(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 631 578**

(51) Int. CI.:

H04W 12/04 (2009.01)
H04W 12/06 (2009.01)
H04L 9/32 (2006.01)
H04L 29/06 (2006.01)
H04L 29/08 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **06.07.2011 PCT/SE2011/050916**

(87) Fecha y número de publicación internacional: **04.10.2012 WO12134369**

(96) Fecha de presentación y número de la solicitud europea: **06.07.2011 E 11862337 (0)**

(97) Fecha y número de publicación de la concesión europea: **19.04.2017 EP 2695410**

(54) Título: **Métodos y aparatos para evitar daños en ataques de red**

(30) Prioridad:

01.04.2011 US 201161470709 P

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
01.09.2017

(73) Titular/es:

TELEFONAKTIEBOLAGET LM ERICSSON (PUBL) (100.0%)
164 83 Stockholm, SE

(72) Inventor/es:

OHLSSON, OSCAR;
LEHTOVIRTA, VESA;
MATTSSON, JOHN y
NORRMAN, KARL

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Métodos y aparatos para evitar daños en ataques de red

Campo Técnico

La presente descripción se refiere generalmente a métodos y aparatos para permitir la comunicación segura entre un terminal de cliente y un servidor web, evitando daños que puedan ocurrir de otro modo cuando una clave de seguridad es robada del cliente.

Antecedentes

La denominada "Arquitectura Genérica de Arranque de un Sistema Operativo" (GBA) es una tecnología estandarizada en el Proyecto de Asociación de Tercera Generación (3GPP), que permite a un cliente en una red de comunicaciones establecer una clave secreta compartida, por ejemplo indicada como “Ks_NAF”, con una Función de Aplicación de Red (NAF) ubicada en un servidor web conectado a la red.

Un uso típico de la GBA es para la autentificación de clientes. Como muchas aplicaciones existentes actualmente están basadas en la web, es decir disponibles sobre la Internet pública o una intranet, la autentificación de cliente es particularmente interesante cuando es realizada desde un navegador utilizado por el cliente. Un escenario típico es aquel en el que el cliente es un terminal de comunicación equipado con un navegador operado por un usuario. Para permitir a la GBA ser utilizada junto con el formulario de Lenguaje de Marcado de Hipertexto (HTML) basándose en la autentificación - que es un método de autentificación ampliamente utilizado en Internet -se ha propuesto añadir una interfaz de programa de aplicación (API) de JavaScript en el cliente para la GBA en las aplicaciones basadas en la web. La API de JavaScript es ejecutada a continuación por el navegador del cliente y puede considerarse como parte del mismo. Tal escenario es descrito por ejemplo en el documento de Nokia Corporation: "GBA usage with Web Browser", 3GPP TSG-SA3 (Security); S3- 110107; SA3#62, 17 de enero de 2011, XP 050636351, Ljublijana, Eslovenia.

La API de JavaScript puede ser utilizada en el cliente según las acciones siguientes 1:1-1:4.

Acción 1:1. El navegador en el cliente descarga una página de inicio de sesión procedente de un servidor web sobre "HTTPS", refiriéndose este último al Protocolo de Transferencia de Hipertexto (HTTP) sobre la seguridad de capa de transporte (TLS) o la Capa de Conexión Segura (SSL). La página de inicio de sesión es una página web que contiene un "formulario HTML" con campos para "nombre de usuario" y "contraseña".

Acción 1:2. La página web descargada contiene también una pieza de JavaScript, en lo sucesivo denominada “secuencia de comandos” para abreviar. La secuencia de comandos obtiene un parámetro denominado "Identidad de Transacción de Arranque de Sistema Operativo, B-TID" que identifica al cliente. La secuencia de comandos obtiene además una clave secreta compartida denominada "Ks_NAF", llamando a la API de JavaScript GBA proporcionada por el cliente.

(B-TID, KS_NAF) = window.document.getGBAKey().

La B-TID actúa como una forma de alias para una IMSI (Identidad de Abonado de Móvil Internacional) del cliente y la Ks_NAF está asociada con el servidor web y posiblemente con otros servidores web así como utilizando la misma NAF. La Ks_NAF es así una clave compartida con el servidor web, que es utilizada en la Acción 1:4 posterior. En la Acción1:2 actual, la API de JavaScript devuelve la Ks_NAF y la B-TID a la secuencia de comandos.

Acción 1:3. La secuencia de comandos rellena el formulario HTML con la B-TID obtenida como nombre de usuario y la Ks_NAF obtenida como contraseña. El formulario HTML resultante es a continuación enviado desde el cliente al servidor web.

Acción 1:4. El servidor web, es decir la NAF, autentifica al cliente validando el nombre de usuario y la contraseña. En más detalle, la NAF rescata la Ks_NAF utilizada por el cliente procedente de una Función de Servidor de Arranque de Sistema Operativo (BSF) asociada con el cliente, presentando la B-TID recibida desde el cliente en la Acción 1:2 a la BSF. La BSF deriva la Ks_NAF, desde una clave compartida con el cliente denominada la "clave maestra, Ks". Esto es así la misma clave secreta que el cliente obtuvo en la acción 1:2. La Ks_NAF es única para cada NAF y Ks.

Sin embargo, la propuesta existente para añadir una API de JavaScript en el cliente es vulnerable a los así llamados "ataques de inyección de secuencia de comandos". Encontrando caminos de inyección de secuencias de comandos maliciosos en páginas webs, un atacante puede obtener privilegios de acceso elevado, o acceso, al contenido sensible de la página, cookies, y una variedad de otra información que puede mantenerse por el navegador en nombre del usuario. Si una página web es susceptible a la inyección de secuencia de comandos de esta manera o no depende de su contenido y las atenuaciones de seguridad son implementadas por el propietario del servidor web, que efectivamente es el propietario de la página.

Se ha identificado un problema con la propuesta anterior en el que la clave secreta compartida Ks_NAF devuelta por la API de JavaScript al cliente en el procedimiento anterior tiene un alcance muy amplio de uso. Una clave "robada" por un atacante a través de la inyección de secuencia de comandos puede permanecer válida y útil a través de un tipo de dominio Sistema de Nombres de Dominio (DNS), por ejemplo "naf.com", y a través de diferentes sesiones HTTP con el servidor/servidores que utilizan esa NAF.

Para ilustrar esto, se considerará un ejemplo con un anfitrión "naf.com" que es un servidor web que tiene dos páginas web accesibles indicadas como "login.html" y "vulnerable.html". La primera página login.html ha sido meticulosa y completamente codificada y es en sí misma invulnerable a cualquier tipo de ataque. La segunda página vulnerable.html, por otro lado, contiene un defecto sutil que la hace efectivamente susceptible a la inyección de código. Dándose cuenta de esto, un atacante inyecta la siguiente secuencia de comandos "maliciosa " en la segunda página web:

```
<SCRIPT type=”text/javascript”>
(B-TID Ks_NAF) = window.document.getGBAKey();
… upload (B-TID, Ks_NAF) to attacker …
</SCRIPT>
```

El siguiente usuario que visita la segunda página web naf.com/vulnerable.html y obtiene la clave compartida según el procedimiento anterior, cargará inconscientemente su clave obtenida al atacante por medio de la secuencia de comandos maliciosa anterior inyectada en la segunda página web. Como resultado, el atacante puede a su vez utilizar esa clave para iniciar sesión en naf.com/login.html y ser autentificado correctamente.

