(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 401 819**

(51) Int. CI.:

H04L 29/08 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **09.06.2008 E 08760741 (2)**

(97) Fecha y número de publicación de la concesión europea: **02.01.2013 EP 2165510**

(54) Título: **Acceso a recursos mediante un módulo de seguridad**

(30) Prioridad:

11.06.2007 DE 102007026870

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
24.04.2013

(73) Titular/es:

GIESECKE & DEVRIENT GMBH (100.0%)
PRINZREGENTENSTRASSE 159
81677 MÜNCHEN, DE

(72) Inventor/es:

HINZ, WALTER y
SPITZ, STEPHAN

(74) Agente/Representante:

ARPE FERNÁNDEZ, Manuel

DESCRIPCIÓN

Acceso a recursos mediante un módulo de seguridad

La invención se refiere en general al campo técnico de los módulos de seguridad y más en particular al campo de la conmutación de un acceso a un recurso mediante un módulo de seguridad. En la selección de palabras utilizada para el presente documento, un módulo de seguridad puede presentar por ejemplo la forma constructiva de una tarjeta chip o de un módulo chip compacto o de un soporte de datos USB.

Ya se conoce el método de utilizar un módulo de seguridad como *proxy* (estación intermedia) en la comunicación entre un equipo terminal y un servidor. El documento WO 2006/029758 A1, por ejemplo, muestra una tarjeta chip que se comunica por una parte con un equipo terminal y por otra parte – mediante el equipo terminal intercalado y una red de ordenadores – con un servidor. Un navegador ejecutado en el equipo terminal aborda la tarjeta chip como *proxy* para la comunicación con el servidor. La tarjeta chip hace aquí las veces de medio auxiliar del navegador. En concreto ejecuta de forma independiente un procedimiento de autentificación si éste resulta necesario.

Sin embargo, en el sistema conocido por el documento WO 2006/029758 A1 existe el problema de que el navegador debe configurarse adecuadamente para hacer posible un funcionamiento de la tarjeta chip como *proxy*. Con este fin puede ser necesario, por ejemplo, introducir una dirección IP de la tarjeta chip en un menú de configuración del navegador. En particular hay que cambiar una y otra vez la configuración del navegador si el equipo terminal se hace funcionar a veces con la tarjeta chip conectada y a veces sin conectar. Esto resulta engorroso y puede menoscabar la aceptación del sistema por parte de los usuarios técnicamente menos versados.

El aporte a conferencia "Internet-Sicherheit mit Security-Token" (seguridad en Internet con testigo de seguridad) de Stephan Spitz y Walter Hinz, D·A·CH Mobility, octubre de 2006, describe la utilización de una tarjeta chip de Internet como *proxy* de autentificación. Por ejemplo, un usuario puede establecer un enlace con un servidor de banca de manera indirecta a través de la tarjeta chip de Internet llamando en primer lugar una página web local en la tarjeta y activando a continuación en esta página web un hipervínculo al servidor de banca. Sin embargo, no se dan a conocer detalles con respecto al funcionamiento de esta solución.

El documento EP 1021020 B1 da a conocer un ordenador personal que ejecuta un navegador y un *proxy* y está conectado a una tarjeta chip. El *proxy* analiza las peticiones HTTP entrantes. Si una de tales peticiones HTTP define un acceso a la tarjeta chip, el *proxy* ejecuta el acceso a la tarjeta chip y dispone al mismo tiempo todas las conversiones de formato necesarias. El *proxy* permite así un acceso remoto a una tarjeta chip 'normal'.

El documento WO 99/56210 A1 muestra un sistema en el que, en una red de comunicación móvil, la funcionalidad de un navegador está repartida entre un cliente y un componente estacionario. El componente estacionario analiza los documentos pedidos por el cliente y sustituye los elementos de datos a recargar por caracteres de sustitución. A continuación, el componente estacionario recarga automáticamente dichos elementos de datos y los transmite al cliente.

El documento JP 2002-312325 A da a conocer una tarjeta chip capaz de autentificarse en varios servicios distintos.

El documento normativo "Hypertext Transfer Protocol - HTTP/1.1" de R. Fielding et al., publicado por *World Wide Web Consortium*, RFC 2616, define peticiones y respuestas según el protocolo HTTP y explica, entre otras cosas, los términos URL, servidor de origen, pasarela, túnel y *proxy*, que en parte se utilizan también en el presente documento.

Por el documento "Traditional IP Network Adress Translator (Traditional NAP)" de P. Srisuresh et al., enero de 2001, disponible el 15.1.2008 en http://tools.ietf.org/html/rfc3022, se conoce el concepto NAT, utilizado también en el presente documento, para la conmutación entre direcciones IP externas e internas.

La invención tiene el objetivo de poner a disposición una técnica segura y/o de fácil utilización para el acceso a un recurso mediante un módulo de seguridad.

Según la invención, este objetivo se logra mediante un procedimiento con las características de la reivindicación 1, un módulo de seguridad según la reivindicación 16 y un producto de programa informático según la reivindicación 17. Las reivindicaciones dependientes se refieren a características opcionales de algunas configuraciones de la invención.

La invención parte de la idea fundamental de abordar el módulo de seguridad no como un *proxy*, sino como un servidor de origen. Esto puede lograrse, por ejemplo, introduciendo en un navegador de un equipo terminal externo un URL que contenga como indicación de anfitrión una dirección del módulo de seguridad. El URL contiene además un identificador. El módulo de seguridad analiza el URL y determina de este modo tanto una indicación de anfitrión como una indicación de ruta del recurso. A continuación, el módulo de seguridad accede al recurso mediante estas indicaciones de anfitrión y de ruta.

La invención ofrece la considerable ventaja de no resultar necesario configurar el navegador para la utilización de un módulo de seguridad configurado como *proxy*. Esto es importante especialmente si el módulo de seguridad ha de ser utilizado por un usuario inexperto o si el navegador se hace funcionar a veces con y a veces sin módulo de seguridad conectado. Además, el módulo de seguridad puede determinar todos los ajustes de seguridad al acceder al recurso y evitar así posibles problemas en caso de una configuración no óptima del navegador.

En algunas formas de realización, el módulo de seguridad presenta una tabla de conversión, a la que se recurre, al menos, para determinar la indicación de anfitrión del recurso. El usuario puede entonces acceder cómodamente al servidor y/o los recursos admisibles según la tabla de conversión. Con esta medida se evitan además ataques basados en indicaciones de anfitrión escritas incorrectamente. En algunas configuraciones, la indicación de ruta del recurso se determina también mediante un acceso a la tabla de conversión, mientras que en otras formas de realización la indicación de ruta forma parte del identificador o puede deducirse esquemáticamente a partir del mismo.

En una configuración preferida está previsto que el módulo de seguridad analice también respuestas entrantes procedentes de un recurso y sustituya, al menos, algunos de los URL contenidos en las mismas por URL modificados. Los URL modificados presentan como indicación de anfitrión una dirección del módulo de seguridad. Esta medida hace que un siguiente acceso al recurso, en el que el usuario siga una remisión contenida en la respuesta, se conduzca también a través del módulo de seguridad.

Además, en muchas configuraciones se sustituyen, al menos, algunas indicaciones de anfitrión y/o de ruta, contenidas en remisiones de una respuesta entrante en el módulo de seguridad, por identificadores según la tabla de conversión. Esta sustitución constituye en muchas formas de realización la pareja de las modificaciones efectuadas durante el procesamiento de peticiones, de modo que al usuario se le presentan identificadores de manera consistente y al servidor se le presentan las correspondientes indicaciones de anfitrión y/o de ruta de manera consistente.

Si una respuesta contiene remisiones con indicaciones de anfitrión y/o de ruta aún desconocidas para el módulo de seguridad, en algunas configuraciones la tabla de conversión se completa de forma dinámica, mientras que en otras configuraciones se sustituyen remisiones con indicaciones de anfitrión y/o de ruta desconocidas por remisiones o instrucciones de seguridad predefinidas.

En algunas formas de realización, el módulo de seguridad presenta una función para la introducción automática de datos de formulario y/o una función de registro.

En algunas configuraciones, el módulo de seguridad es un aparato compacto y protegido contra manipulaciones (*tamperproof device*).

El producto de programa informático puede presentar instrucciones de programa en un soporte de datos físico o no físico, que puede utilizarse por ejemplo en la fabricación o inicialización de un módulo de seguridad según la invención. Un soporte de datos de este tipo puede ser, por ejemplo, un CD-ROM o una memoria de semiconductores o una señal transmitida a través de una red de ordenadores.

