(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 388 784**

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

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Número de solicitud europea: **06018823 .2**
(96) Fecha de presentación: **08.09.2006**
(97) Número de publicación de la solicitud: **1763193**
(97) Fecha de publicación de la solicitud: **14.03.2007**

(54) Título: **Procedimiento de transmisión de un mensaje comprimido**

(30) Prioridad:
12.09.2005 US 222771

(45) Fecha de publicación de la mención BOPI:
18.10.2012

(45) Fecha de la publicación del folleto de la patente:
18.10.2012

(73) Titular/es:
HOB GMBH & CO. KG
SCHWADERMUHLSTRASSE 3
90556 CADOLZBURG, DE

(72) Inventor/es:
Brandstätter, Klaus

(74) Agente/Representante:
Carpintero López, Mario

DESCRIPCIÓN

Procedimiento de transmisión de un mensaje comprimido

Campo de la invención

La presente invención versa acerca de un procedimiento de transmisión de un mensaje mediante la transmisión de datos comprimidos entre un remitente y un destinatario por medio de una red de datos.

Antecedentes de la invención

Antes de pasar a la invención, ha de darse una explicación de los antecedentes de la transmisión de datos comprimidos aplicada a diferentes tipos de protocolos de transmisión y a la eficacia de la compresión de datos. Considerando que la porción de los datos no comprimidos que puede usarse por el algoritmo de compresión de datos ejerce una influencia fundamental en la eficacia de un procedimiento de compresión para analizar los datos y su estructura, cuantos más datos pueda analizar el algoritmo de compresión de datos, mayor será la eficacia del procedimiento de compresión mientras los datos tengan una estructura más o menos homogénea. Por ejemplo, si hay que comprimir un documento de texto, el resultado de la compresión será óptimo si el algoritmo de compresión puede analizar cierta cantidad de texto antes de empezar a comprimir. Si el algoritmo tuviera que gestionar pequeñas porciones de texto independientemente entre sí, el resultado sería, desde luego, peor.

Según parece, un algoritmo que no gestione una cantidad de datos en su conjunto, sino que comprima los datos bloque a bloque, cada uno independientemente de los demás, en la parte remitente de una transmisión de datos y que descomprima los datos al revés en el lado destinatario no alcanzará un máximo de eficacia.

Un algoritmo que, por una parte, divida los datos en bloques menores pero que, por otra parte, almacene información sobre cada bloque comprimido y, por ende, sobre las últimas partes del documento ya comprimidas, optimiza la compresión de bloque respectivo siguiente y, por ende, funcionará con un rendimiento mayor. En este caso, aumenta la información sobre los datos con cada bloque comprimido y también aumentará la eficacia de la compresión si los datos tienen una estructura apropiada. Dado que, como se ha mencionado en lo que antecede, la compresión depende de la información que ya se ha reunido sobre el documento, el destinatario también tiene que analizar los datos entrantes y reunir información de la misma forma que lo hace el remitente para poder descomprimir los datos. Como consecuencia del análisis de los datos ya comprimidos o descomprimidos, tanto el remitente como el destinatario tienen exactamente la misma información sobre los datos en cuestión cuando se procesa un dato en particular. Eso quiere decir en particular que las partes individuales del documento tienen que pasar por el algoritmo de compresión exactamente en el mismo orden que pasan por el algoritmo de descompresión.

Básicamente, ahora la transmisión de datos entre un remitente y un destinatario por medio de una red de datos está organizada por un protocolo que define las reglas de establecimiento de una conexión entre el remitente y el destinatario y para el intercambio de datos entre los mismos. Aquí se hace una distinción en informática entre los que se denominan protocolos sin conexión y orientados a conexiones para el enlace en red. En el caso de un protocolo sin conexión, el remitente y el destinatario no intercambian ninguna información que no sean los datos propiamente dichos. Si se envía más de un paquete, cada paquete es transmitido de forma absolutamente independiente de cualquiera de los demás y el remitente no recibe confirmación de si el paquete ha sido recibido o no. Así, no es seguro si el primer paquete enviado será el primer paquete recibido; es decir, el orden en el que los paquetes son recibidos no es necesariamente igual al orden en el que fueron enviados. Ni siquiera se garantiza que un paquete haya alcanzado su destino en absoluto.

En el caso de un protocolo orientado a conexiones, los paquetes de datos contienen información adicional que permite que los socios de la comunicación averigüen, cuando una conexión se inicia o termina, en qué orden fueron enviados los paquetes individuales y si falta un paquete. Además, normalmente hay una definición en el sentido de que el remitente aguarda una respuesta de confirmación de la recepción por parte del destinatario y de qué hacer si falta un paquete; por ejemplo, volver a enviar el paquete.