En otro ejemplo, un documento HTML vulnerable descargado es procesado en un navegador de cliente. El documento contiene un comando que instruye al navegador para descargar y ejecutar una secuencia de comandos de un tercero que puede ser un anuncio, un contador de visitantes u otra aplicación aparentemente inofensiva. La secuencia de comandos del tercero sin embargo contiene también una línea que rescata la Ks_NAF y la B-TID del cliente, que es consiguientemente llevado a cabo cuando es ejecutada la secuencia de comandos. La secuencia de comandos añade también un enlace a la página web procesada que es un comando para cargar la Ks_NAF y la B-TID a un tercero "malvado". Por lo tanto, el tercero obtiene la Ks_NAF y la B-TID del cliente para un uso posterior ilícito.

Es así un problema que un atacante puede obtener información sensible de un cliente particular tal como una clave secreta compartida, y utilizar esa información para autentificación y verificación falsa en próximas sesiones con servidores web donde esa información es válida.

Sumario

Es un objeto de la invención abordar al menos algunos de los asuntos y problemas antes resumidos. Es posible conseguir estos objetos y otros utilizando métodos y aparatos como se ha definido en las reivindicaciones independientes adjuntas.

Según un aspecto, se ha proporcionado un método en un terminal de cliente para permitir la comunicación segura con un servidor web. En este método, el terminal de cliente obtiene una página web desde el servidor web en una sesión con el servidor web, determina un conjunto de parámetros de contexto, P1,…Pn que pertenecen a dicha sesión y/o página web, y crea una clave específica de contexto, Ks_NAF’, basándose en los parámetros de contexto. El terminal de cliente utiliza a continuación la clave específica de contexto Ks_NAF’ creada para la verificación del cliente en el servidor web si una clave específica de contexto determinada en el servidor web coincide con la clave específica de contexto creada por el terminal de cliente.

Según otro aspecto, un terminal de cliente es configurado para permitir la comunicación segura con un servidor web. El terminal de cliente comprende un módulo de recepción adaptado para obtener una página web desde el servidor web en una sesión con el servidor web, una clave que crea el módulo adaptado para determinar un conjunto de parámetros de contexto, P1,…Pn que pertenecen a dicha sesión y/o página web, y para crear una clave específica de contexto Ks_NAF’, basándose en los parámetros de contexto, y un módulo de envío adaptado para enviar una solicitud de inicio de sesión que indica la clave específica de contexto Ks_NAF’ creada para el servidor web. Por lo tanto, la verificación del cliente es permitida si una clave específica de contexto determinada en el servidor web coincide con la clave específica de contexto creada por el terminal de cliente.

En esta solución, la clave específica de contexto es hecha así específica de contexto para ser útil para la autentificación o verificación solamente en el contexto actual e inútil de manera efectiva, es decir inválida, para autentificación o verificación en otro contexto. Por lo tanto, se puede evitar el daño en el caso en que la clave específica de contexto es robada en un ataque de red o similar.

El método anterior y el terminal de cliente pueden ser configurados e implementados según diferentes realizaciones opcionales. En una realización posible, la clave específica de contexto Ks_NAF’ creada es indicada en una solicitud

de inicio de sesión para el servidor web. Los parámetros de contexto pueden incluir al menos uno de: un componente de trayecto de un Localizador Universal de Recursos, URL, de la página web, una identidad de sesión HTTP actual de la sesión, y una identidad de sesión de seguridad de capa de transporte, TLS / capa de conexión segura, SSL de la sesión.

En otra realización posible, la clave de contexto específica Ks_NAF' es creada aplicando una Función de Derivación de Clave, KDF, predefinida, a los parámetros de contexto, siendo conocida la Función de Derivación de Clave para el servidor web. En este caso, un parámetro agregado, S puede ser creado basándose en los parámetros de contexto P1,…Pn de una manera predeterminada, y la Función de Derivación de Clave puede ser aplicada utilizando como entrada el parámetro agregado S y una clave específica de aplicación, Ks_NAF, de tal manera que

$$Ks\_NAF' = KDF(Ks\_NAF, S(P1,\ldots Pn))$$

Además, la clave específica de aplicación Ks_NAF puede haber sido derivada a partir de una clave maestra, Ks, obtenida procedente de una Función de Servicio de Arranque de Sistema Operativo. Los parámetros de contexto pueden ser determinados en comunicación con el servidor web que tiene una Función de Aplicación de Red.

Según otro aspecto, se ha proporcionado un método en un servidor web para permitir la comunicación segura con un terminal de cliente. En este método, el servidor web envía una página web al terminal de cliente en una sesión con el terminal de cliente, y recibe una solicitud de inicio de sesión procedente del terminal de cliente que indica una clave específica de contexto, Ks_NAF', creada por el terminal de cliente. El servidor web determina a continuación un conjunto de parámetros de contexto P1,…Pn, que pertenecen a dicha sesión y/o página web, y determina una clave específica de contexto, Ks_NAF', basándose en los parámetros de contexto. El servidor web puede verificar a continuación el cliente si la clave específica de contexto determinada en el servidor web coincide con la clave específica de contexto recibida desde el terminal de cliente.

Según otro aspecto, un servidor web es configurado para permitir la comunicación segura con un terminal de cliente. El servidor web comprende un módulo de envío adaptado para enviar una página web al terminal de cliente en una sesión con el terminal de cliente, y un módulo de recepción adaptado para recibir una solicitud de inicio de sesión procedente del terminal de cliente que indica una clave específica de contexto, Ks_NAF', creada por el terminal de cliente. El servidor web comprende también un módulo de verificación adaptado para determinar un conjunto de parámetros de contexto, P1,…Pn, que pertenecen a dicha sesión y/o página web, determinar una clave específica de contexto, Ks_NAF', basándose en los parámetros de contexto, y verificar el cliente si la clave específica de contexto determinada en el servidor web coincide con la clave específica de contexto recibida desde el terminal de cliente.

El método y el servidor web anteriores pueden ser configurados e implementados según las diferentes realizaciones opcionales. En una realización posible, los parámetros de contexto incluyen al menos uno de: un componente de trayecto de un Localizador Universal de Recursos de la página web, una identidad de sesión HTTP actual de la sesión y una identidad de sesión TLS/SSL actual de la sesión. Como en el terminal de cliente anterior, la clave específica de contexto, Ks_NAF' puede ser creada aplicando una Función de Derivación de Clave, KDF, predefinida, para los parámetros de contexto, siendo así conocida la Función de Derivación de Clave para el terminal de cliente. En ese caso, un parámetro agregado, S, puede ser creado basándose en los parámetros de contexto P1,…Pn de una manera predeterminada, y la Función de Derivación de Clave puede ser aplicada utilizando como entrada el parámetro agregado S y una clave específica de aplicación, Ks_NAF, de tal manera que

$$Ks\_NAF' = KDF(Ks\_NAF, S(P1,\ldots Pn).$$

La clave específica de aplicación Ks_NAF puede obtenerse a partir de una Función de Servicio de Arranque de Sistema Operativo asociada con el terminal de cliente.