De la descripción detallada siguiente de varios ejemplos de realización se desprenden otras características, ventajas y objetivos de la invención. En dicha descripción se remite a los dibujos esquemáticos, en los que:

- la figura 1 muestra un diagrama de bloques de un sistema según un primer ejemplo de realización de la invención durante la transmisión de peticiones,

- la figura 2 muestra un diagrama de bloques del sistema de la figura 1 durante la transmisión de respuestas,

- la figura 3 muestra un diagrama de bloques como el de la figura 2 en un segundo ejemplo de realización,

- la figura 4 muestra un diagrama de bloques como el de la figura 1 en un tercer ejemplo de realización,

- la figura 5 muestra un diagrama de bloques como el de la figura 2 en el tercer ejemplo de realización y

- las figuras 6 y 7 muestran representaciones de un ejemplo de cómo se desarrolla la utilización del módulo de seguridad para acceder a un portal de banca.

En las figuras 1 a 5 se muestran un equipo terminal 10, un servidor 12 y un módulo de seguridad 14. En el presente ejemplo de realización, el equipo terminal 10 es un ordenador personal en el que se ejecuta un navegador de Internet usual, simbolizado en las figuras 1 a 5 por una ventana de navegador 16. En alternativas de realización puede emplearse como equipo terminal 10 también un equipo móvil – por ejemplo un teléfono móvil o un PDA – con un navegador correspondiente.

El servidor 12 está configurado como un ordenador de alto rendimiento o como una agrupación (*cluster)* de ordenadores. El servidor 12 pone a disposición un recurso 18 al que un usuario desea acceder desde el navegador

del equipo terminal 10. El recurso 18 puede ser por ejemplo una página de Internet o una aplicación de Internet, por ejemplo un portal de banca.

El equipo terminal 10 y el servidor 12 están conectados a una red de ordenadores 20, por ejemplo Internet o una intranet. Además, el módulo de seguridad 14 está conectado al equipo terminal 10, por ejemplo mediante una interfaz USB o una interfaz para tarjetas chip. En las figuras 1 a 5, estas conexiones están representadas mediante flechas continuas, mientras que los procesos de comunicación se ilustran con flechas de trazo discontinuo.

En el presente ejemplo de realización, el módulo de seguridad 14 está configurado como una tarjeta chip con un procesador 22, una memoria 24 y una interfaz de comunicación 26. La memoria 24 presenta un paquete de programas 28 que, entre otras cosas, pone a disposición un sistema operativo 30, una pila de protocolos 32 y un programa de conmutación 34 para implementar la presente invención. La memoria 24 contiene además una tabla de conversión 36 y un archivo de registro 38.

El módulo de seguridad 14 está diseñado para recibir peticiones (*requests*) del equipo terminal 10 y enviar respuestas (*responses*) al equipo terminal 10. Para ello se utiliza en el nivel físico el protocolo predefinido por la interfaz 26, por ejemplo USB. En los niveles superiores se emplean los protocolos puestos a disposición por la pila de protocolos 32. Éstos son, por ejemplo, SLIP o RNDIS o CDC o EEM para la capa de enlace y TCP/IP para la capa de red y la capa de transporte. El programa de conmutación 34 implementa la comunicación en la capa de aplicación, por ejemplo mediante HTTP o HTTPS.

La comunicación entre el equipo terminal 10 y el módulo de seguridad 14 se realiza en una red privada con un rango de direcciones local, por ejemplo el rango de direcciones IP 192.168.0.x. El equipo terminal 10 se comunica además con el servidor 12 mediante la red de ordenadores externa 20. Además, el equipo terminal 10 conmuta entre la red privada y la red de ordenadores 20 mediante una función NAT-Router (NAT = *Network Address Translation* = conversión de direcciones de red), puesta a disposición por el sistema operativo del equipo terminal 10 o por un programa adicional instalado en el equipo terminal 10. Por lo tanto, el módulo de seguridad 14 puede acceder al servidor 12 mediante el equipo terminal 10 y la red de ordenadores 20.

Las funciones de comunicación brevemente expuestas en los dos párrafos anteriores son en sí conocidas y se explican, por ejemplo, en el aporte a conferencia "Internet-Sicherheit mit Security-Token" mencionado al principio.

A continuación se describe con referencia a las figuras 1 y 2 un ejemplo de un proceso de comunicación en el que un usuario del equipo terminal 10 accede al recurso 18 mediante el navegador ejecutado en el equipo terminal 10 y mediante el módulo de seguridad 14. El módulo de seguridad 14 sirve aquí de estación lógica intermedia para la transmisión de datos HTTP(S) entre el equipo terminal 10 y el servidor 12. Sin embargo, el módulo de seguridad 14 no se aborda como un *proxy* u otra estación intermedia, sino como un servidor de origen (*origin server*). Esto se logra mediante la utilización de URL especiales (URL = *Uniform Resource Locator* = localizador uniforme de recursos), que, si bien corresponden en su sintaxis a las normas pertinentes – por ejemplo RFC2616 –, tienen un significado modificado con relación al significado normal.

El proceso comienza cuando el usuario teclea un URL especial 40 en una barra de direcciones de la ventana del navegador 16 o sigue una remisión que contiene un URL 40 de este tipo. Al igual que un URL conforme a la norma, el URL 40 contiene una indicación *PROT* del protocolo a utilizar, una indicación de anfitrión y una indicación de ruta. En el presente ejemplo, la comunicación entre el equipo terminal 10 y el módulo de seguridad 14 se basa en HTTP como protocolo, de modo que *PROT* quiere decir "http:". En variantes de realización puede emplearse en cambio HTTPS u otro protocolo protegido.

Como indicación de anfitrión, el URL 40 contiene una dirección $HOST_{SM}$, en la que puede accederse al módulo de seguridad 14 desde el equipo terminal 10. La dirección $HOST_{SM}$ puede ser, por ejemplo, una dirección IP numérica o un nombre de anfitrión, si en la red privada formada por el equipo terminal 10 y el módulo de seguridad 14 está implementada una función DNS-Server.

Así pues, el módulo de seguridad 14 se aborda como un servidor de origen – y no como un *proxy* –, por lo que no es necesaria una configuración especial del navegador.

Como indicación de ruta, el URL 40 contiene un identificador ID_1. El identificador ID_1 tiene la sintaxis de una ruta de URL según el estándar HTTP. Sin embargo, el módulo de seguridad 14 no interpreta el identificador ID_1 como una ruta de un sistema de archivos local, sino que más bien determina a partir del identificador ID_1 el recurso deseado – por ejemplo el recurso 18 –, que se halla fuera del módulo de seguridad 14 en un servidor externo, por ejemplo el servidor 12. Esto incluye, como se explicará más adelante con mayor detalle, la determinación de una indicación de anfitrión y una indicación de ruta del recurso 18 en función del identificador ID_1. Al contrario de lo que ocurre con la significación usual de los URL, en la presente invención la indicación de anfitrión del URL 40 no puede utilizarse para la identificación del servidor externo 12, porque esta identificación de anfitrión se necesita para la dirección $HOST_{SM}$ del módulo de seguridad 14.

El navegador ejecutado en el equipo terminal 10 convierte el URL 40 en una primera petición 42 – en el presente ejemplo una petición HTTP –, que se envía directamente al módulo de seguridad 14. La primera petición 42 contiene

como indicación de anfitrión la dirección $HOST_{SM}$ del módulo de seguridad 14 y como indicación de ruta el identificador ID_1. Por ejemplo, si el URL 40 reza "http://192.168.0.2/Bankportal", puede enviarse la siguiente primera petición 42 al módulo de seguridad 14:

GET Bankportal HTTP/1.1

HOST: 192.168.0.2

En el módulo de seguridad 14 se convierte la primera petición entrante 42 en una segunda petición 44. Para ello, en el presente ejemplo de realización, el programa de conmutación 34 accede a la tabla de conversión 36 con el fin de obtener del identificador contenido en la primera petición 42 un URL del recurso deseado. En el ejemplo mostrado en la figura 1, la tabla de conversión 36 presenta una entrada que asigna el URL aquí designado con URL_1 al identificador ID_1. La parte aquí relevante de la tabla de conversión 36 puede ser por ejemplo como se indica a continuación:

Identificador	**URL**
ID_1 = Bankportal	URL_1 = https://direkt.postbank.de/portalApp/index.jsp
...	...

Una vez que el programa de conmutación 34 ha encontrado en la tabla de conversión 36 la entrada con el identificador ID_1, determina a partir del URL URL_1 correspondiente el protocolo $URL_{1,PROT}$ – aquí https – a utilizar para el acceso al recurso, así como una indicación de anfitrión $URL_{1,HOST}$ del recurso 18 – aquí "direkt.postbank.de" – y una indicación de ruta $URL_{1,PATH}$ del recurso 18 – aquí "portalApp/index.jsp" –. Se entiende que la indicación de ruta de un recurso puede ser también la ruta vacía "/".