Es posible usar un protocolo sin conexión como protocolo base para un protocolo orientado a conexiones. Por ejemplo, con el protocolo TCP/IP, se usa el protocolo sin conexión IP para enviar paquetes TCP orientados a conexiones, es decir, en TCP, se intercambian paquetes IP que contienen información sobre el inicio y el fin de una conexión y sobre el orden de los paquetes.

A partir de lo anterior, resulta evidente que el tráfico de datos con protocolos sin conexión no puede ser comprimido tan eficazmente como los datos enviados con un protocolo orientado a conexiones. La razón es que, según las reglas del protocolo de transmisión usado, los datos se separan en paquetes individuales antes de ser enviados por separado, y uno tras otro, al destinatario. En el caso de un protocolo sin conexiones, solo se puede comprimir cada paquete por sí solo, independiente de cualquier otro, porque no se sabe cuándo comienza ni termina la transmisión de una cantidad conectada de datos ni en qué orden llegan los paquetes al extremo receptor, considerando que la compresión no puede realizarse tan eficazmente como si se usase un protocolo orientado a conexiones, en el que los paquetes son enviados en un orden específico y toda la transmisión puede ser comprimida en contexto.

Al contrario que un protocolo sin conexiones —tal como se ha afirmado en lo que precede—, cuando se usa un protocolo orientado a conexiones sería posible comprimir todo el conjunto de datos enviado durante una conexión como si fuera un documento homogéneo. Sin embargo, el problema está en que no se sabe cuándo acaba una transmisión individual ni cuándo el remitente aguarda que responda el destinatario. En este caso, toda la conexión moriría si no se enviasen los datos, porque el remitente sigue aguardando más datos que comprimir. Así, normalmente solo se comprime aquella parte de los datos que el programa remitente pone en un bloque. Para distinguir estos bloques de los paquetes de datos creados por el protocolo de transmisión se los denomina registros. Sin embargo, en vez de comprimir cada registro por separado de los demás, el algoritmo de compresión podría reunir información sobre cada registro que use para optimizar la compresión para el registro siguiente, tal como se describe en lo que precede, según se describe en el documento US 2005/0086383. Esto se denomina “compresión orientada a registros”. La acción de emplear las memorias tampón usadas por el algoritmo de compresión y de transmitir los datos comprimidos al protocolo de la red es denominada “volcado”. Con una compresión orientada a registros, tal volcado se efectúa después de cada registro.

Aunque la compresión orientada a registros no es problema usando un protocolo orientado a conexiones, no puede ser usada con protocolos sin conexión, dado que ahí, tal como se describe en lo que antecede, los paquetes individuales de datos son transmitidos independientemente sin ocuparse de su orden. Pero ni siquiera con los protocolos orientados a conexiones la compresión orientada a registros es necesariamente la técnica más eficaz que puede usarse, como puede verse en lo siguiente.

Resumen de la invención

Un objeto común de la invención es proporcionar procedimientos de transmisión de un mensaje mediante la transmisión de datos comprimidos entre un remitente y un destinatario, procedimientos de transmisión que están mejorados en lo que se refiere a la eficacia de la compresión de los datos.

El principal aspecto de la invención bajo este objeto es proporcionar un procedimiento de transmisión de un mensaje mediante la transmisión de datos comprimidos entre un remitente y un destinatario que permita una compresión más eficiente orientada a registros con protocolos sin conexión.

Se toma en consideración este aspecto mediante un procedimiento de transmisión de un mensaje mediante la transmisión de datos comprimidos entre un remitente y un destinatario por medio de una red de datos que comprende las etapas de:

– a) proporcionar una pasarela del lado del remitente asociada con dicho remitente y una pasarela del lado del destinatario asociada con dicho destinatario,
– b) el remitente fracciona el mensaje en registros de datos y envía tales registros de datos a la pasarela del lado del remitente con un protocolo orientado a conexiones,
– c) dicha pasarela del lado del remitente, en una compresión orientada a registros, comprime sucesivamente dichos registros de datos en registros de datos comprimidos por medio de un procedimiento de compresión que reúne y analiza dichos registros de datos y optimiza la compresión de un registro de datos a otro,
– d) dicha pasarela del lado del remitente transforma los registros de datos comprimidos en un protocolo sin conexión,
– e) dicha pasarela del lado del remitente transmite dichos registros de datos comprimidos en paquetes a la pasarela del lado del destinatario por medio de dicha red de datos con el protocolo sin conexión,
– f) dicha pasarela del lado del destinatario vuelve a transformar dichos registros de datos comprimidos al protocolo orientado a conexiones tras volver a establecer el orden de los paquetes,
– g) dicha pasarela del lado del destinatario descomprime dichos registros de datos comprimidos de dichos paquetes en dichos registros de datos, y
– h) dicha pasarela del lado del destinatario envía dichos registros de datos al destinatario con dicho protocolo orientado a conexiones.