Otras características y beneficios posibles de esta solución se pondrán de manifiesto a partir de la descripción detallada siguiente.

Breve descripción de los dibujos

La invención será ahora descrita en más detalle por medio de realizaciones ejemplares y con referencia a los dibujos adjuntos, en los que:

La figura 1 es un diagrama de bloques que ilustra un escenario de comunicación para verificar un cliente, según algunas realizaciones posibles.

La figura 2 es un diagrama de flujo que ilustra un procedimiento en un terminal de cliente, según otras realizaciones posibles.

La figura 3 es un diagrama de flujo que ilustra un procedimiento en un servidor web, según otras realizaciones posibles.

La figura 4 es un diagrama de bloques que ilustra un terminal de cliente y un servidor web en más detalle, según otras realizaciones posibles.

La figura 5 es un diagrama que ilustra acciones en un ejemplo de cómo un cliente puede ser verificado en un servidor web en la práctica, según otras realizaciones posibles.

Descripción detallada

incluso si no se impide que ocurra un ataque de inyección de código como tal como se ha descrito anteriormente, se sugiere ahora una solución para limitar el daño resultante del ataque configurando una función tal como una API de JavaScript GBA en un terminal de cliente de una manera novedosa como sigue. El daño que puede ser causado cuando una clave secreta es expuesta cuando se está accediendo a una página web, por ejemplo de la manera antes descrita, se puede evitar asegurando que la clave secreta devuelta por la API de JavaScript en el cliente es específica de contexto e inútil de manera efectiva, es decir inválida, para autentificación o verificación en otro contexto. Esto puede lograrse vinculando la clave anterior Ks_NAF a una o más de: la página web actual, la sesión HTTP actual, la sesión TLS/SSL actual, o cualquier combinación de las mismas. Por lo tanto, se obtiene una clave específica de contexto que puede ser utilizada para la autentificación del cliente en el contexto actual solamente pero no en otros contextos o sesiones.

Para construir la clave específica de contexto, el terminal de cliente puede realizar el siguiente procedimiento con las acciones 2:1-2:4 con el fin de comunicar con un servidor web que tiene una NAF:

Acción 2:1. En algún punto, el terminal de cliente obtiene una clave maestra "Ks" en una comunicación de arranque de sistema operativo con una BSF.

Acción 2:2. El terminal de cliente deriva una clave NAF específica de aplicación "Ks_NAF" a partir de la clave maestra Ks.

Acción 2:3. El terminal de cliente contacta con la NAF en el servidor web y determina un conjunto de n parámetros de contexto P1,…Pn utilizando un procedimiento predefinido. Cualquier número n de parámetros de contexto puede ser utilizado para esta solución, incluyendo n=1.

Acción 2:4. El terminal de cliente crea una clave específica de contexto, indicada aquí como Ks_NAF', aplicando una "Función de Derivación de Clave" KDF predefinida con la Ks_NAF y los parámetros de contexto P1,…Pn como entrada. En una realización posible, esta operación puede ser realizada de tal manera que

$$Ks\_NAF' = KDF(Ks\_NAF, S(P1,\ldots Pn).$$

donde S(P1,…Pn) es un parámetro creado basándose en los n parámetros de contexto P1,…,Pn de una manera predeterminada. Por ejemplo, el parámetro S puede ser una cadena de octetos que contiene una concatenación de los parámetros de contexto P1,…Pn. En esta descripción, el término "Función de Derivación de Clave" es utilizado generalmente para representar cualquier función que crea la clave específica de contexto con al menos los parámetros de contexto como entrada de manera que la clave específica de contexto está limitada efectivamente a los parámetros de contexto. Por ejemplo, la Función de Derivación de Clave KDF puede ser una agregación de varias funciones de tal manera que la salida de una función es utilizada como entrada para otra función, y así sucesivamente.

Por ejemplo, las acciones 2:1 y 2:2 pueden ser realizadas de la misma manera que en un procedimiento GBA ordinario mientras las acciones 2:3 y 2:4 son realizadas para obtener la clave específica de contexto. El terminal de cliente puede utilizar a continuación la clave específica de contexto para la autentificación correcta hacia el servidor web. En este proceso, el servidor web puede determinar la clave específica de contexto de la misma manera, es decir utilizando los mismos parámetros de contexto P1,…Pn y la KDF y verificar así la clave cuando es recibida procedente del terminal de cliente. El terminal de cliente utiliza la clave específica de contexto Ks_NAF' creada para la verificación indicándoselo al servidor web de una manera adecuada. Por ejemplo, la Ks_NAF' como tal, o una representación de la misma tal como un identificador de autenticación asociado con la clave específica de contexto Ks_NAF', puede ser enviada al servidor web en una solicitud de inicio de sesión que inicia el procedimiento de autentificación en el servidor web. En cualquier caso, la solicitud de inicio de sesión indica básicamente la clave específica de contexto Ks_NAF' creada.

En la figura 1, se ha ilustrado una arquitectura para utilizar la GBA en un navegador de un cliente que puede ser empleada para la solución descrita aquí. En esta figura, un terminal de cliente 100 comprende un Módulo de Identidad de Abonado, SIM, 100a y un cliente 100b GBA que puede comunicar con una BSF 102 sobre una interfaz Ub y utilizar una unidad de entrada/salida "I/O" 100c. La BSF 102 tiene acceso a la información de cliente en un Servidor de Abonado Doméstico (HSS) o una entidad de Registro Doméstico de Ubicación (HLR) 104 sobre una interfaz Zh. La SIM 100a contiene información de identidad del cliente.

El terminal de cliente 100 comprende también un navegador 100d que puede comunicar con una NAF de un servidor web 106 sobre una interfaz Ua y utilizar la unidad de entrada/salida I/O 100c. Como es indicado en la figura, tanto el terminal de clientes 100 como la NAF en el servidor web 106 son configurados con la KDF antes descrita y el

parámetro S(P1,…Pn) que son conocidos así por ambas partes que pueden crear y determinar la misma clave específica de contexto Ks_NAF' a partir de ellas. Además, la NAF en el servidor web 106 tiene acceso a la información de cliente desde la BSF 102 sobre una interfaz Zn. Dependiendo de la implementación, la clave específica de contexto Ks_NAF' puede ser creada en el terminal de cliente 100 por el navegador 100d o por un bloque funcional separado, no mostrado, que está conectado al mismo. En la siguiente descripción, el término "cliente" o "terminal de cliente" se puede referir a cualquier unidad funcional adecuada para realizar las acciones y funciones descritas en el terminal de cliente.

