo de CONTROL 68 solicita una conexión TCP por medio del módulo de gestión de FLUJOS CONTINUOS 70. El módulo de GESTOR DE FLUJOS CONTINUOS 70 debe, en primer lugar, intentar conectarse al destino especificado. Si la conexión tiene éxito, entonces el módulo de GESTOR DE FLUJOS CONTINUOS 70 conectará una sesión de FLUJO CONTINUO al intermediario de servicio 14 y notificará al módulo de CONTROL 68 el éxito de la conexión y la sesión de FLUJO CONTINUO que es responsable de la conexión TCP. A continuación, el módulo de GESTOR DE FLUJOS CONTINUOS 70 crea un PROXY DE FLUJOS CONTINUOS 74 para derivar datos de un lado a otro entre la sesión de FLUJO CONTINUO y la conexión TCP local 78. No obstante, si la conexión TCP falla, entonces el módulo de GESTOR DE FLUJOS CONTINUOS 70 envía una notificación al módulo de CONTROL 68, el cual envía el mensaje apropiado de respuesta de error. Alternativamente, un módulo de GESTOR DE FLUJOS CONTINUOS 70 del agente 16 puede preconectar sesiones de FLUJO CONTINUO al intermediario de servicio 16 y asignar sesiones de FLUJO CONTINUO preconectadas cuando sea necesario, de esta manera se pueden mejorar los tiempos de establecimiento de las conexiones TCP.

Si un intermediario de servicio 14 envía un mensaje CONEXIÓN UDP al agente 16, entonces el módulo de CONTROL 68 solicita una Asociación de Datagrama (DA) UDP por medio del gestor de DGRAM 72. El GESTOR de DGRAM 72 asigna la DA a una RETRANSMISIÓN DE DGRAM 76. Si la RETRANSMISIÓN DE DGRAM 76 no tiene ya una sesión de DGRAM para intermediario de servicio 14, entonces la misma se establece en este momento. Una RETRANSMISIÓN DE DGRAM 76 tiene la capacidad de gestionar múltiples DAs. El GESTOR DE DGRAM 72 envía un mensaje de respuesta al intermediario de servicio 14 identificando la sesión de DGRAM a la cual vincular la DA del intermediario de servicio 14.

La RETRANSMISIÓN DE DGRAM 76 del agente 16 implementa preferentemente una función de temporizador para monitorizar DAs que están activas y DAs no vinculadas que ya no se están usando. Cuando una DA no está vinculada, la RETRANSMISIÓN DE DGRAM 76 del agente 16 envía a la RETRANSMISIÓN DE DGRAM 52 del intermediario de servicio 14 un mensaje al respecto. Después de que la DA se haya desvinculado, cualquier tráfico para la DA desvinculada se descarta.

En la práctica, la mayoría de protocolos UDP son protocolos de estilo orden-respuesta de una sola vez. Teniendo esto en mente, un agente podría implementar inicialmente un tiempo límite de reposo muy corto, por ejemplo 30 segundos, y a continuación, si se intercambian múltiples datagramas, podría prolongar el tiempo límite de reposo. Por ejemplo, después de dos datagramas de respuesta, el agente 16 puede prolongar el tiempo límite de reposo para esa DA a 15 minutos.

En esta forma de realización, tanto los intermediarios 16 como los agentes 14 se identifican y se autentican mutuamente usando certificados digitales X.509. El encaminamiento en el sistema se puede basar en un servicio de red o dirección de destino.