La pasarela del lado del remitente garantiza que se realice la debida gestión de los datos del mensaje conociendo el o los protocolos de transmisión usados. Tienen que conocerse las reglas y las estructuras del o de los protocolos para poder usar un procedimiento de compresión idealmente adecuado para las necesidades del protocolo particular. En principio, ambas pasarelas tienen funciones acordes; concretamente, analizan el flujo de datos entre remitente y destinatario, comprimen los datos salientes y descomprimen los entrantes. Si hace falta, podrían añadir al flujo de datos algunos datos para el uso de la pasarela socia. Sin embargo, en la práctica, las dos pasarelas son programas diferentes, dado que muchos protocolos requieren un comportamiento diferente por parte del lado del remitente o del destinatario, respectivamente.

Resumiendo, cuando se vale de un protocolo sin conexiones, la pasarela que comprime los datos que han de ser transmitidos reordena los paquetes de red y, así, puede usar una compresión que está mejorada por el conocimiento sobre el “historial de datos”, es decir, la estructura de datos en paquetes de red previos.

Ahora, según la primera realización de la invención en la etapa b) anteriormente mencionada, los paquetes de datos enviados por el remitente con un protocolo orientado a conexiones son enriquecidos con información de control por

el protocolo orientado a conexiones, lo que garantiza el orden y la integridad de los datos. Ahora bien, para usar una compresión orientada a conexiones con un protocolo orientado a la seguridad, como IPSec, en el que los paquetes son cifrados en aras de la seguridad, los procesos de compresión y descompresión de los datos tienen que estar conectados con los procesos de transformar los datos desde el protocolo orientado a conexiones al de sin conexiones y viceversa. Esto se realiza, por una parte, mediante las anteriores etapas c) y d) del procedimiento en lo referente al lado del remitente y, por otra parte, mediante las etapas f) y g) del procedimiento en lo referente al lado del destinatario.

Según una realización preferente, se introduce en el procedimiento una ventana de datos, que tiene un tamaño máximo predefinido de datos, para el proceso de compresión de la etapa c) en un ciclo. Esto resuelve el problema de que un protocolo orientado a conexiones, como TCP/IP, envíe acuses para confirmar que en el otro extremo de la conexión se han recibido los datos. En este protocolo, el remitente solo envía cierta cantidad de datos de este tamaño de ventana, el número exacto de cuales depende de la implementación y la configuración, y puede cambiar incluso durante una conexión. El remitente deja entonces de enviar hasta que una cantidad suficiente de los datos enviados haya sido objeto de acuse antes de que envíe los siguientes paquetes de datos sin superar el tamaño de la ventana. De hecho, para cada bloque de datos, el remitente aguarda cierta cantidad de tiempo y, si no se recibió ningún acuse de ese bloque durante ese tiempo, se vuelve a enviar el bloque. Si el procedimiento de compresión quisiese intentar reunir más datos que el tamaño de la vente y, en consecuencia, los datos comprimidos no fuesen presentados al lado del destinatario, el remitente dejaría de enviar datos y esperaría los acuses, que nunca ocurrirían. En ese caso, la comunicación, sencillamente, moriría, porque la pasarela del lado del remitente ignoraría los bloques enviados de nuevo, porque ya los había recibido, y el remitente no enviaría ningún bloque nuevo hasta que recibiera los acuses. Por ello, la pasarela y el algoritmo de compresión tienen que ocuparse de que se envíe al extremo receptor una cantidad suficiente de datos, para que se creen suficientes acuses para hacer que el remitente envíe más datos. Esto puede incluir cambiar el tamaño de la ventana de TCP.

Breve descripción de los dibujos

La Fig. 1 es un diagrama de bloques de una red de datos entre un remitente y un destinatario que representa una compresión orientada a registros con protocolos sin conexión según la presente invención, y

la Fig. 2 es un diagrama de bloques de una red de datos entre un cliente de correo electrónico y un servidor de correo electrónico usando una compresión orientada a ficheros para las transmisiones de correo electrónico que no se encuentran bajo la presente invención.

Descripción de las realizaciones preferentes

La Fig. 1 representa una conexión 1 de Internet como la red de datos entre un cliente 2 como remitente y un servidor 3 como destinatario. En el extremo remitente se asocia con dicho cliente una pasarela 4 del lado del cliente que representa una pasarela del lado del remitente. Igualmente, en el extremo receptor se asocia con el servidor 3 una pasarela 5 del lado del servidor que representa una pasarela del lado del destinatario. Ambas pasarelas 4, 5 son programas de soporte lógico que se ejecutan en el cliente 2 y el servidor 3, respectivamente. Dichas pasarelas 4, 5 podrían estar separadas del cliente 2 o el servidor 3 en lo que se refiere a organización y técnica de programa, pero también podrían ejecutarse en un solo ordenador bajo un solo sistema operativo.