Se describirá a continuación un procedimiento ejecutado en un terminal de cliente para permitir la comunicación segura con un servidor web, con referencia al diagrama de flujo en la figura 3. Este procedimiento puede ser aplicado en el terminal de cliente 100 descrito para la figura 1 anterior. En una primera acción 200, el terminal de cliente establece una sesión con el servidor web, por ejemplo sobre la Internet pública o una intranet. En esta sesión, el terminal de cliente obtiene una página web desde el servidor como es mostrado en una siguiente acción 202.

El terminal de cliente a continuación, determina un conjunto de parámetros de contexto, P1,…Pn, que pertenecen a la sesión en curso y/o la página web obtenida, en otra acción 204. Por ejemplo, los parámetros de contexto pueden ser cualquier número de parámetros que incluyen al menos uno de: un componente de trayecto de un Localizador Universal de Recursos (URL) de la página web, una identidad de sesión HTTP actual de la sesión, una identidad de sesión TLS/SSL actual de la sesión, que será descrito en más detalle a continuación.

En una siguiente acción 206, el terminal de cliente crea una clave específica de contexto, Ks_NAF', basándose en los parámetros de contexto determinados anteriormente. Por ejemplo, la Ks_NAF' puede ser creada aplicando una Función de Derivación de Clave, KDF, predefinida, a los parámetros de contexto, siendo conocida la Función de Derivación de Clave para el servidor web también. Como se ha mencionado antes, la KDF puede ser una sola función o agregada desde varias funciones. Además, un parámetro agregado, S(P1,…Pn), puede ser creado basándose en los parámetros de contexto P1,…Pn de una manera predeterminada, y la Función de Derivación de Clave puede ser aplicada utilizando el parámetro agregado S(P1,…Pn) y una clave específica de aplicación, Ks_NAF como entrada, de tal manera que

$$Ks\_NAF' = KDF(Ks\_NAF, S(P1,\ldots Pn)).$$

Además, el terminal de cliente puede haber derivado la clave específica de aplicación Ks_NAF desde una clave maestra, Ks, obtenida desde una BSF, como la BSF 102 mostrada en la figura 1. El terminal de cliente puede determinar también los parámetros de contexto en comunicación con una NAF en el servidor web por ejemplo para obtener la información de sesión para ser utilizada como parámetros de contexto.

Una acción 208 final ilustra que el terminal de cliente utiliza la clave específica de contexto Ks_NAF' creada para la verificación del cliente en el servidor web, por ejemplo enviando al servidor web una solicitud de inicio de sesión que indica la clave específica de contexto Ks_NAF' creada incluyendo la clave específica de contexto como tal o una representación de la misma. Por ello, el servidor web es capaz de verificar el cliente si una clave específica de contexto determinada en el servidor web coincide con la clave específica de contexto indicada por el terminal de cliente, que será descrito en la siguiente figura.

Un procedimiento ejecutado en un servidor web para permitir la comunicación segura con un terminal de cliente, será ahora descrito con referencia al diagrama de flujo en la figura 3. Este procedimiento puede ser aplicado en el servidor web 106 descrito para la figura 1 anterior, y/o en combinación con el procedimiento de la figura 2. En una primera acción 300, el servidor web establece una sesión con el terminal de cliente, que corresponde a la acción 200 anterior. En una siguiente acción 302, el servidor envía una página web al terminal de cliente durante la sesión, que corresponde con la acción 202 anterior.

En algún punto en la sesión, el servidor web recibe una solicitud de inicio de sesión desde el terminal de cliente, en una siguiente acción 304, donde una clave específica de contexto, Ks_NAF', creada por el terminal de cliente, es indicada en la solicitud de inicio de sesión. Esta acción corresponde así a la acción 208 anterior. El servidor web determina a continuación un conjunto de parámetros de contexto, P1,…Pn, que pertenecen a la sesión actual y/o a la página web anterior, en una acción 306, y determina una clave específica de contexto, Ks_NAF', basándose en los parámetros de contexto determinados anteriormente, en otra acción 308.

Las acciones 306 y 308 son realizadas de la misma manera que las acciones 204 y 206 realizadas por el terminal descritas para la figura 2 anterior, y la clave específica de contexto resultante debería por tanto ser la misma tanto en el terminal como en el servidor. En una siguiente acción 310, el servidor web comprueba si la clave recibida en la solicitud de inicio de sesión de la acción 304 coincide con la clave determinada en la acción 306. Si no es así, el terminal de cliente se considera que no es fiable y la solicitud de inicio de sesión es rechazada en una acción 312. Por otro lado, si las claves coinciden, es decir la clave Ks_NAF' determinada por el servidor web corresponde con la clave Ks_NAF' creada por el terminal de cliente, el cliente es fiable y puede ser verificado en una acción 314.

La figura 4 es un diagrama de bloques que ilustra un terminal de cliente 400 y servidor web 402 configurado para emplear la solución antes descrita según otro ejemplo ilustrativo pero no limitativo. El servidor web 402 tiene también una NAF implementada. El terminal de cliente 400 y el servidor web 402 pueden ser configurados para actuar como sigue.

Un módulo de recepción 400a en el terminal de cliente 400 está adaptado para obtener o descargar una página web desde un módulo de envío 402a en el servidor 402 durante una sesión. La página web puede ser una página de inicio de sesión. Un módulo de creación de clave 400b en el terminal de cliente 400 está adaptado para determinar un conjunto de n parámetros de contexto P1,…Pn que pertenece a la sesión actual y/o a la página web obtenida. Los parámetros de contexto pueden ser determinados utilizando un procedimiento predefinido configurado en el cliente.

El módulo de creación de clave 400b está adaptado además para crear una clave específica de contexto Ks_NAF' aplicando una KDF, y opcionalmente un parámetro agregado S(P1,…Pn), por ejemplo como se ha descrito antes, al menos para dicho conjunto de parámetros de contexto. En más detalle, el parámetro agregado S es creado basándose en los parámetros de contexto P1,…Pn de una manera predeterminada, y la Función de Derivación de Clave es aplicada utilizando como entrada el parámetro agregado S(P1,…Pn) y opcionalmente también una clave específica de aplicación, Ks_NAF, como se ha descrito también anteriormente para la figura 2.

Un módulo de envío 400c en el terminal de cliente 400 está adaptado para enviar una solicitud de inicio de sesión, indicando la clave específica de contexto creada, a un módulo de recepción 402b en el servidor 402. Como se ha mencionado antes, la clave específica de contexto puede ser indicada en la solicitud de inicio de sesión incluyendo la Ks_NAF' como tal o una representación de la misma, tal como un identificador de autentificación asociado con la clave específica de contexto, en la solicitud de inicio de sesión.