En el ejemplo de realización aquí descrito, cada entrada de la tabla de conversión 36 contiene todo el URL como una cadena de caracteres, mientras que en ciertas alternativas de realización el protocolo, la indicación de anfitrión y la indicación de ruta pueden estar contenidos en campos de datos separados de la tabla de conversión 36. Además, en algunas configuraciones, la tabla de conversión 36 puede presentar varias partes. Una parte se direcciona mediante una primera sección del identificador y contiene las indicaciones de anfitrión y, en caso dado, el protocolo respectivo, mientras que otra parte se direcciona mediante una segunda sección del identificador y contiene las indicaciones de ruta.

El programa de conmutación 34 genera ahora la segunda petición 44, que contiene la indicación de anfitrión $URL_{1,HOST}$ y la indicación de ruta $URL_{1,PATH}$ del URL URL_1. La segunda petición 44 se envía mediante el equipo terminal 10 y la red de ordenadores 20 al servidor designado por la indicación de anfitrión $URL_{1,HOST}$, aquí por ejemplo el servidor 12. Para ello se utiliza el protocolo especificado $URL_{1,PROT}$.

Por consiguiente, si, como en el ejemplo aquí descrito, ha de emplearse el protocolo HTTPS para la comunicación entre el módulo de seguridad 14 y el servidor 12, se establece en primer lugar, de manera en sí ya conocida, un canal de transmisión de datos protegido entre el módulo de seguridad 14 y el servidor 12. Para ello se autentifica el servidor 12, comprobando el módulo de seguridad 14 un certificado del mismo. Además se negocia entre el módulo de seguridad 14 y el servidor 12 una clave de sesión con la que a continuación se codifican todos los demás procesos de comunicación. Estas funciones son independientes del equipo terminal 10, de modo que ni siquiera una configuración incorrecta o insegura del equipo terminal 10 y/o del navegador ejecutado en el mismo supone riesgo alguno para la seguridad.

Si, como en el ejemplo recién descrito, la comunicación entre el módulo de seguridad 14 y el servidor 12 se realiza a través de un canal protegido, se impide con fiabilidad toda influencia no deseada del equipo terminal 10. Aunque el módulo de seguridad 14 se comunica con el servidor 12 mediante el protocolo HTTP no protegido, el equipo terminal 10 únicamente conmuta las peticiones y respuestas, sin evaluar ni modificar su contenido.

Como se muestra en las figuras 2 y 3, el servidor 12 responde a la segunda petición 44 con una primera respuesta 46, que llega al módulo de seguridad 14 a través de la red de ordenadores 20 y el equipo terminal 10. El programa de conmutación 34 del módulo de seguridad 14 analiza la primera respuesta 46 en cuanto a remisiones que contengan un URL o un identificador local de recursos. Como se describe más adelante con mayor detalle, el programa de conmutación 34 modifica algunas de estas remisiones, o todas ellas, para adaptarlas al esquema del URL 40. La respuesta correspondientemente modificada se emite entonces como segunda respuesta 48 al equipo terminal 10 y se visualiza en el mismo en la ventana del navegador 16. El análisis y la modificación de las

respuestas por el programa de conmutación 34 afecta no sólo a remisiones contenidas en páginas HTML, sino por ejemplo también a remisiones en respuestas HTTP que provoquen un reencaminado (*redirection*).

La modificación de remisiones por el programa de conmutación 34 hace que se aborde de nuevo el módulo de seguridad 14 cuando el usuario hace clic en una remisión modificada o el navegador sigue una instrucción de reencaminado HTTP. Además, en las remisiones modificadas se utilizan los identificadores de la tabla de conversión 36 – por ejemplo ID_1 – en lugar de los URL correspondientes – por ejemplo URL_1 –. Con esta medida, el usuario sólo puede ver en la barra de direcciones de la ventana del navegador 16 los identificadores respectivos y no el URL asignado.

La figura 2 muestra a modo de ejemplo una forma de realización en la que el programa de conmutación 34 modifica sólo los URL o identificadores de recursos que estén contenidos en remisiones de la primera respuesta 46 y para los que exista una entrada en la tabla de conversión 36. Esto es aplicable por ejemplo al URL URL_1 de la primera respuesta 46. El programa de conmutación 34 genera en la segunda respuesta 48, a partir del URL URL_1, un URL modificado que presenta como indicación de protocolo el protocolo *PROT* utilizado para la comunicación entre el equipo terminal 10 y el módulo de seguridad 14, como indicación de anfitrión la dirección $HOST_{SM}$ del módulo de seguridad 14 y como indicación de ruta el identificador ID_1 de la entrada en la tabla de conversión 36. Por lo tanto, el URL modificado en la segunda repuesta 48 tiene, al igual que el URL 40, la forma siguiente:

$$PROT\ //\ HOST_{SM}\ /\ ID_1$$

En el ejemplo de realización según la figura 2, el URL URL_2 contenido también en la primera respuesta 46 se deja sin cambios en la segunda respuesta 48, porque la tabla de conversión 36 no contiene ninguna entrada para el URL URL_2. Si el usuario sigue la remisión correspondiente abandona la comunicación protegida con el módulo de seguridad 14.

En una modificación de la configuración según la figura 2, las URL para las que no existe ninguna entrada en la tabla de conversión 36 se borran en la segunda respuesta 48 o se sustituyen por una instrucción de seguridad o por una remisión predeterminada, por ejemplo a una página explicativa. Esta medida asegura que el usuario no abandone sin querer el área de recursos accesibles predefinida por el módulo de seguridad 14.

La forma de realización ilustrada en la figura 3 se diferencia de la configuración según la figura 2 en que la tabla de conversión 36 se amplía dinámicamente. Esto puede ser deseable para poder registrar en la tabla de conversión 36 URL no previsibles, por ejemplo URL con un identificador de sesión.

En la configuración según la figura 3, todos los URL e identificadores locales de recursos contenidos en remisiones en la primera respuesta 46 se modifican en la forma arriba descrita. Si la tabla de conversión 36 contiene ya una entrada apropiada para un URL – aquí por ejemplo la entrada para el URL URL_1 –, al generar el URL modificado se utiliza el identificador indicado en esta entrada – aquí por ejemplo el identificador ID_1 –. Sin embargo, si en la tabla de conversión 36 no existe aún ninguna entrada para un URL o un identificador local de recursos – como es el caso por ejemplo para el URL URL_2 –, el programa de conmutación 34 determina un nuevo identificador – aquí por ejemplo el identificador ID_2. El programa de conmutación 34 crea una nueva entrada con los valores ID_2 y URL_2 en la tabla de conversión 36. El programa de conmutación 34 genera además el URL modificado en la segunda respuesta 48 utilizando el nuevo identificador ID_2, como se muestra en la figura 3.

En otras configuraciones se aplican estrategias de sustitución modificadas. Por ejemplo, los URL contenidos en la primera respuesta 46 pueden recibir un tratamiento distinto al que reciben los identificadores locales de recursos. Como alternativa o adicionalmente, en algunas formas de realización no se crean en la tabla de conversión 36 nuevas entradas para todos, sino sólo para algunos URL y/o identificadores de recursos aún desconocidos. Por ejemplo, puede estar previsto crear una nueva entrada sólo si un URL contenido en la primera respuesta 46 presenta una de varias indicaciones de anfitrión predeterminadas. Como ya se ha descrito anteriormente, las remisiones con otros URL se dejan sin cambios en la segunda respuesta o bien se sustituyen por otras indicaciones – por ejemplo una instrucción de seguridad – no extraídas de la tabla de conversión 36.

En los ejemplos de realización hasta ahora descritos, las entradas de la tabla de conversión 36 contenían en cada caso el identificador completo del URL 40 – por ejemplo ID_1 – y el URL correspondiente completo – por ejemplo URL_1 –. En alternativas de realización, el identificador presenta en cambio varios componentes, de los cuales sólo uno sirve para el direccionamiento de una entrada en la tabla de conversión 36. Al menos uno de los otros componentes del identificador se procesa independientemente de la tabla de conversión 36. Este componente puede por ejemplo incluirse sin cambios en el URL modificado, o pueden efectuarse únicamente sustituciones esquemáticas – por ejemplo sustituciones de caracteres especiales –.