REIVINDICACIONES

1. Sistema de transmisión de datos para un intercambio seguro de datos utilizando un protocolo de control de transmisión entre un cliente (10) y un servidor (12), comprendiendo el sistema un agente (16) y un intermediario (14) conectados a través de por lo menos un cortafuegos para intercambiar datos sobre un enlace de red no asegurada (18), en el que:

el agente (16) establece una sesión de control segura con el intermediario (14) utilizando un transporte seguro sobre el enlace de red insegura (18);

al recibir un paquete TCP SYN desde el cliente (10) a través de una red segura, el intermediario (14) es operativo para capturar el paquete TCP SYN y enviar un paquete de control modificado al agente (16) utilizando la sesión de control segura;

el agente (16) es operativo para recibir el paquete de control modificado desde el intermediario (14) y para generar su propio paquete TCP SYN y para enviar su propio paquete TCP SYN al servidor (12);

al recibir un paquete de respuesta desde el servidor (12), el agente (16) es operativo para enviar un paquete de respuesta al intermediario (14) utilizando la sesión de control segura;

y al recibir un paquete de respuesta desde el agente (16), el intermediario (14) es operativo para enviar un paquete de respuesta al cliente (10);

en el que en el caso de que un intercambio de paquetes de control TCP entre el agente y el servidor indique el establecimiento de una sesión TCP, el agente (16) es operativo para establecer un canal de datos entre el agente y el intermediario con el fin de crear un canal TCP transparente entre el cliente (10) y el servidor (12), y, en el caso de que un intercambio de paquetes de control TCP entre el agente y el servidor indique un fallo en el establecimiento de una sesión TCP, el paquete de respuesta reenviado por el intermediario (14) al cliente (10) indica que la conexión ha fallado.

2. Sistema de transmisión de datos según la reivindicación 1, en el que el cliente (10) está conectado al intermediario (14) para el intercambio de datos a través de una red asegurada.

3. Sistema de transmisión de datos según la reivindicación 2, en el que la red asegurada está conectada a la red no asegurada mediante uno o más de un proxy o dispositivo de traducción de direcciones de red (NAT).

4. Sistema de transmisión de datos según cualquiera de las reivindicaciones anteriores, en el que la sesión de control segura es una conexión del protocolo de seguridad de capa de transporte (TLS) que presenta una conexión realizada a través del puerto TCP 443.

5. Sistema de transmisión de datos según la reivindicación 4, en el que la conexión TLS se realiza a través de un proxy local.

6. Sistema de transmisión de datos según la reivindicación 5, en el que el proxy local utiliza el protocolo HTTP CONNECT.

7. Sistema de transmisión de datos según cualquiera de las reivindicaciones anteriores, en el que el paquete de respuesta es un paquete TCP SYN ACK o un paquete TCP RST ACK.

8. Sistema de transmisión de datos según cualquiera de las reivindicaciones anteriores, en el que el paquete de respuesta son paquetes del protocolo de mensajes de control de Internet.

9. Sistema de transmisión de datos según cualquiera de las reivindicaciones anteriores, operativo para prestar servicio a múltiples redes virtuales a través de una red de acceso de cliente que es responsable de gestionar el acceso de clientes.

10. Sistema de transmisión de datos según la reivindicación 9, en el que la red de acceso de cliente utiliza uno o ambos de los servicios convencionales de AAA (Autorización, Autenticación y Contabilidad) y VPN.

11. Procedimiento de transmisión de datos para un intercambio seguro de datos utilizando un protocolo de control de transmisión entre un cliente (10) y un servidor (12), que utiliza un intermediario (14) y un agente (16) conectados a través de por lo menos un cortafuegos para intercambiar datos sobre un enlace de red no asegurada (18), en el que:

el agente establece una sesión de control segura con el intermediario utilizando un transporte seguro sobre el enlace de red insegura (18);

al recibir un paquete TCP SYN desde el cliente (10), el intermediario (14) captura el paquete TCP SYN y envía un paquete de control modificado al agente (16) utilizando la sesión de control segura;

el agente (16) recibe el paquete de control modificado desde el intermediario (14),

genera su propio paquete TCP SYN y envía su propio TCP SYN al servidor (12);

al recibir un paquete de respuesta desde el servidor (12), el agente (16) envía un paquete de respuesta al intermediario (14) utilizando la sesión de control segura; y

al recibir el paquete de respuesta, el intermediario (14) envía un paquete de respuesta al cliente (10);

en el que, en el caso de que el intercambio de paquetes de control TCP entre el agente y el servidor indique el establecimiento de una sesión TCP, el agente (16) establece un canal de datos entre el agente y el intermediario con el fin de crear un canal TCP transparente entre el cliente (10) y el servidor (12), y, en el caso de que un intercambio de paquetes de control TCP entre el agente y el servidor indique un fallo en el establecimiento de una sesión TCP, el paquete de respuesta reenviado por el intermediario (14) al cliente (10) indica que la conexión ha fallado.