El cliente 2 envía un mensaje con el protocolo TCP/IP fraccionando el mensaje en registros de datos y enviando dichos registros de datos a la pasarela 4 del lado del cliente con el protocolo TCP orientado a conexiones enumerado 6. La pasarela 4 del lado del cliente, según su función, analiza el flujo de datos entrante desde el cliente 2 y comprime sucesivamente los registros de datos entrantes en registros de datos comprimidos en una compresión orientada a registros. Esto significa que, en el procedimiento de compresión, se reúnen y analizan los registros de datos del cliente 2 y, en consecuencia, el procedimiento de compresión es optimizado de un registro de datos a otro.

La pasarela 4 del lado del cliente transforma estos registros de datos comprimidos en un protocolo sin conexión, como IPSec, tal como se representa con el número 7 en la Fig. 1, y transmite dichos registros de datos comprimidos en paquetes 8 a través de la conexión 1 de Internet con el protocolo IPSec a la pasarela 5 del lado del remitente en el extremo de recepción. Volcar las memorias tampón de la pasarela 4 del lado del cliente después de comprimir cada paquete se corresponde con el envío de un registro de datos comprimidos. En el extremo receptor, la pasarela 5 del lado del servidor vuelve a transformar dichos registros de datos comprimidos al protocolo TCP orientado a conexiones, restableciendo el orden de los paquetes 8 por la información correspondiente incluida en los registros de datos, debido a la emisión de los registros de datos por el cliente 2 con el protocolo TCP orientado a conexiones en 6. A continuación, la pasarela 5 del lado del servidor descomprime dichos registros de datos comprimidos reordenados y envía los así generados registros de datos, que representan el mensaje que ha de transmitirse, al servidor 3 con el protocolo TCP orientado a conexiones, tal como se representa con el número 9 en la Fig. 1.

El protocolo TCP orientado a conexiones usado como base de la transmisión de datos descrita funciona con acuses de confirmación de que los datos del mensaje que ha de transmitirse entre el cliente y el servidor 3 han sido recibidos en el extremo receptor de la conexión. Para evitar un conflicto entre el procedimiento de acuse y el procedimiento de compresión de datos, este reúne y comprime en un ciclo registros de datos de un tamaño total de

datos, que es un tamaño máximo predefinido de datos de una ventana de datos. Este denominado tamaño de ventana depende de la implementación y la configuración de la transmisión de datos. Al restringir a un ciclo del procedimiento de compresión a este tamaño de ventana y transmitir registros de datos dentro de los límites de esta ventana de datos, se garantiza que, en cualquier momento, el servidor 3 devuelva en una transmisión un acuse de recibo de los registros de datos a través de la pasarela 5 del lado del servidor a una pasarela 4 del lado del cliente y al cliente 2, el cual, tras recibir dicho acuse, sigue transmitiendo registros de datos en el siguiente ciclo de transmisión de datos a la pasarela 4 del lado del cliente para su compresión y transmisión nuevamente a la pasarela 5 del lado del servidor y al servidor 3. Así, la conexión de datos entre el cliente 2 y el servidor 3 no puede morir, lo que fue esbozado como un grave problema en la introducción de la memoria.

En este sentido, puede resultar ventajosa una gestión especial del tamaño de la ventana. La pasarela 4, 5 que realice la compresión podría alterar el tamaño de la ventana, o esa pasarela 4, 5 podría permitir un volcado para garantizar que permanece dentro de los límites del tamaño de la ventana.

Normalmente, el tamaño de la ventana se negocia entre el cliente y el servidor, que son los dos socios de la sesión en ambos extremos de una comunicación. Según la invención, ahora la pasarela 4, 5 respectiva que proporciona la compresión o bien conoce este tamaño de ventana monitorizando las referidas “negociaciones” o altera activamente el tamaño de la ventana enviando los necesarios parámetros en paquetes de red a los socios de la sesión, es decir, el cliente 2 y el servidor 3.

En conexión con la Fig. 1, se explicó una realización de una compresión, una transmisión y una descompresión de datos con un protocolo TCP/IP usando IPSec. Sin embargo, puede usarse la técnica correspondiente de compresión y transmisión en otros entornos y protocolos, como el Web-SecureProxy con SSL del solicitante. Ejemplos adicionales son los protocolos 3270 o TN 3270, que se usan para una comunicación de clientes con ordenadores centrales. Todos estos protocolos —como en el TN 5250— terminan en FF EF. Ahora, con el procedimiento de la invención, el flujo de datos en el lado del remitente puede ser analizado por la pasarela del lado del remitente y, si se encuentra esta secuencia en el flujo de datos (y no comienza ningún bloque nuevo de registros), es necesario y se realiza un volcado con este procedimiento de compresión y transmisión orientado a registros. Si no, la pasarela remitente puede seguir reuniendo datos para obtener más información sobre los datos y alcanzar así una mejor relación de compresión.