En respuesta a ello, un módulo de verificación 402c en el servidor 402 está adaptado para determinar una clave específica de contexto para el cliente, que se puede hacer determinando los parámetros de contexto P1,…Pn y aplicando la KDF, y opcionalmente el parámetro agregado S(P1,…Pn) al menos a los parámetros de contexto determinados. El módulo de verificación 402c es adaptado a continuación para verificar el cliente del terminal 400 si la clave específica de contexto en el servidor web 402 coincide con la clave específica de contexto creada por el terminal de cliente 400.

Debería observarse que la figura 4 ilustra simplemente distintos módulos o unidades funcionales en el servidor web 402 y el terminal de cliente 400 en un sentido lógico, aunque la persona experta es libre de implementar estas funciones en la práctica utilizando medios de software y hardware adecuados. Así, este aspecto de la solución no está limitado generalmente a las estructuras mostradas del servidor web 402 y del terminal de cliente 400, mientras sus módulos funcionales 402a-c y 400a-c pueden ser configurados para funcionar según las características descritas para cualquiera de las figuras 1-3 anteriores, donde sea apropiado.

Los módulos funcionales 402a-c y 400a-c descritos antes pueden ser implementados en el servidor web 402 y el terminal de cliente 400, respectivamente, como módulos de programa de un programa informático respectivo que comprende medios de código que, cuando se ejecutan por un procesador "P" en cada uno del servidor web 402 y del terminal de cliente 400 hace que realicen las acciones anteriormente descritas. Cada procesador P puede ser una sola Unidad de Procesamiento Central (CPU), o podría comprender dos o más unidades de procesamiento. Por ejemplo, el procesador P puede incluir microprocesadores de propósito general, procesadores de conjunto de instrucciones y/o conjuntos de chips relacionados y/o microprocesadores de propósito especial tal como Circuitos Integrados Específicos de Aplicación (ASIC). El procesador P puede comprender un almacenamiento para propósitos de almacenamiento en memoria caché.

Cada programa informático puede ser llevado por un producto de programa informático bien en el servidor web 402 o bien en el terminal de cliente 400, respectivamente, en la forma de una memoria "M" conectada a cada procesador P. El producto de programa informático o memoria M comprende un medio legible por ordenador sobre el cual es almacenado el programa informático. Por ejemplo, la memoria M puede ser una memoria flash, una Memoria de Acceso Aleatorio (RAM), una Memoria de Sólo Lectura (ROM) o una ROM Programable Eléctricamente que se Puede Volver a Grabar (EEPROM), y los módulos del programa podrían ser distribuidos en realizaciones alternativas sobre diferentes productos de programa informático en la forma de memorias dentro del servidor web 402 y del terminal de cliente 400.

Serán ahora descritos en más detalle algunos ejemplos posibles pero no limitativos para implementar la solución para un cliente que utiliza un navegador en un terminal de comunicación.

La clave específica de contexto puede ser creada como sigue, según algunas realizaciones posibles. En vez de devolver la clave secreta compartida Ks_NAF a la secuencia de comandos, por ejemplo como la acción 1:2 anterior, la API de JavaScript GBA en el navegador devuelve una clave específica de contexto Ks_NAF' que está limitada a un conjunto predefinido de n parámetros de contexto "P1,…Pn". Los valores de esos parámetros de contexto pueden ser determinados por el servidor web para la sesión actual y/o para la página web. A menos que los parámetros de

contexto sean idénticos cuando la clave específica de contexto es creada y cuando es utilizada, la clave Ks_NAF' será rechazada por el servidor web como inválida si es utilizada por un cliente en otro contexto.

El vínculo de la clave específica de contexto Ks_NAF' con los parámetros de contexto puede ser hecho utilizando una función de derivación de clave KDF adicional para calcular/crear Ks_NAF' como:

$$Ks\_NAF' = KDF(Ks\_NAF, S(P1,\ldots Pn))$$

donde S(P1,…Pn) es una cadena de octetos construida a partir de los n parámetros de contexto, que puede ser vista como un parámetro agregado.

El proceso de agregación S para construir la cadena de octetos S(P1,…Pn) a partir de los n parámetros de contexto es predefinido así y ha sido configurado en el terminal del cliente. Como se ha mencionado antes, algunos ejemplos de tales parámetros de contexto P1,…Pn que pueden ser utilizados para crear la clave específica de contexto en esta solución incluyen, pero no están limitados a, los siguientes:

- Un componente del trayecto de un URL (Localizador Universal de Recursos) de la página web.
- Una cookie de ID (Identidad) de sesión o alguna otra cookie basada en HTTP enviada desde el servidor web al terminal de cliente, que en esta descripción se denomina "identidad de sesión HTTP".
- Un único valor asociado con una sesión TLS/SSL establecida entre el servidor web y el terminal de cliente, sobre la cual es descargada la página web, por ejemplo el "ID de sesión" TLS/SSL, un "master_secret" ("maestra secreta"), un "Mensaje finalizado", o algún otro valor aleatorio generado como parte de un procedimiento de configuración de conexión TLS/SSL. En esta descripción, este valor se denomina como una "identidad de sesión TLS/SSL".

Es posible utilizar solamente uno o dos parámetros de contexto, es decir n=1 o n=2, respectivamente. Por ejemplo, es posible vincular la clave específica de contexto a la página de inicio de sesión y a la sesión TLS/SSL en curso utilizando el trayecto de URL de la página de inicio de sesión y el ID de Sesión TLS/SSL actualmente utilizado por el cliente.

La función de derivación de clave KDF y la construcción del parámetro agregado S(P1,…Pn) puede ser realizado como sigue, según otras realizaciones posibles. Aunque puede haber distintas opciones posibles para elegir una función de derivación de clave KDF y una construcción de S(P1,…Pn) para esta solución, reutilizar la KDF y el S(P1,…Pn) definidos en la GBA podría ofrecer algunos beneficios con respecto a la implementación.

A modo de ejemplo, el 3GPP especifica la KDF y la construcción del S(P1,…Pn) generalmente como:

$$KDF(key, S(P1,\ldots Pn)) = HMAC\text{-}SHA\text{-}256(key, S(P1,\ldots Pn))$$

En este caso, la construcción de la cadena de octetos S(P1,…Pn) a partir de los n parámetros de entrada es además especificada por el algoritmo siguiente:

$$S(P1,\ldots Pn) = FC \,||\, P1 \,||\, L1 \,||\, \ldots \,||\, Pn \,||\, Ln$$

donde

FC es un solo octeto utilizado para distinguir diferentes ejemplos del algoritmo,
P1,…Pn son las n codificaciones del parámetro de contexto,
L1,…L2 son las longitudes de las codificaciones de parámetros de contexto de entrada correspondientes P1,…Pn, y || indica concatenación

Debería observarse que la indexación de n parámetros de entrada utilizados antes difiere de la indexación de n+1 parámetros de entrada utilizados en el apéndice B.2 de referencia [1]. Las definiciones anteriores de KDF y S pueden ser utilizadas así para esta solución.