12. Procedimiento de transmisión de datos según la reivindicación 11, en el que el cliente (10) se conecta al intermediario (14) para el intercambio de datos a través de una red asegurada.

13. Procedimiento de transmisión de datos según la reivindicación 12, en el que la red asegurada se conecta a la red no asegurada mediante uno o más de un proxy o un dispositivo de traducción de direcciones de red (NAT).

14. Procedimiento de transmisión de datos según cualquiera de las reivindicaciones 11 a 13, en el que la sesión de control segura es una conexión del protocolo de seguridad de capa de transporte (TLS) que presenta una conexión realizada a través del puerto TCP 443.

15. Procedimiento de transmisión de datos según la reivindicación 14, en el que la conexión TLS se realiza a través de un proxy local.

16. Procedimiento de transmisión de datos según la reivindicación 15, en el que el proxy local utiliza el protocolo HTTP CONNECT.

17. Procedimiento de transmisión de datos según cualquiera de las reivindicaciones 11 a 16, en el que el paquete de respuesta TCP es un paquete TCP SYN ACK o un paquete TCP RST ACK.

18. Procedimiento de transmisión de datos según cualquiera de las reivindicaciones 11 a 17, en el que el paquete de respuesta son paquetes del protocolo de mensajes de control de Internet.

19. Procedimiento de transmisión de datos según cualquiera de las reivindicaciones 11 a 18 operativo para prestar servicio a múltiples redes virtuales a través de una red de acceso de cliente.

20. Procedimiento de transmisión de datos según la reivindicación 19, en el que la red de acceso de cliente es responsable de gestionar el acceso de clientes.

21. Procedimiento de transmisión de datos según la reivindicación 20, en el que la red de acceso de cliente utiliza uno o ambos de los servicios convencionales de AAA (Autorización, Autenticación y Contabilidad) y VPN (red privada virtual).