Las figuras 4 y 5 ilustran un ejemplo de realización de este tipo, en el que el identificador ID_1 contenido en el URL 40 presenta dos componentes: un identificador de anfitrión $ID_{1,HOST}$ y una indicación de ruta $ID_{1,PATH}$. El identificador de anfitrión $ID_{1,HOST}$ sirve para la identificación del servidor 12 y, mediante la tabla de conversión 36, se obtiene del mismo una dirección de anfitrión $URL_{1,HOST}$. La indicación de ruta $ID_{1,PATH}$ designa la ruta absoluta del recurso 18 deseado, en una forma habitual para las indicaciones de ruta de URL. Todo el URL 40 sigue sintácticamente las normas HTTP; el URL 40 puede ser por ejemplo como sigue:

http://192.168.0.2/Bank/ portalApp/ index.jsp

En el URL 40 que se acaba de indicar, "192.168.0.2" es la dirección $HOST_{SM}$ del módulo de seguridad 14, "Bank" es el identificador de anfitrión $ID_{1,HOST}$ y "/portalApp/index.jsp" es la indicación de ruta $ID_{1,PATH}$.

El módulo de seguridad 14 recibe esta información en la primera petición 42 y busca el identificador de anfitrión $ID_{1,HOST}$ en la tabla de conversión 36 para determinar la dirección de anfitrión $URL_{1,HOST}$ correspondiente. La parte aquí relevante de la tabla de conversión 36 puede ser por ejemplo como se indica a continuación:

Identificador de anfitrión	Dirección de anfitrión
$ID_{1,HOST}$ = Bank	$URL_{1,HOST}$ = direkt.postbank.de
...	...

El módulo de seguridad 14 genera ahora la segunda petición 44 con la dirección de servidor $URL_{1,HOST}$ extraída de la tabla de conversión 36 y la indicación de ruta $ID_{1,PATH}$ extraída del identificador ID_1. En el presente ejemplo, la segunda petición 44 es entonces como sigue:

GET /portalApp/index.jsp HTTP/1.1

HOST: direkt.postbank.de

Así, en el ejemplo de realización según la figura 4, la segunda petición 44 es idéntica a la segunda petición 44 del ejemplo de realización según la figura 1. Los dos ejemplos de realización tampoco se diferencian en lo relativo al envío de la segunda petición al servidor 12 y la recepción de la primera respuesta 46. Sin embargo, existen diferencias en el procesamiento de la primera respuesta 46, como se muestra en la figura 5. Análogamente al procesamiento de peticiones, en el procesamiento de respuestas para obtener URL modificados el módulo de seguridad 14 registra respectivo sólo una indicación de anfitrión determinada a partir de la tabla de conversión 36, mientras que la indicación de ruta contenida en el URL de la primera respuesta 46 permanece inalterada o se modifica sólo esquemáticamente.

En el ejemplo representado en la figura 5, la primera respuesta 46 contiene una remisión con un URL URL_1, que presenta una indicación de anfitrión $URL_{1,HOST}$ y una indicación de ruta $URL_{1,PATH}$. El programa de conmutación 34 ejecutado en el módulo de seguridad 14 analiza el URL URL_1, determina la entrada de la tabla de conversión 36 asignada a la indicación de anfitrión $URL_{1,HOST}$ y genera un URL modificado para la segunda respuesta 48. De manera similar a lo que ocurría en los ejemplos de realización según las figuras 2 y 3, el URL modificado está construido de modo que el desarrollo de la comunicación entre el equipo terminal 10 y el módulo de seguridad 14 se continúe si el usuario sigue la remisión con el URL modificado.

Por lo tanto, como se muestra en la figura 5, el URL modificado contiene como indicación de protocolo el protocolo *PROT* utilizado para la comunicación entre el equipo terminal 10 y el módulo de seguridad 14, como indicación de anfitrión la dirección $HOST_{SM}$ del módulo de seguridad 14 y como indicación de ruta una ruta que se compone del identificador de anfitrión $ID_{1,HOST}$ según la entrada de la tabla de conversión 36 y la indicación de ruta $URL_{1,PATH}$ del URL URL_1. En suma resulta de ello en la segunda respuesta 48 una remisión con un URL que presenta la forma siguiente:

$$PROT\ //\ HOST_{SM}\ /\ ID_{1,HOST}\ URL_{1,PATH}$$

De manera similar a cómo ya se ha descrito en relación con las figuras 2 y 3, en el ejemplo mostrado en la figura 5 la situación de que la tabla de conversión 36 no contenga ninguna entrada para una indicación de anfitrión – aquí por ejemplo $URL_{2,HOST}$ – contenida en un URL entrante puede tratarse también de forma diferente. En algunas configuraciones, el URL en cuestión puede incluirse sin cambios en la segunda respuesta 48 o sustituirse por un URL predeterminado o borrarse o sustituirse por una instrucción de seguridad. En cambio, en otras configuraciones se amplía la tabla de conversión 36 dinámicamente. Con este fin, el programa de conmutación 34 genera un nuevo identificador de anfitrión – por ejemplo $ID_{2,HOST}$ – y lo escribe junto con la indicación de anfitrión del URL – por ejemplo $URL_{2,HOST}$ – en una nueva entrada de la tabla de conversión 36. En la segunda respuesta 48 se genera a continuación un URL modificado, con la forma antes descrita.

En otra modificación de la configuración mostrada en las figuras 4 y 5 se utilizan identificadores de anfitrión – por ejemplo $ID_{1,HOST}$ – que pueden convertirse en indicaciones de anfitrión correspondientes – por ejemplo $URL_{1,HOST}$ – sin recurrir a una tabla de conversión. Por ejemplo es posible elegir los identificadores de anfitrión de manera que éstos puedan convertirse esquemáticamente – por ejemplo sustituyendo caracteres de división – en indicaciones de anfitrión conformes con la norma. En un caso extremo es incluso posible utilizar como identificadores de anfitrión indicaciones de anfitrión conformes con la norma, de modo que ni siquiera sea necesaria una conversión. Las

configuraciones sin tabla de conversión son particularmente sencillas y ahorran espacio de memoria. Sin embargo, existe el riesgo de que se menoscabe la seguridad, porque es posible que los usuarios llamen recursos a los que no se podría acceder con las entradas predefinidas de una tabla de conversión.

En cambio, en todos los ejemplos de realización que emplean la tabla de conversión 36, al menos, el principio de la comunicación está limitado a los recursos accesibles a través de un identificador y una entrada correspondiente en la tabla de conversión 36. Esto impide por ejemplo intentos de estafa en los que se atrae a usuarios a páginas falsificadas cuyo URL es tan similar al URL de una página deseada que puede confundirse con el mismo. Además, para el usuario puede resultar más cómodo utilizar un identificador en lugar de un URL, que posiblemente sea largo. Por motivos de seguridad también puede ser ventajoso que no sea necesario dar a conocer al usuario el URL del recurso deseado.

El contenido inicial de la tabla de conversión 36 se establece en un proceso de instalación (*Setup-Phase*). Este proceso de instalación puede efectuarlo por ejemplo el fabricante del módulo de seguridad 14 o un proveedor de servicios – por ejemplo un banco que ofrezca a sus clientes el módulo de seguridad 14 – o también el cliente final. Para ello, el módulo de seguridad 14 ofrece una interfaz de configuración web protegida por contraseña, a través de la cual pueden incorporarse entradas a la tabla de conversión 36.

La interfaz de configuración web sirve también para ajustar todos los demás parámetros de servicio del módulo de seguridad 14. Así, por ejemplo, es posible ajustar el módulo de seguridad 14 de modo que – independientemente de las funciones de seguridad ya descritas – permita el acceso sólo a determinados recursos o determinadas páginas web. Otras posibilidades de ajuste se refieren por ejemplo al tratamiento que el módulo de seguridad 14 da a las *cookies* durante la comunicación con el servidor 12.

En todas las configuraciones aquí descritas, el módulo de seguridad 14 puede presentar opcionalmente otras funciones. En este caso se trata en particular de funciones relevantes para la seguridad o funciones que utilizan la memoria de datos contenida en el módulo de seguridad 14 y/o la potencia de cálculo del procesador 22. Una función de este tipo es, por ejemplo, que el módulo de seguridad 14 pueda almacenar datos de registro para el acceso a recursos externos e introducirlos automáticamente en forma de datos de formulario. Esta función se describe a continuación haciendo referencia a las figuras 6 y 7 en el ejemplo de una introducción de datos de registro en un portal bancario.