Con los protocolos 3270 también es posible la transferencia de ficheros de campos estructurados, por ejemplo para la transmisión de imágenes terminales, transmitiendo mayores bloques de datos en el flujo de datos. Cuando el cliente y el servidor escogen tal transferencia de datos, todo el contenido del fichero transmitido puede ser comprimido por completo usando una compresión orientada a ficheros. En consecuencia, no es necesario ningún volcado de los registros de datos. La pasarela del lado del cliente o la pasarela del lado del servidor que proporcione la compresión tiene que monitorizar el flujo de datos para encontrar el “desencadenante” para una transferencia de fichero de campos estructurados, que es, para la transmisión servidor-cliente, WCC = 0XF3 o 0X11. Para una transmisión cliente-servidor, el código de operación es 0X88. El código de operación para la siguiente transferencia de fichero es 0XD0.

Con referencia a la Fig. 2, se explica un procedimiento de transmisión de un mensaje mediante la transmisión de datos comprimidos entre un remitente y un destinatario usando una compresión orientada a ficheros que no se encuentra bajo la presente invención y que es ejemplificado por la gestión de transmisiones de correo electrónico con los protocolos SMTP, POP3 e IMAP4. Nuevamente, tenemos una conexión 1 de Internet entre un cliente 2’ de correo electrónico y un servidor 3 de correo electrónico, por ejemplo un servidor 3’ de Microsoft® MS Exchange. Además, el cliente 2’ de correo electrónico está dotado de una pasarela 4’ del lado del cliente; de forma similar, el servidor 3’ de correo electrónico está dotado de una pasarela 5’ del lado del servidor. El cliente 2’ de correo electrónico envía y recibe correo electrónico dirigido al servidor 5’ de correo electrónico, y procedente del mismo, usando el protocolo SMTP para el envío de los correos y el POP3 o el IMAP4 para recibir correos procedentes del servidor 5’ de correo electrónico. En estos últimos casos, el servidor 5’ de correo electrónico es el extremo remitente y el cliente 2’ es el extremo destinatario de la conexión.

Básicamente, el cliente 2’ de correo electrónico inicia una conexión enumerada 10 en la Fig. 2 con la pasarela 4’ del lado del cliente que indica qué tipos de protocolos se usan para el envío y la recepción de correos electrónicos desde el servidor 3’ de correo, es decir, según la Fig. 2, SMTP para el envío y POP3 o IMAP4 para la recepción. La pasarela 4’ del lado del cliente analiza el mensaje de correo electrónico enviado por el cliente 2’ de correo electrónico en busca de partes compresibles del mensaje, como textos, y de componentes incompresibles del mensaje. En este sentido, “incompresible” debería abarcar partes del mensaje que no sean compresibles eficientemente, como datos gráficos ya comprimidos, información corta, como cabeceras de mensajes o registros de datos que tiene sentido que sean procesados por una compresión orientada a registros. En consecuencia, la pasarela 4’ del lado del cliente analiza los mensajes en busca de estas partes compresibles del mensaje y de componentes no compresibles del mensaje.

A continuación, la pasarela 4’ del lado del cliente establece una conexión 11 que informa a la pasarela 5’ del lado del servidor cuáles de los referidos protocolos se usan. Esto se realiza enviando secuencias, tal como se ejemplifica en lo que sigue:

“HOB RD-VPN SMTP V1.1” para el protocolo SMTP de envío
“HOB RD-VPN POP3 V1.1” o
“HOB RD-VPN IMAP4 V1.1”
para usar POP3 o IMAP4 como protocolo para recibir correos electrónicos procedentes del servidor 3’ de correo electrónico.

En consecuencia, la pasarela 5’ del lado del servidor inicia una conexión 12 con el servidor 3’ de correo electrónico. Al principio, se intercambian la información sobre protocolos que van a usarse e información adicional, como datos del remitente y el destinatario del correo, a través de la pasarela 4’ del lado del cliente y la pasarela 5’ del lado del servidor. Esta información es transmitida, tal cual, o sea, sin comprimir, por la pasarela 4’ del lado del cliente y la pasarela 5’ del lado del servidor como pasarelas del lado del remitente y del lado del destinatario, respectivamente. Como mucho, puede aplicarse una compresión orientada a registros, tal como se ha explicado en conexión con la Fig. 1, a las partes introductorias del mensaje, por ejemplo en lo referente a la información de dirección o nombre.

Cuando ha de enviarse el mensaje propiamente dicho, por ejemplo la parte de texto del correo electrónico, la pasarela 4’ del lado del cliente comprime todos los datos de texto y los envía con el protocolo SMTP orientado a conexiones a la pasarela 5’ del lado del servidor como un fichero de datos comprimidos.

La pasarela 5’ del lado del servidor descomprime este fichero de datos comprimidos y combina la parte descomprimida del mensaje con los componentes no comprimidos del mensaje formando el mensaje completo, el cual, a su vez, es enviado al servidor 3’ de correo electrónico para su gestión ulterior.