Algunos ejemplos de selección de parámetros de contexto para utilizar en esta solución serán ahora descritos, según otras realizaciones posibles. Como se ha mencionado previamente, uno de los problemas con la propuesta de 3GPP existente es que la clave secreta compartida devuelta por la API de JavaScript al cliente es típicamente válida a través de un dominio DNS completo, por ejemplo "naf.com". Esto puede ser evitado según esta solución utilizando una clave específica de contexto en su lugar, como se ha descrito anteriormente.

Se describirá ahora cómo una clave específica de contexto puede ser limitada a un componente del trayecto de una URL de página web, según otras realizaciones posibles. Utilizando el ejemplo anterior de nuevo donde un anfitrión "naf.com" tiene dos páginas web "login.html" y "vulnerable.html", una clave ordinaria obtenida desde la página web naf.com/vulnerable.html de una manera convencional puede ser utilizada después por un atacante para iniciar sesión mediante naf.com/login.html por medio de la secuencia de comandos maliciosa anteriormente descrita. Esta

amenaza puede ser mitigada vinculando una clave específica de contexto al componente del trayecto de la URL del servidor web, donde la clave específica de contexto es calculada a partir de un parámetro de contexto "Pi" que está basado en la URL de la página web:

Pi = abs_path

donde abs_path es especificado en HTTP 1.1.

Se describirá ahora cómo una clave específica de contexto puede ser limitada a una sesión HTTP actualmente utilizada por el terminal de cliente, según otras realizaciones posibles. Los servidores web utilizan típicamente los ID de sesión para gestionar el estado del cliente a través de las solicitudes HTTP. El ID de sesión es almacenado en el navegador y es incluido también en cada solicitud para páginas web desde el navegador- normalmente en la forma de una cookie. Así, la clave puede ser limitada a la sesión HTTP actual configurando un parámetro del contexto "Pi" como:

Pi = HTTP Session identity

Sin embargo, diferentes lenguajes de servidor web tienden a utilizar diferentes nombres para la cookie de ID de sesión. Ejemplos de nombres que utilizan algunos lenguajes de programación cuando nombran su cookie incluyen JESSIONID (JSP), PHPSESSID (PHP), y ASPSESSIONID (Microsoft ASP). Esto puede ser manejado incluyendo el nombre en la llamada función API. Otra alternativa posible es utilizar una cookie estandarizada que lleva una copia del ID de sesión o algún otro valor aleatorio.

Preferiblemente, la cookie puede ser marcada como "solamente HTTP" con el fin de impedir que cualquier secuencia de comandos del lado del cliente modifique o extraiga un valor. La seguridad puede ser mejorada además marcando la cookie como "no persistente", ya que una cookie no persistente es almacenada típicamente en la memoria del navegador y es borrada al salir.

Se describirá ahora cómo una clave específica de contexto puede ser limitada a la sesión TLS/SSL, según otras realizaciones posibles. Algunos sitios seguros conocidos, por ejemplo bancos etc., utilizan el ID de sesión TLS/SSL en vez de cookies para mantener el seguimiento de las sesiones con clientes. En este caso la sesión HTTP completa, que incluye cualesquiera solicitudes HTTP y respuestas entre el navegador y el servidor web, ocurren dentro de una sola sesión TLS/SSL.

Existen distintas razones por las que el seguimiento de sesión web mediante TLS/SSL es poco común, incluyendo por ejemplo:

- Algunos servidores web no soportan el seguimiento mediante TLS/SSL (la capa HTTP es en gran medida independiente de la capa TLS/SSL)
- Algunos sitios más grandes utilizan descargas TLS/SSL por razones de rendimiento. En este caso la gestión TLS/SSL es manejada por un servidor separado (con hardware de propósito especial) enfrente del servidor web.
- Los navegadores no siempre se comportan como se espera. Los navegadores a menudo limitan el número de conexiones abiertas (por ejemplo n conexiones máximas por pestaña y m conexiones en total) y manejan funciones de tiempo de espera de manera diferente.

Para sitios web que soportan el seguimiento mediante TLS/SSL, vincular la clave a un ID de sesión TLS/SSL o master_secret, o a algún otro parámetro específico de sesión, puede ser un modo útil de limitar los efectos del robo de la clave. Un parámetro de contexto Pi para utilizar en esta solución puede ser así configurado como:

Pi = TLS/SSL Session ID / master_secret / other session-specific parameter

Utilizar la master_secret u otro parámetro específico de sesión en vez del ID de Sesión podría ser incluso más seguro, pero extraer tales valores puede requerir cambios de la implementación TLS/SSL.

Un beneficio de utilizar una AP de JavaScript para la GBA en el cliente es que se puede conseguir simplicidad. Con tal API preparada, cualquier aplicación web podría comenzar utilizando la GBA con un mínimo de esfuerzo.

Entre otras cosas, esta novedosa solución puede proporcionar la ventaja de defenderse contra uno de los ataques más comunes en Internet, a saber la inyección antes descrita de la secuencia de comandos maliciosa. Devolviendo una clave que es específica de contexto en vez de una clave general/convencional, una clave específica de contexto robada será efectivamente inválida para el atacante en su otro/otros contextos.

La figura 5 es un diagrama de acciones tomadas por un terminal de cliente y un servidor web que ilustra otro ejemplo para emplear la solución antes descrita en un cliente y en un servidor web (NAF), respectivamente. Esta figura muestra así diferentes acciones posibles que implican al terminal de cliente 500 y al servidor web 502,

respectivamente, que pueden proporcionar una comunicación segura entre los dos nodos basándose en una verificación correcta y fiable del cliente en el servidor web 502. Este diagrama ilustra principalmente cómo la misma clave específica de contexto puede ser creada en ambos nodos como una base para la verificación del cliente de terminal 500.

Una primera acción 5:1 ilustra que el terminal de cliente 500 obtiene una página web que contiene una secuencia de comandos procedente del servidor web 502. El terminal 500 ejecuta a continuación la secuencia de comandos proporcionada en la página web, en una acción 5:2. Por lo tanto, la secuencia de comandos solicita una clave específica de contexto Ks-NAF' procedente del cliente en una acción 5:3, por ejemplo llamando a una API de GBA en el terminal 500. En respuesta a ello, el cliente obtiene una clave maestra Ks válida en una siguiente acción 5:4. En esta acción, el cliente puede comprobar primero si existe una clave maestra válida en el terminal que no ha expirado. Si tal clave maestra válida no está disponible, el cliente contactará con una BSF y realizará un procedimiento de arranque de sistema operativo para obtener una clave maestra válida desde la BSF.

En una siguiente acción 5:5 el cliente deriva una NAF de clave específica Ks_NAF a partir de la clave maestra Ks, y determina un conjunto de parámetros de contexto predefinidos en otra acción 5:6, básicamente de la manera descrita para los ejemplos anteriores. El cliente es capaz ahora de crear la clave específica de contexto Ks_NAF' solicitada basándose en la clave específica NAF determinada anteriormente Ks_NAF y los parámetros de contexto predefinidos, en una siguiente acción 5:7. Este proceso puede ser realizado como se ha descrito antes lo cual no será repetido aquí.