Según la figura 6, el desarrollo comienza cuando, en la etapa 50, el usuario conecta el módulo de seguridad 14 al equipo terminal 10 y llama una página-índice puesta a disposición por el módulo de seguridad 14. El usuario puede hacer esto último, por ejemplo, introduciendo una dirección del módulo de seguridad 14 en la barra de direcciones de la ventana del navegador 16. Acto seguido, el módulo de usuario 14 pide en la etapa 52 una contraseña (PIN). En cuanto el usuario ha introducido la contraseña (etapa 54) y ésta ha sido comprobada con éxito por el módulo de seguridad 14 (etapa 56), el módulo de seguridad 14 envía en la etapa 58 la página-índice al navegador ejecutado en el equipo terminal 10.

La página-índice predefinida por el módulo de seguridad 14 presenta al usuario diversas ofertas soportadas por el módulo de seguridad 14. En el desarrollo mostrado en la figura 7 a modo de ejemplo, en la etapa 60 el usuario hace clic en una remisión que contiene un identificador de un portal bancario, por ejemplo una remisión con el URL 40. Acto seguido, el navegador envía la primera petición 42, que el módulo de seguridad 14 convierte en la segunda petición 44 en la forma ya descrita, para llamar la página del portal bancario correspondiente (etapa 62).

La primera respuesta 46 recibida del servidor 12 representa una página de inicio del portal bancario, en la que se han de introducir datos de registro adecuados, por ejemplo un número de cuenta y un PIN. El módulo de seguridad 14 modifica en primer lugar en la forma ya descrita las remisiones contenidas en la primera respuesta 46 (etapa 64). Además, en la etapa 66, el módulo de seguridad 14 introduce caracteres de sustitución en los campos de entrada de la página de inicio ("máscara de usuario"). La página de inicio así modificada se transmite en la etapa 68 al equipo terminal 10 como segunda respuesta 48 y se visualiza en el mismo en la ventana del navegador 16.

En la etapa 70, el usuario confirma la página de inicio haciendo clic en una remisión correspondiente (modificada). Acto seguido, el navegador envía una nueva primera petición 42 al módulo de seguridad 14. El módulo de seguridad 14 genera a partir de ésta una nueva segunda petición 44. En la etapa 72 se sustituyen los caracteres de sustitución de los campos de entrada por los datos de registro verdaderos. En la etapa 74, el módulo de seguridad 14 determina una dirección de destino (indicación de anfitrión e indicación de ruta) y, en la etapa 76, envía la nueva segunda petición 44 al servidor 12.

En la etapa 78, el módulo de seguridad 14 recibe entonces una página del servidor 12, que se procesa como nueva primera respuesta 46 y se transmite al equipo terminal 10 como nueva segunda respuesta 48. La comunicación puede continuar el tiempo que se desee en la forma descrita.

En todas las configuraciones aquí descritas, el módulo de seguridad 14 puede presentar como complemento adicional una función de registro (*logging*). En ésta, el módulo de seguridad 14 registra en el archivo de registro persistente 38 un juego ajustable de parámetros y valores de los procesos de comunicación llevados a cabo. Éstos pueden ser por ejemplo direcciones IP llamadas, horofechadores, el número de paquetes de datos transmitidos,

contadores de secuencias, etc. Estos parámetros y valores pueden determinarse con poco esfuerzo, porque la pila de protocolos 32 del módulo de seguridad 14 desarrolla toda la comunicación y, por lo tanto, todos los parámetros y valores están disponibles en el módulo de seguridad 14.

Los datos registrados pueden ser leídos del módulo de seguridad 14 por una instancia autorizada que se identifique mediante una clave o una contraseña. Como alternativa o adicionalmente, el módulo de seguridad 14 puede también enviar los datos directamente a un servidor predeterminado.

La idea que se acaba de describir de una "tarjeta chip de Internet elaboradora de registros" se considera no sólo como un complemento de las funciones del módulo de seguridad 14, sino también como una invención autónoma, independiente de las características de un módulo de seguridad abordado como servidor de origen y una conversión de peticiones en el módulo de seguridad. Las elaboraciones de registros pueden soportarse por ejemplo en las siguientes aplicaciones:

- transmisión de contenido VoIP o multimedia con (S)RTP y/o SIP y/o MIKEY y/o DTLS,
- transmisión de correo electrónico con (S)MIME,
- transmisión HTML con HTTP(S) y
- RPC (*Remote Procedure Calls*), por ejemplo mediante SOAP y HTTP(S).

Los parámetros y valores que pueden registrarse pueden ser parámetros del protocolo TCP/IP o UDP/IP subyacente, del protocolo de seguridad, como SSL/TLS, IPSec, DTLS o MIKEY, o también parámetros del protocolo de aplicación, como (S)RTP, (S)MIME, SOAP o HTTP(S). Estos parámetros y valores pueden extraerse de las etiquetas de cabecera de protocolo de los mensajes intercambiados y almacenarse en el archivo de registro 38.

Un registro realizado en el módulo de seguridad 14 y por lo tanto protegido contra manipulaciones puede utilizarse, por ejemplo, para aportar pruebas de conexión, por ejemplo una prueba del envío de un correo electrónico o una prueba del acceso a un recurso determinado. Otra aplicación es el cálculo del pago (*billing*) a partir de la información registrada. La elección en cuanto a cuáles de los posibles parámetros y valores a registrar, y durante cuánto tiempo han de registrarse, se toma en función del uso previsto.

Se entiende que las formas de realización y variantes de realización aquí descritas deben considerarse únicamente como ejemplos. El técnico en la materia puede inferir directamente otras modificaciones y combinaciones de las características aquí descritas.

REIVINDICACIONES

1. Procedimiento para acceder a un recurso (18), realizándose dicho procedimiento mediante un módulo de seguridad portátil (14) configurado como una tarjeta chip o un módulo chip o un soporte de datos USB e incluyendo el procedimiento:

- recepción de una primera petición (42) procedente de un equipo terminal (10) que se halla situado en una red privada con el módulo de seguridad (14), primera petición (42) que aborda el módulo de seguridad (14) como un servidor de origen y contiene un identificador (ID_1) asignado al recurso (18),

- determinación de, al menos, una indicación de anfitrión ($URL_{1,HOST}$) y una indicación de ruta ($URL_{1,PATH}$, $ID_{1,PATH}$) del recurso (18) en función del identificador (ID_1) contenido en la primera petición (42),

- envío, mediante una red de ordenadores externa (20), de una segunda petición (44) que contiene, al menos, la indicación de ruta determinada ($URL_{1,PATH}$, $ID_{1,PATH}$) a un servidor (12) designado por la indicación de anfitrión ($URL_{1,HOST}$) determinada,

- recepción de una primera respuesta (46) procedente del servidor (12) en respuesta a la segunda petición (44), primera respuesta (46) que contiene al menos una remisión con un URL (URL_1, URL_2),

- modificación del URL (URL_1, URL_2) de la primera respuesta (46) para obtener una segunda respuesta (48), presentando el URL modificado una dirección ($HOST_{SM}$) del módulo de seguridad (14) como indicación de anfitrión, y

- envío de la segunda respuesta (48) al equipo terminal (10).

2. Procedimiento según la reivindicación 1, caracterizado porque el módulo de seguridad (14) determina, al menos, la indicación de anfitrión ($URL_{1,HOST}$) a partir de una tabla de conversión (36) almacenada en el módulo de seguridad (14).

3. Procedimiento según la reivindicación 2, caracterizado porque el módulo de seguridad (14) también determina la indicación de ruta ($URL_{1,PATH}$) a partir de la tabla de conversión (36).

4. Procedimiento según la reivindicación 2, caracterizado porque el módulo de seguridad (14) determina la indicación de ruta ($ID_{1,PATH}$) a partir del identificador (ID_1) independientemente de la tabla de conversión (36).

5. Procedimiento según la reivindicación 4, caracterizado porque el módulo de seguridad (14) determina la indicación de ruta ($ID_{1,PATH}$) como parte del identificador (ID_1).

6. Procedimiento según la reivindicación 5, caracterizado porque, en función de, al menos, una parte (URL_1, URL_2, $URL_{1,HOST}$) del URL (URL_1, URL_2) contenido en la primera respuesta (46), se determina un identificador (ID_1, ID_2, $ID_{1,HOST}$) que constituye, al menos, una parte de una indicación de ruta del URL modificado.

7. Procedimiento según la reivindicación 6 en combinación con una de las reivindicaciones 2 a 5, caracterizado porque el identificador (ID_1, ID_2, $ID_{1,HOST}$) se determina a partir de la tabla de conversión (36).

8. Procedimiento según la reivindicación 7, caracterizado porque se determina una entrada de la tabla de conversión (36) que presenta el URL (URL_1, URL_2) contenido en la primera respuesta (46) y presenta también el identificador (ID_1, ID_2) y porque la indicación de ruta del URL modificado está formada por el identificador (ID_1, ID_2).