Aunque la anterior transmisión del mensaje se refiere a un correo electrónico enviado desde el cliente 2’ de correo electrónico al servidor 3’ de correo electrónico, tal como se esquematiza con la doble flecha 13 en la Fig. 2, las transmisiones de correo electrónico desde el servidor 3’ de correo electrónico hasta el cliente 2’ de correo electrónico (véase la doble flecha 14 en la Fig. 2) son gestionadas con el protocolo POP3 o el IMAP4 a través de las conexiones 10, 11, 12.

Tal como puede verse por lo que precede, debido a la compresión de la transmisión de los datos, deben transferirse menos datos a través de la conexión 1 de Internet entre el cliente 2’ de correo electrónico y el servidor 3’ de correo electrónico.

Se da consideración especial al hecho de que, en general, los datos de correo electrónico están cifrados con el denominado cifrado Base64, lo que da origen a una posibilidad de compresión adicional, que puede ser ejemplificada como sigue:

Un correo electrónico de prueba tiene el contenido siguiente:

Hallo Test
---MIME---
AAAABBBB

En el sistema hexadecimal, este correo electrónico está representado por el siguiente código hexadecimal:

48 61 6C 6C 6F 20 54 65 73 74 0D 0A 2D 2D 2D 4D 49 4D 45 2D 2D 2D 0D 0A 41 41 41 41 42 42 42 42 0D 0A 2E 0D 0A

Este código hexadecimal es comprimido preliminarmente por el procedimiento siguiente:

La porción “Hallo Test” es texto regular, que no está cifrada en Base64, lo que está marcado por medio de una secuencia de escape:

F1 18.

Después de eso, sigue el código hexadecimal para el texto “Hallo Test” de forma no enmendada:

48 61 6C 6C 6F 20 54 65 73 74 0D 0A 2D 2D 2D 4D 49 4D 45 2D 2D 2D 0D 0A

Sigue la porción MIME del correo electrónico, cifrada en Base 64, que es introducida por la siguiente secuencia de escape:

F2 06.

Los siguientes datos del correo electrónico son decodificados quitando el cifrado Base64, es decir,

41 41 41 41 = 000000 000000 000000 000000

se transfiere a

0000 0000 0000 0000 0000 0000 = 00 00 00

La secuencia

42 42 42 42 = 000001 000001 000001 000001

se transfiere a

0000 0100 0001 0000 0100 0001 = 04 10 41

Resumiendo, la secuencia

41 41 41 41 42 42 42 42

es reducida a

00 00 00 04 10 41

Por último, la secuencia de final de correo electrónico es sustituida por una secuencia especial F0.

Así, la secuencia final de código hexadecimal precomprimida se lee:

F1 18 48 61 6C 6C 6F 20 54 65 73 74 0D 0A 2D 2D 2D 4D 49 4D 45 2D 2D 2D 0D 0A F2 06 00 00 00 04 10 41 F0

Con el cifrado Base64 aumenta el volumen de los datos, dado que el cifrado da como resultado 4 bytes de datos de transmisión por cada 3 bytes de datos netos. Por ello, la pasarela 4' o 5' del lado del remitente suprime el cifrado Base64 antes de que se efectúe la compresión propiamente dicha. Por supuesto, la pasarela 5' o 4' del lado del destinatario tiene que volver a realizar entonces un cifrado Base64 de los datos descomprimidos.

Generalizando el anterior ejemplo de eliminación del cifrado Base64, la pasarela 4' del lado del remitente permite una compresión múltiple al analizar los datos que han de ser comprimidos, como el correo electrónico, en lo que respecta a una sobrecarga eliminable de datos, como la porción de cifrado en Base64, con base en el conocimiento del protocolo de correo electrónico que realmente describe dicho cifrado en Base64 de la porción MIME, eliminando esta sobrecarga de datos como una compresión previa y comprimiendo adicionalmente los datos restantes en las correspondientes etapas de compresión.

Para decir a la pasarela 4' o 5' del lado del destinatario cuándo empieza y termina la parte comprimida de la transmisión, y si debe efectuarse el cifrado en Base64, las pasarelas 4', 5' usan el protocolo siguiente, tal como ya se esbozó en lo que antecede: Los datos se separan en bloques, que, por razones específicas del protocolo, deberían tener un tamaño, si es posible, de 16383 (= 2^{14}-1) bytes. Estos bloques contienen la información siguiente para controlar la compresión:

Un byte 0xF1 seguido por un número entero que indica una longitud inicia un bloque comprimido de la longitud especificada. Un byte 0xF2 seguido por un número entero que indica una longitud inicia un bloque comprimido de la longitud especificada que tiene que ser cifrado en Base64. Un solo byte 0xF0 sustituye las marcas habituales de fin de mensaje del protocolo usado (por ejemplo, 0x0D 0x0A 0x2E 0x0D 0x0A para SMTP y POP3), identificando, por lo tanto, el fin de un mensaje comprimido.