En una siguiente acción 5:8, el cliente devuelve la clave específica de contexto Ks_NAF' solicitada a la secuencia de comandos que a continuación envía por consiguiente la clave Ks_NAF' junto con el parámetro B-TID al servidor web 502 en otra acción 5:9. Como se ha mencionado antes, el parámetro B-TID identifica al cliente, el cual el servidor 502 utiliza para obtener la clave NAF específica Ks_NAF a partir de la BSF en otra acción 5:10.

El servidor 502 es capaz ahora de determinar los mismos parámetros de contexto predefinidos en otra acción 5:12 y de determinar la misma clave específica de contexto Ks_NAF' basándose en la clave NAF específica descrita anteriormente Ks_NAF y los parámetros de contexto predefinidos, en una siguiente acción 5:12. El servidor 502 realiza las acciones 5:11 y 5:12 básicamente de la misma manera que el cliente ha realizado las acciones 5:6 y 5:7, así para proporcionar la misma clave específica de contexto Ks_NAF' resultante. Finalmente, el servidor 502 puede verificar o autentificar al cliente en una acción 5:13, si la clave Ks_NAF' creada en la acción 5:12 coincide con la clave Ks_NAF' recibida en la acción 5:9.

La solución según cualquiera de las realizaciones y ejemplos descritos anteriormente permite a un navegador de cliente y a un servidor web establecer una clave secreta compartida, es decir la clave específica de contexto anterior, de una manera fácil y segura. La seguridad proviene parcialmente de que la clave está limitada al contexto de navegación particular lo que impide de manera efectiva a un atacante robar una clave en un contexto, por ejemplo, a través de la inyección de secuencia de comandos, y utilizarla en otro contexto. La solución puede ser utilizada así con la secuencia de comandos de lado del cliente y la GBA, como se ha descrito anteriormente.

Además, la clave específica de contexto puede ser utilizada también para otros propósitos de seguridad dentro del contexto y sesión actual con el terminal y el servidor web, tal como para encriptación de datos y mensajes comunicados. La seguridad es proporcionada ya que no se puede utilizar otra clave en esta sesión/contexto, tal como una clave robada procedente de otro contexto o sesión.

Aunque la invención ha sido descrita con referencia a realizaciones ejemplares específicas, la descripción está solamente destinada de manera general a ilustrar el concepto inventivo y no debería ser tomada como limitativa del alcance de la invención como es definida por las reivindicaciones adjuntas.

REIVINDICACIONES

1.- Un método en un terminal de cliente (100) para permitir una comunicación segura con un servidor web (106), comprendiendo el método:

- la obtención (202) de una página web que comprende un formulario HTML y un JavaScript procedentes del servidor web en una sesión con el servidor web,
- la determinación (204) de un conjunto de parámetros de contexto, P1,…Pn que pertenecen a dicha sesión y/o página web,
- la derivación de una clave específica de aplicación, Ks_NAF, basándose en una Clave maestra, Ks,
- la creación (206) de una clave específica de contexto, Ks_NAF', basándose en dichos parámetros de contexto y en la clave específica de aplicación, Ks_NAF,
- el envío de la clave específica de contexto, Ks_NAF', al JavaScript utilizando una API de JavaScript, de manera que (208) la clave específica de contexto Ks_NAF' creada puede ser utilizada para la verificación por el cliente en el servidor web de si una clave específica de contexto determinada en el servidor web coincide con la clave específica de contexto creada por el terminal de cliente.

2.- Un método según la reivindicación 1, en donde la clave específica de contexto Ks_NAF' creada es indicada en la solicitud de inicio de sesión en el servidor web.

3.- Un método según la reivindicación 1 o 2, en donde dichos parámetros de contexto incluyen al menos uno de: un componente de trayecto de un Localizador Universal de Recursos (URL) de dicha página web, una identidad de sesión HTTP actual de dicha sesión, y una identidad de sesión de Seguridad de Capa de Transporte (TLS)/Capa de Conexión Segura (SSL) actual de dicha sesión.

4.- Un método según una cualquiera de las reivindicaciones 1-3, en donde la clave específica de contexto Ks_NAF' es creada aplicando una Función de Derivación de Clave, KDF predefinida, a dichos parámetros de contexto y a la clave específica de aplicación, Ks_NAF, siendo conocida la Función de Derivación de Clave por el servidor web.

5.- Un método según la reivindicación 4, en donde un parámetro agregado, S, es creado basándose en los parámetros de contexto P1,…Pn de una manera predeterminada, y la Función de Derivación de Clave es aplicada utilizando como entradas dicho parámetro agregado S y la clave específica de aplicación, Ks_NAF, de tal manera que

$$Ks\_NAF' = KDF(Ks\_NAF, S(P1,\ldots Pn)).$$

6.- Un método según la reivindicación 5, que comprende además la obtención de la clave maestra, Ks, a partir de una Función de Servicio de Arranque de Sistema Operativo, BSF.

7.- Un método según una cualquiera de las reivindicaciones 1-6, en donde los parámetros de contexto son determinados en comunicación con el servidor web que tiene una Función de Aplicación de Red, NAF.

8.- Un terminal de cliente (100) configurado para permitir la comunicación segura con un servidor web (106) estando adaptado el terminal de cliente para llevar a cabo:

la obtención de una página web que comprende un formulario HTML y un JavaScript procedente del servidor web en una sesión con el servidor web,
la determinación de un conjunto de parámetros de contexto, P1,…Pn, que pertenecen a dicha sesión y/o página web,
la derivación de una clave específica de aplicación, Ks_NAF, basándose en una Clave maestra, Ks,
la creación de una clave específica de contexto, Ks_NAF', basándose en dichos parámetros de contexto y en la clave específica de aplicación, Ks_NAF,
el envío de la clave específica de contexto, Ks_NAF', al JavaScript utilizando una API de JavaScript, de manera que la clave específica de contexto Ks_NAF' creada puede ser utilizada para la verificación por el cliente en el servidor web de si una clave específica de contexto determinada en el servidor web coincide con la clave específica de contexto creada por el terminal de cliente.

9.- Un terminal de cliente según la reivindicación 8, adaptado además para crear la clave específica de contexto Ks_NAF' aplicando una Función de Derivación de Clave, KDF, predefinida, a dichos parámetros de contexto y a la clave específica de aplicación, Ks_NAF.

10.- Un terminal de cliente según la reivindicación 9, adaptado además para crear un parámetro agregado, S, basándose en los parámetros de contexto P1,…Pn de una manera predeterminada, y la Función de Derivación de Clave es aplicada utilizando como entradas dicho parámetro agregado S y la clave específica de aplicación, Ks_NAF, de tal manera que

Ks_NAF' = KDF(Ks_NAF, S(P1,…Pn)).