9. Procedimiento según la reivindicación 7, caracterizado porque se determina una entrada de la tabla de conversión (36) que presenta una indicación de anfitrión ($URL_{1,HOST}$) del URL (URL_1) contenido en la primera respuesta (46) y presenta también el identificador en forma de un identificador de anfitrión ($ID_{1,HOST}$) y porque la indicación de ruta del URL modificado está formada por el identificador de anfitrión ($ID_{1,HOST}$) y una indicación de ruta ($URL_{1,PATH}$) del URL (URL_1) contenido en la primera respuesta (46).

10. Procedimiento según una de las reivindicaciones 7 a 9, caracterizado porque la tabla de conversión (36) se amplía dinámicamente con identificadores recién determinados (ID_2, $ID_{2,HOST}$).

11. Procedimiento según una de las reivindicaciones 1 y 6 a 10, caracterizado porque la primera respuesta (46) contiene al menos una remisión con un URL (URL_2) que, al crearse la segunda respuesta (48), se incluye sin cambios en la segunda respuesta (48) o se sustituye por una remisión con un URL predeterminado o se borra o se sustituye por una instrucción de seguridad.

12. Procedimiento según una de las reivindicaciones 1 a 11, caracterizado porque el módulo de seguridad (14) presenta además una función para la introducción automática de datos de formulario y/o una función de registro.

13. Módulo de seguridad portátil (14) con un procesador (22) y al menos una memoria (24), caracterizado porque el módulo de seguridad (14) está preparado para ejecutar un procedimiento según una de las reivindicaciones 1 a 12.

14. Producto de programa informático que presenta instrucciones de programa para un procesador (22) de un módulo de seguridad (14), produciendo las instrucciones de programa al procesador (22) del módulo de seguridad (14) la ejecución de un procedimiento según una de las reivindicaciones 1 a 12.

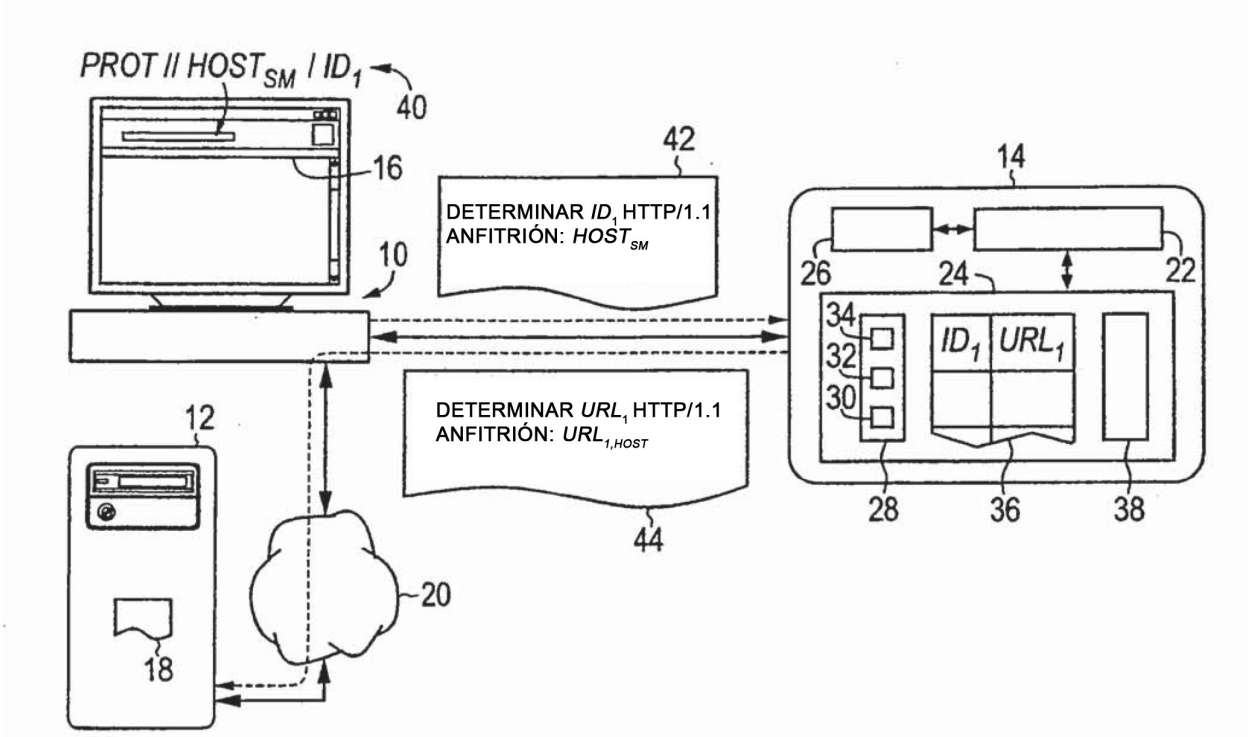
PROT // HOST$_{SM}$ / ID$_1$
40
16
42
DETERMINAR ID$_1$ HTTP/1.1
ANFITRIÓN: HOST$_{SM}$
14
10
26
24
22
34
32
30
ID$_1$
URL$_1$
12
DETERMINAR URL$_1$ HTTP/1.1
ANFITRIÓN: URL$_{1,HOST}$
28
36
38
44
20
18

Fig. 1

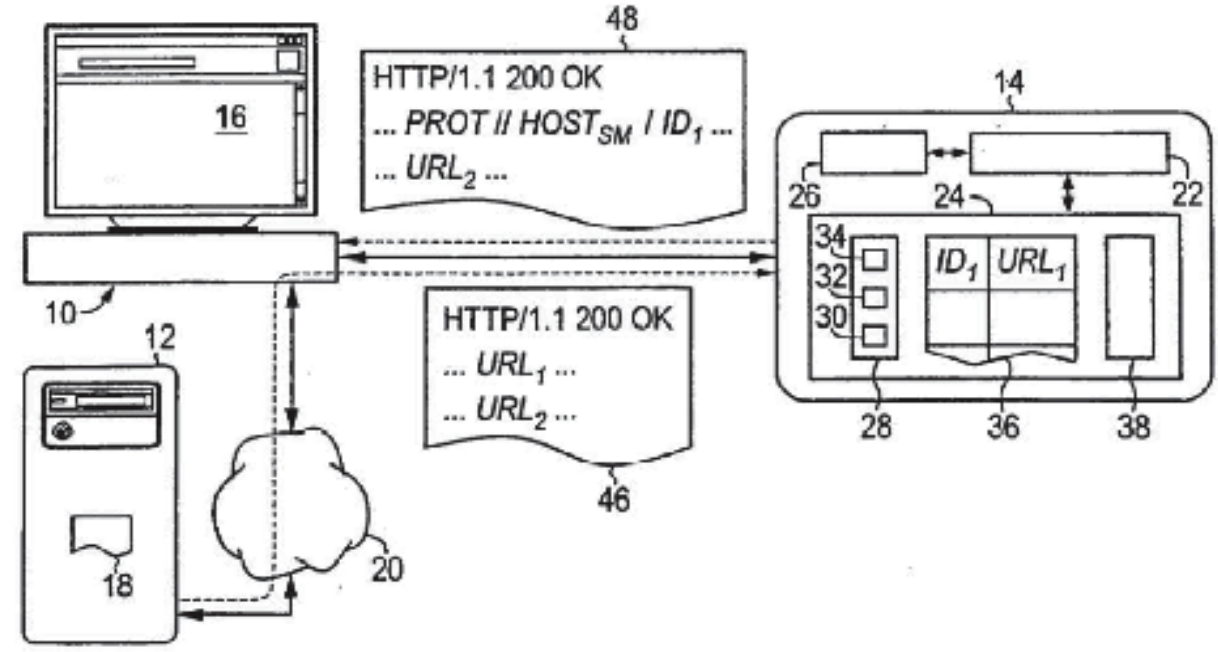
48
16
HTTP/1.1 200 OK
... PROT // HOST$_{SM}$ / ID$_1$ ...
... URL$_2$ ...
14
26
24
22
34
32
30
ID$_1$
URL$_1$
10
12
HTTP/1.1 200 OK
... URL$_1$ ...
... URL$_2$ ...
28
36
38
46
18
20