REIVINDICACIONES

1. Un procedimiento de transmisión de un mensaje mediante la transmisión de datos comprimidos entre un remitente (2) y un destinatario (3, 3') por medio de una red (1) de datos que comprende las etapas de:

– a) proporcionar una pasarela (4) del lado del remitente asociada con dicho remitente (2) y una pasarela (5) del lado del destinatario asociada con dicho destinatario (3),
– b) el remitente (2) fracciona el mensaje en registros de datos y envía tales registros de datos a la pasarela (4) del lado del remitente con un protocolo orientado a conexiones,
– c) dicha pasarela (4) del lado del remitente, en una compresión orientada a registros, comprime sucesivamente dichos registros de datos en registros de datos comprimidos por medio de un procedimiento de compresión que reúne y analiza dichos registros de datos y optimiza la compresión de un registro de datos a otro,
– d) dicha pasarela (4) del lado del remitente transforma los registros de datos comprimidos en un protocolo sin conexión,
– e) dicha pasarela (4) del lado del remitente transmite dichos registros de datos comprimidos en paquetes (8) a la pasarela (5) del lado del destinatario por medio de dicha red (1) de datos con el protocolo sin conexión,
– f) dicha pasarela (5) del lado del destinatario vuelve a transformar dichos registros de datos comprimidos al protocolo orientado a conexiones tras volver a establecer el orden de los paquetes (8),
– g) dicha pasarela (5) del lado del destinatario descomprime dichos registros de datos comprimidos de dichos paquetes (8) en dichos registros de datos, y
– h) dicha pasarela (5) del lado del destinatario envía dichos registros de datos al destinatario (3) con dicho protocolo orientado a conexiones.

2. El procedimiento según la reivindicación 1 en el que:

– en dicha etapa c), el procedimiento de compresión en un ciclo reúne y comprime registros de datos de un tamaño total de datos, que es un tamaño máximo predefinido de datos de una ventana de datos,
– en dicha etapa e), el remitente (2) en un ciclo de transmisión de datos transmite registros de datos dentro de dicha ventana de datos, después de lo cual el remitente (2) espera el acuse de la recepción de los registros de datos por parte del destinatario (3),
– después de dicha etapa h), dicho destinatario (3) vuelve a transmitir al remitente (2) un acuse de recibo de los registros de datos a través de la pasarela (5) del lado del destinatario y de la pasarela (4) del lado del remitente, y
– el remitente (2), tras recibir dicho acuse de recibo, transmite registros de datos en el siguiente ciclo de transmisión de datos a la pasarela (4) del lado del remitente.

3. El procedimiento según la reivindicación 1 en el que el tamaño de datos de la ventana de datos es cambiado entre los ciclos de transmisión de datos.

4. El procedimiento según la reivindicación 1 en el que el protocolo orientado a conexiones es un protocolo TCP, con o sin SSL.

5. El procedimiento según la reivindicación 1 en el que el protocolo sin conexión es uno de entre los protocolos IP e IPSec.

6. El procedimiento según la reivindicación 1 en el que la pasarela (4, 4') del lado del remitente permite una compresión múltiple analizando datos que han de ser comprimidos en lo que respecta a una sobrecarga eliminable de datos debida a características especificadas de protocolo, a la eliminación de esta sobrecarga de datos y comprimiendo adicionalmente los datos restantes.

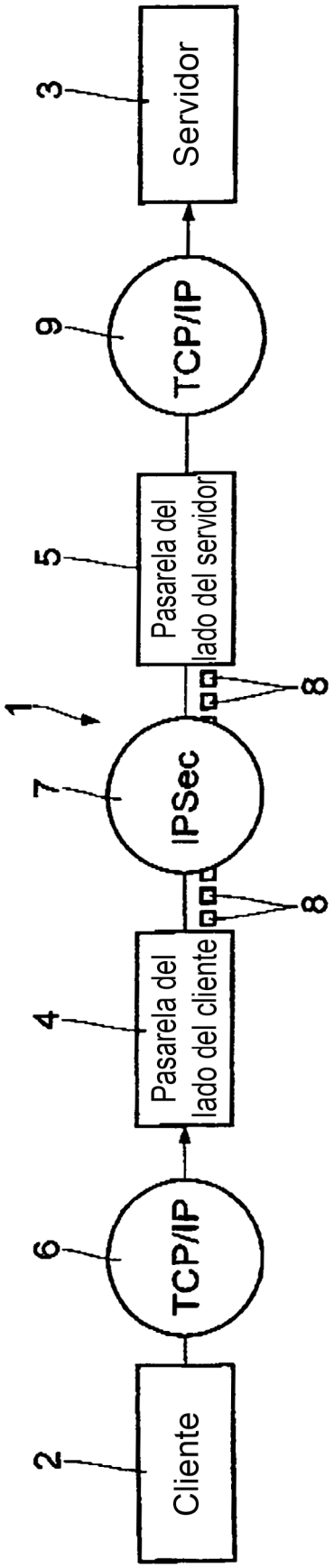
Cliente
2
TCP/IP
6
Pasarela del lado del cliente
4
IPSec
7
1
8
8
Pasarela del lado del servidor
5
TCP/IP
9
Servidor
3