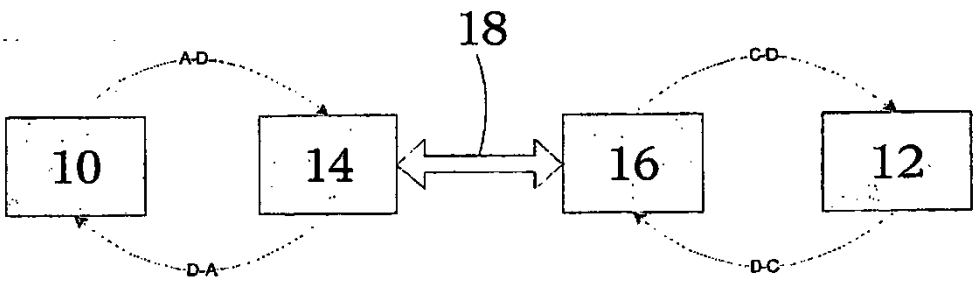

Fig 1

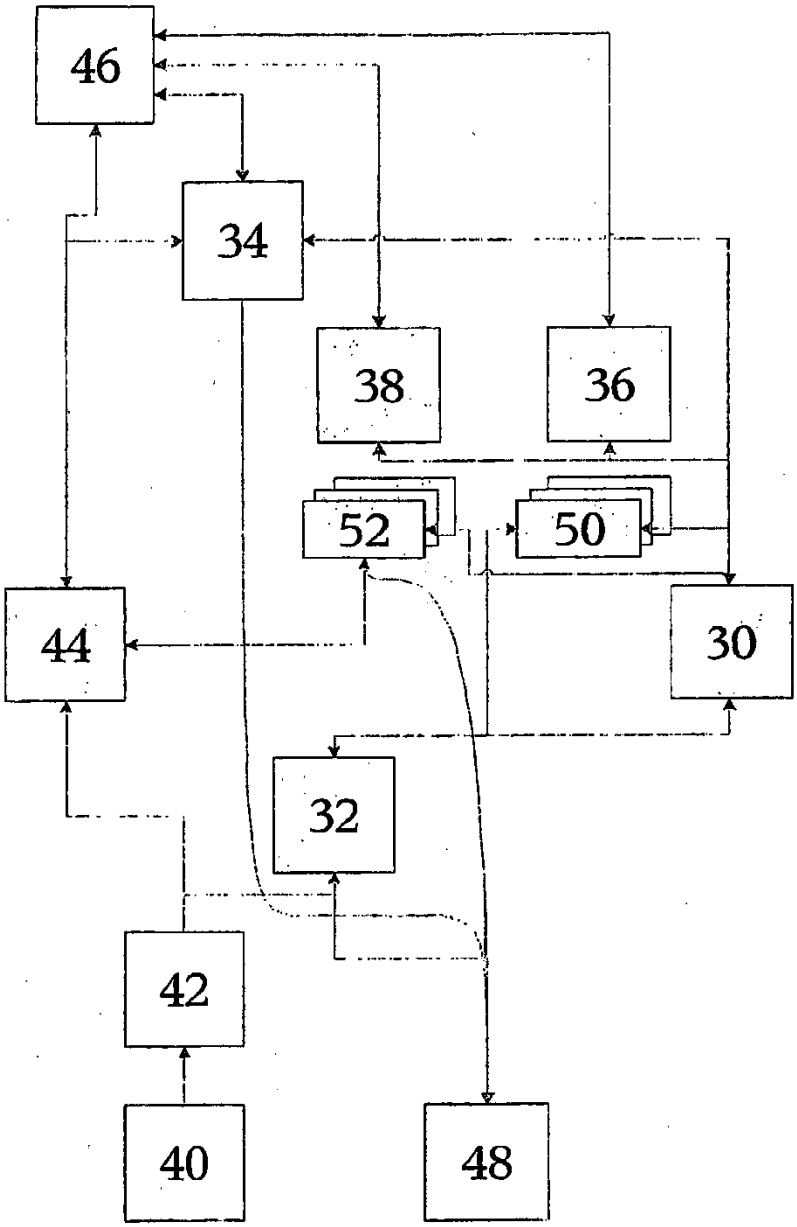

Fig 2

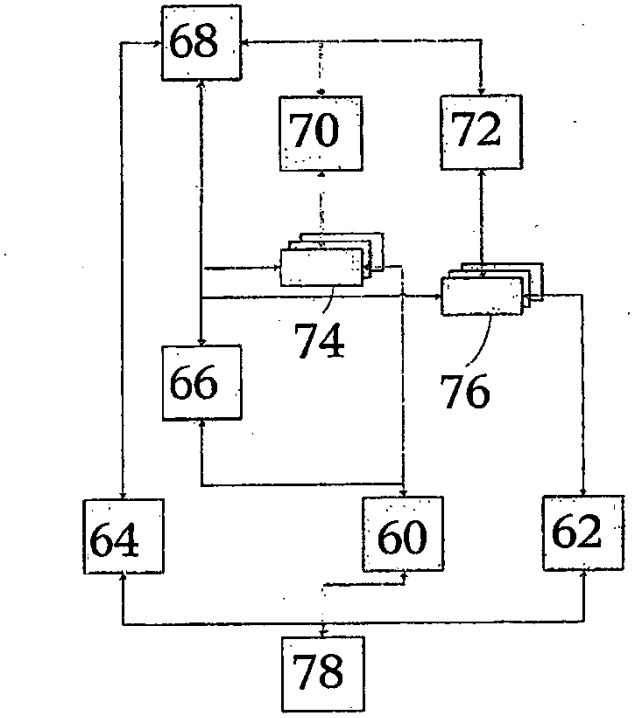

Fig 3