Fig. 2

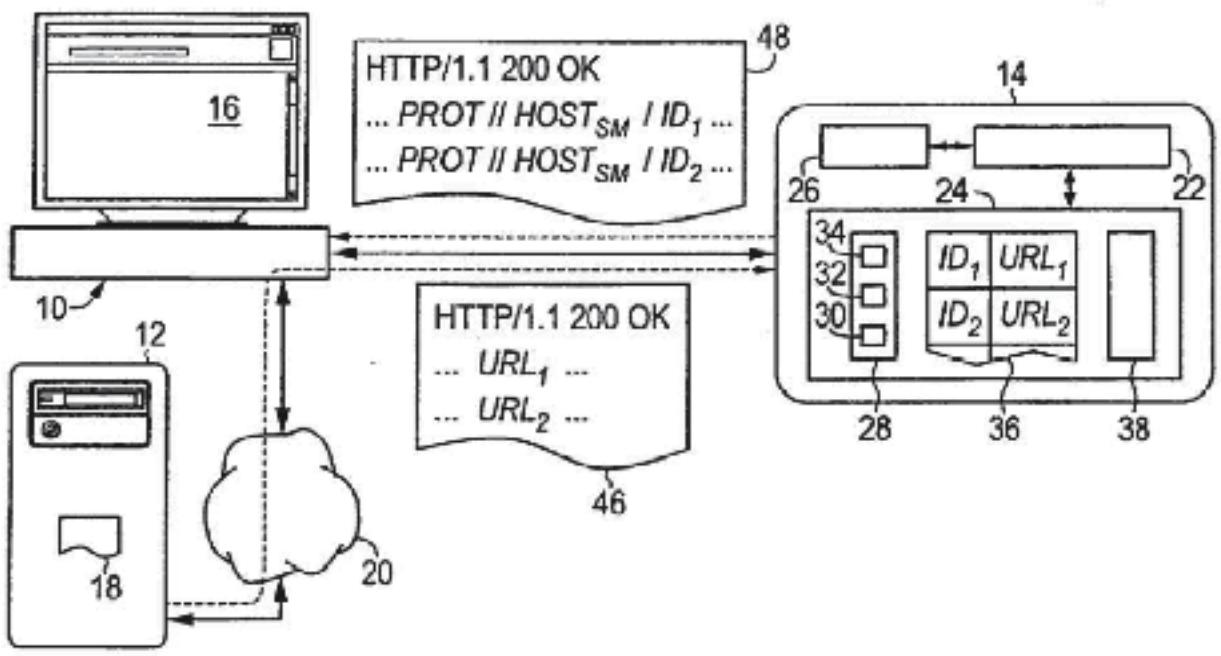
HTTP/1.1 200 OK
... PROT // HOST_SM / ID_1 ...
... PROT // HOST_SM / ID_2 ...
48
14
16
26
24
22
34
32
30
ID_1 URL_1
ID_2 URL_2
10
12
HTTP/1.1 200 OK
... URL_1 ...
... URL_2 ...
28
36
38
46
18
20

Fig. 3

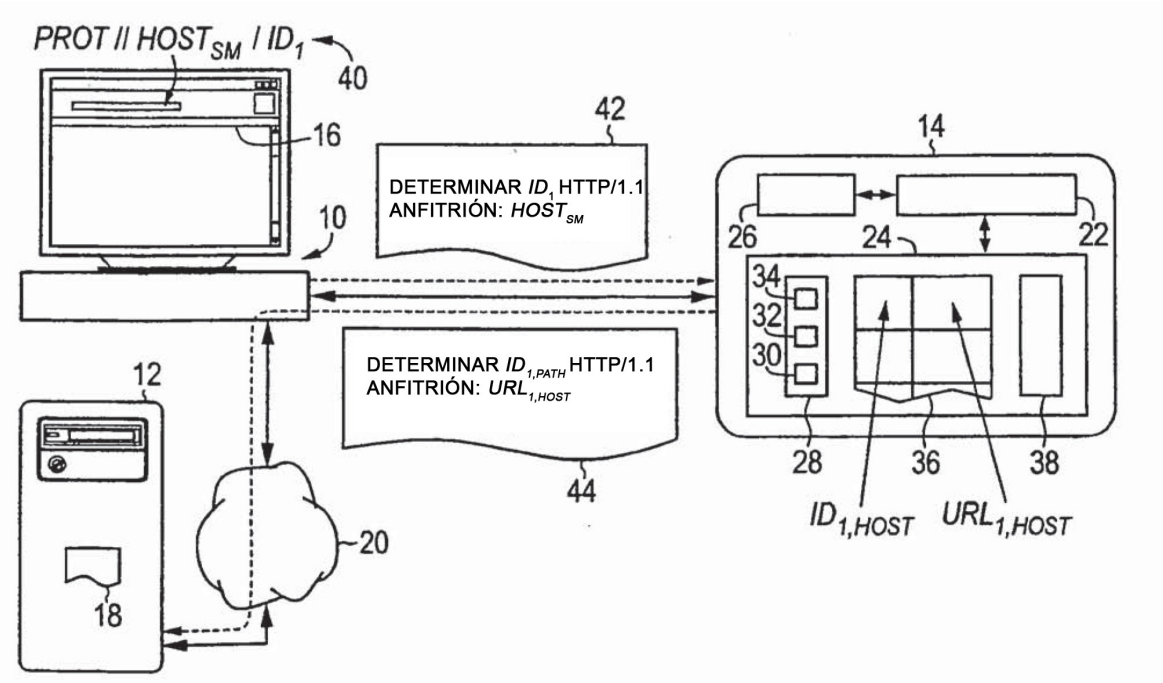
PROT // HOST_SM / ID_1
40
42
16
14
DETERMINAR ID_1 HTTP/1.1
ANFITRIÓN: HOST_SM
10
26
24
22
34
32
30
12
DETERMINAR ID_1,PATH HTTP/1.1
ANFITRIÓN: URL_1,HOST
28
36
38
ID_1,HOST
URL_1,HOST
44
20
18

Fig. 4

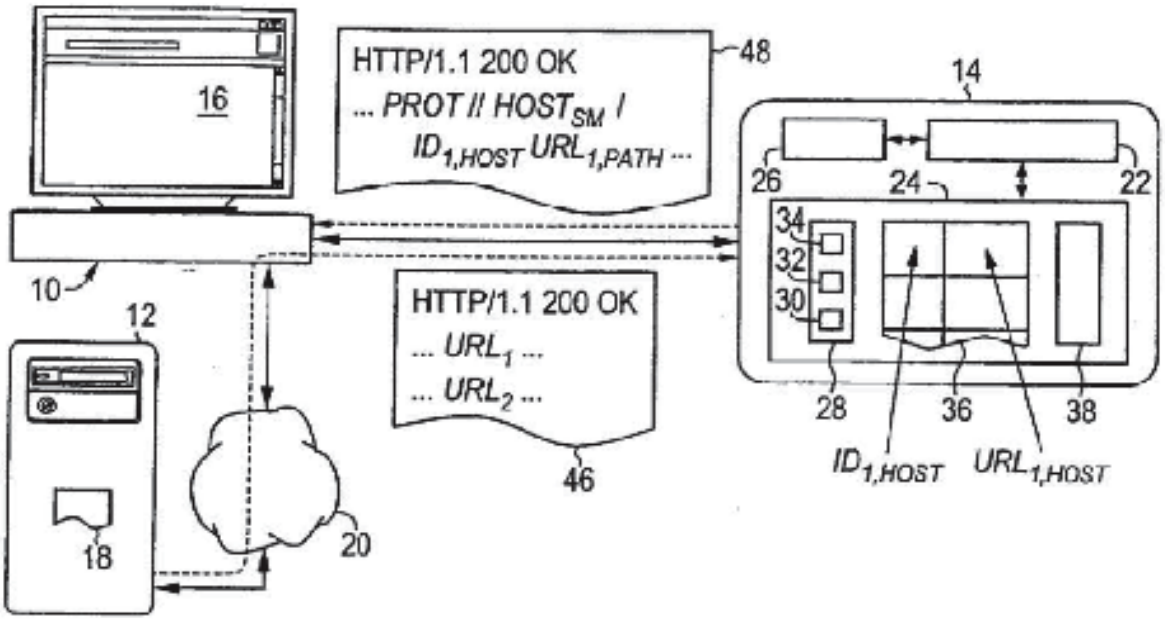
HTTP/1.1 200 OK
... PROT // HOST$_{SM}$ /
ID$_{1,HOST}$ URL$_{1,PATH}$ ...
48
14
16
26
24
22
34
32
30
10
12
HTTP/1.1 200 OK
... URL$_1$ ...
... URL$_2$ ...
46
28
36
38
ID$_{1,HOST}$
URL$_{1,HOST}$
18
20