Fig. 1

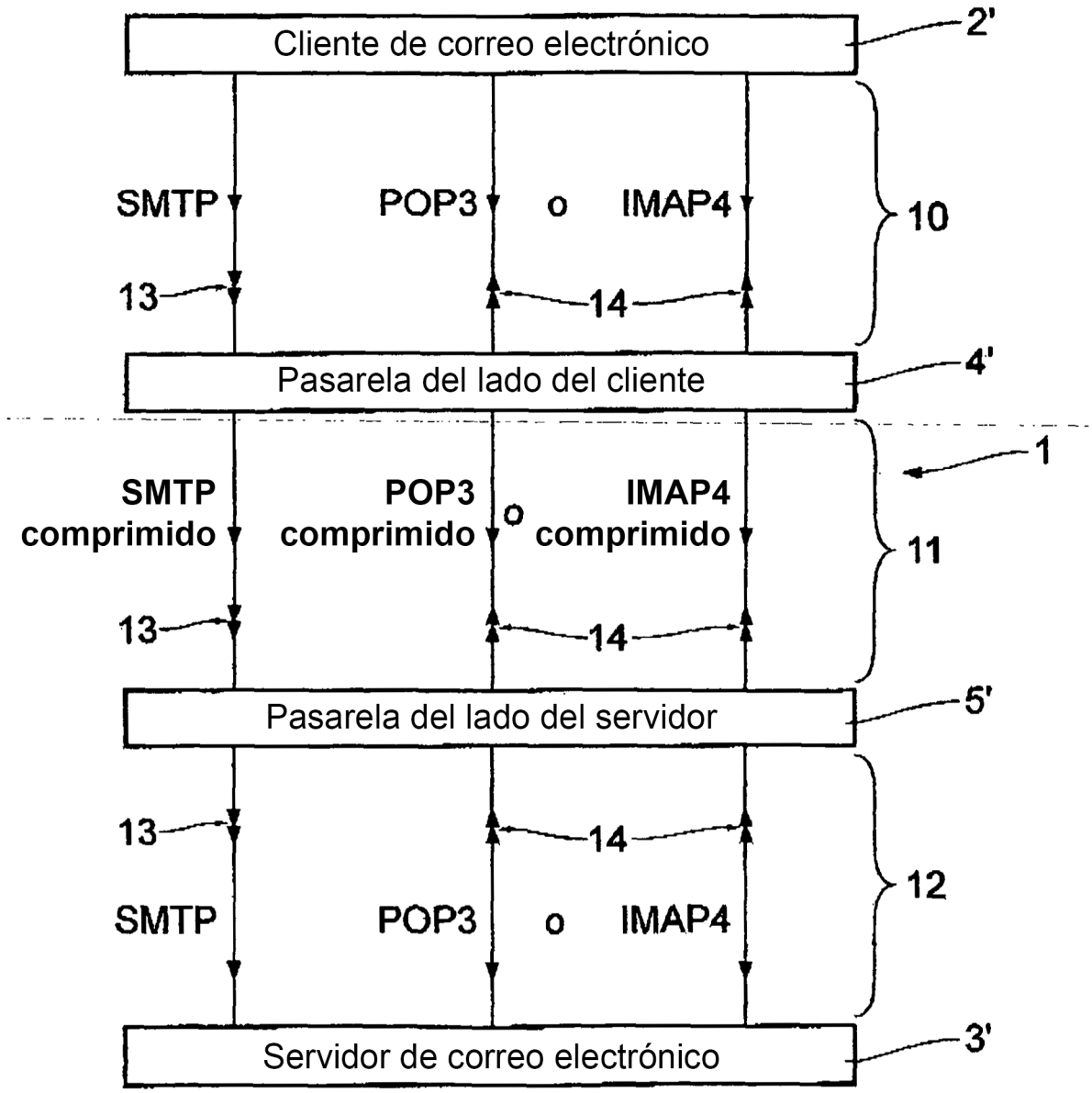
Cliente de correo electrónico
2'
SMTP
POP3
o
IMAP4
10
13
14
Pasarela del lado del cliente
4'
1
SMTP comprimido
POP3 comprimido
o
IMAP4 comprimido
11
13
14
Pasarela del lado del servidor
5'
13
14
12
SMTP
POP3
o
IMAP4
Servidor de correo electrónico
3'

Fig. 2