Fig. 5

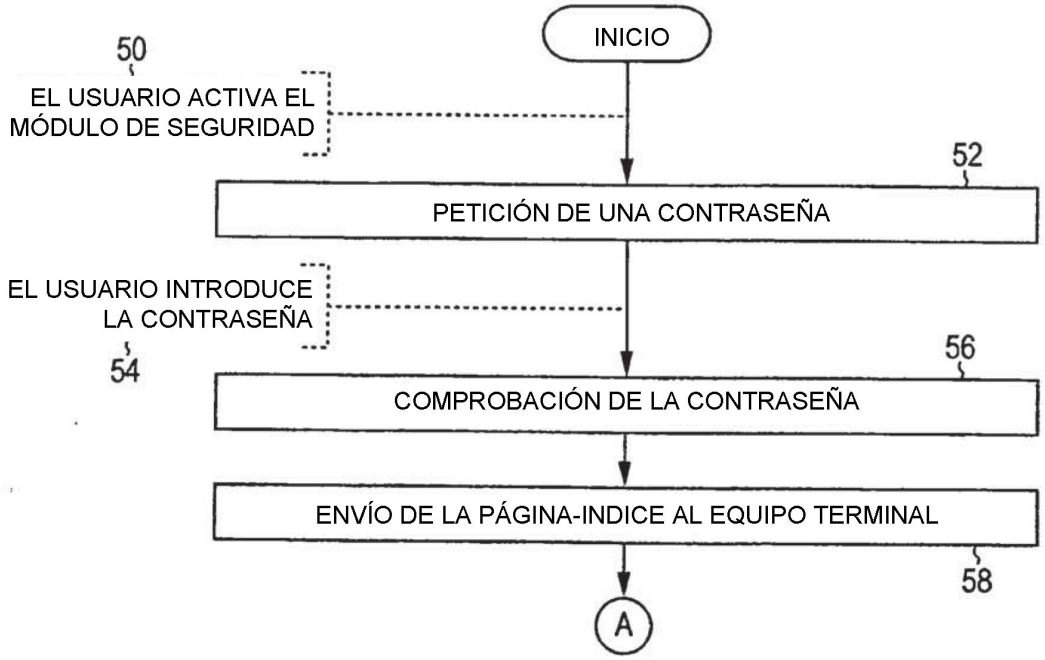
INICIO
50
EL USUARIO ACTIVA EL MÓDULO DE SEGURIDAD
52
PETICIÓN DE UNA CONTRASEÑA
EL USUARIO INTRODUCE LA CONTRASEÑA
54
56
COMPROBACIÓN DE LA CONTRASEÑA
ENVÍO DE LA PÁGINA-INDICE AL EQUIPO TERMINAL
58
A

Fig. 6

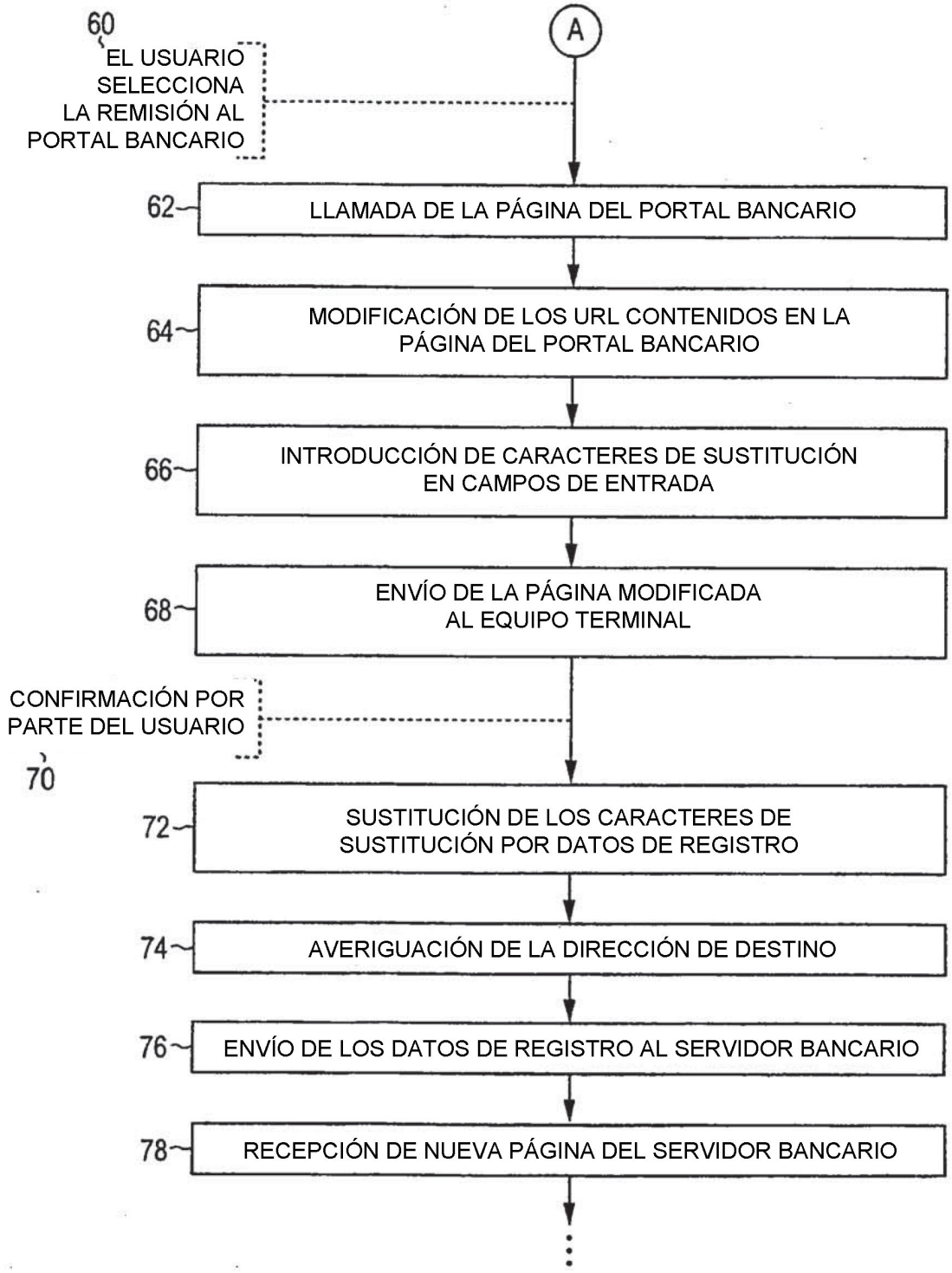
60
EL USUARIO SELECCIONA LA REMISIÓN AL PORTAL BANCARIO
A
62
LLAMADA DE LA PÁGINA DEL PORTAL BANCARIO
64
MODIFICACIÓN DE LOS URL CONTENIDOS EN LA PÁGINA DEL PORTAL BANCARIO
66
INTRODUCCIÓN DE CARACTERES DE SUSTITUCIÓN EN CAMPOS DE ENTRADA
68
ENVÍO DE LA PÁGINA MODIFICADA AL EQUIPO TERMINAL
CONFIRMACIÓN POR PARTE DEL USUARIO
70
72
SUSTITUCIÓN DE LOS CARACTERES DE SUSTITUCIÓN POR DATOS DE REGISTRO
74
AVERIGUACIÓN DE LA DIRECCIÓN DE DESTINO
76
ENVÍO DE LOS DATOS DE REGISTRO AL SERVIDOR BANCARIO
78
RECEPCIÓN DE NUEVA PÁGINA DEL SERVIDOR BANCARIO

Fig. 7

REFERENCIAS CITADAS EN LA DESCRIPCIÓN

La lista de referencias citada por el solicitante lo es solamente para utilidad del lector, no formando parte de los documentos de patente europeos. Aún cuando las referencias han sido cuidadosamente recopiladas, no pueden excluirse errores u omisiones y la OEP rechaza toda responsabilidad a este respecto.

Documentos de patente citado en la descripción

- WO 2006029758 A1 **[0002] [0003]**
- EP 1021020 B1 **[0005]**
- WO 9956210 A1 **[0006]**
- JP 2002312325 A **[0007]**

Bibliografía de patentes citada en la descripción

- **STEPHAN SPITZ ; WALTER HINZ.** Internet-Sicherheit mit Security-Token. *D·A·CH Mobility,* Oktober 2006 **[0004]**
- **R. FIELDING et al.** Hypertext Transfer Protocol - HTTP/1.1 **[0008]**
- **P. SRISURESH et al.** *Traditional IP Network Adress Translator (Traditional NAP,* Januar 2001, http:// tools.ietf.org/html/rfc3022 **[0009]**