11.- Un terminal de cliente según una cualquiera de las reivindicaciones 8-10, adaptado además para determinar los parámetros de contexto en comunicación con el servidor web que tiene una Función de Aplicación de Red, NAF.

12.- Un método en un servidor web (106, 402) para permitir la comunicación segura con un terminal de cliente (100, 400) comprendiendo el método:

- el envío (302) de una página web que comprende un formulario HTML y un JavaScript al terminal de cliente en una sesión con el terminal de cliente,
- la recepción (304) de una solicitud de inicio de sesión procedente del terminal de cliente que indica una clave específica de contexto, Ks_NAF', creada por el terminal de cliente,
- la determinación (306) de un conjunto de parámetros de contexto, P1,…Pn, que pertenecen a dicha sesión y/o página web,
- la determinación (308) de una clave específica de contexto, Ks_NAF', basándose en dichos parámetros de contexto y en una clave específica de aplicación, Ks_NAF,
- la verificación (314) por el cliente de si la clave específica de contexto determinada en el servidor web coincide con la clave específica de contexto creada por el terminal de cliente.

13.- Un método según la reivindicación 12, en donde dichos parámetros de contexto incluyen al menos uno de: un componente de trayecto de un Localizador Universal de Recursos (URL) de dicha página web, una identidad de sesión HTTP actual de dicha sesión, y una identidad de sesión de seguridad de capa de transporte (TLS)/Capa de Conexión Segura (SSL) actual de dicha sesión.

14.- Un método según la reivindicación 12 o 13, en donde la clave específica de contexto Ks_NAF' es creada aplicando una Función de Derivación de Clave, KDF, predefinida, a dichos parámetros de contexto y a la clave específica de aplicación, Ks_NAF, siendo conocida la Función de Derivación de Clave por el terminal de cliente.

15.- Un método según la reivindicación 14, en donde un parámetro agregado, S, es creado basándose en los parámetros de contexto P1,…Pn de una manera predeterminada, y la Función de Derivación de Clave es aplicada utilizando como entradas dicho parámetro agregado S y la clave específica de aplicación, Ks_NAF, de tal manera que

Ks_NAF' = KDF(Ks_NAF, S(P1,…Pn))

16.- Un servidor web (402) configurado para permitir la comunicación segura con un terminal de cliente (400), estando adaptado el servidor web para llevar a cabo:

el envío de una página web que comprende un formulario HTML y un JavaScript al terminal de cliente en una sesión con el terminal de cliente,
la recepción de una solicitud de inicio de sesión procedente del terminal de cliente que indica una clave específica de contexto, Ks_NAF', creada por el terminal de cliente,
la determinación de un conjunto de parámetros de contexto, P1,…Pn, que pertenecen a dicha sesión y/o página web,
la determinación de una clave específica de contexto, Ks_NAF', basándose en dichos parámetros de contexto y en una clave específica de aplicación, Ks_NAF,

la verificación por el cliente de si la clave específica de contexto determinada en el servidor web coincide con la clave específica de contexto creada por el cliente.

17.- Un servidor web según la reivindicación 16, adaptado además para crear la clave específica de contexto Ks_NAF' aplicando una Función de Derivación de Clave, KDF, predefinida, a dichos parámetros de contexto y a la clave específica de aplicación, Ks_NAF.

18.- Un servidor web según la reivindicación 17, adaptado además para crear un parámetro agregado, S, basado en los parámetros de contexto P1,…Pn de una manera predeterminada, y aplicar la Función de Derivación de Clave utilizando como entradas dicho parámetro agregado S y la clave específica de aplicación, Ks_NAF, de tal manera que

Ks_NAF' = KDF(Ks_NAF, S(P1,…Pn)).

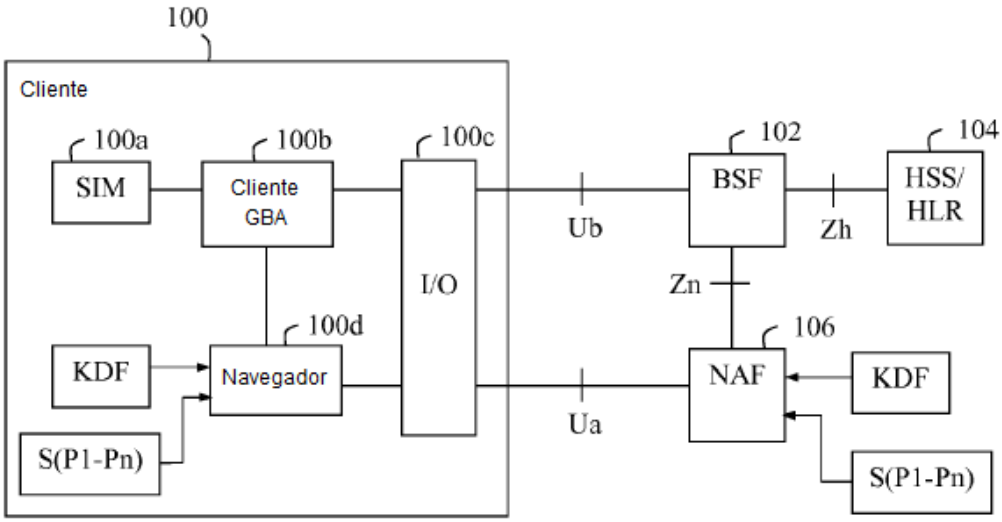
100
Cliente
100a
SIM
100b
Cliente GBA
100c
I/O
100d
KDF
Navegador
S(P1-Pn)
Ub
Ua
102
BSF
Zh
104
HSS/ HLR
Zn
106
NAF
KDF
S(P1-Pn)

Fig. 1

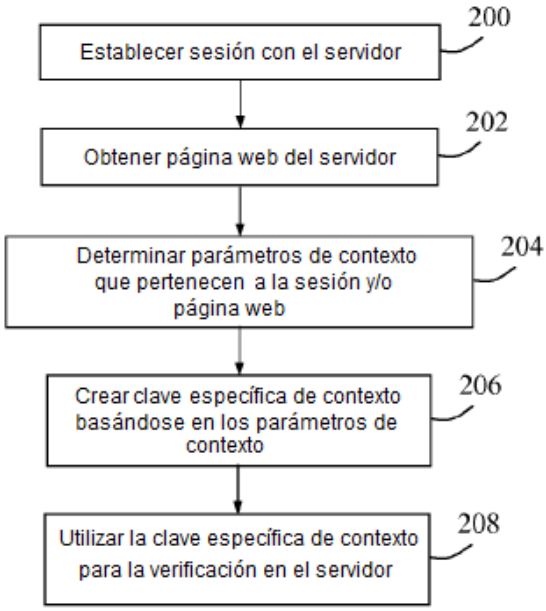
200
Establecer sesión con el servidor
202
Obtener página web del servidor
204
Determinar parámetros de contexto que pertenecen a la sesión y/o página web
206
Crear clave específica de contexto basándose en los parámetros de contexto
208
Utilizar la clave específica de contexto para la verificación en el servidor

Fig. 2

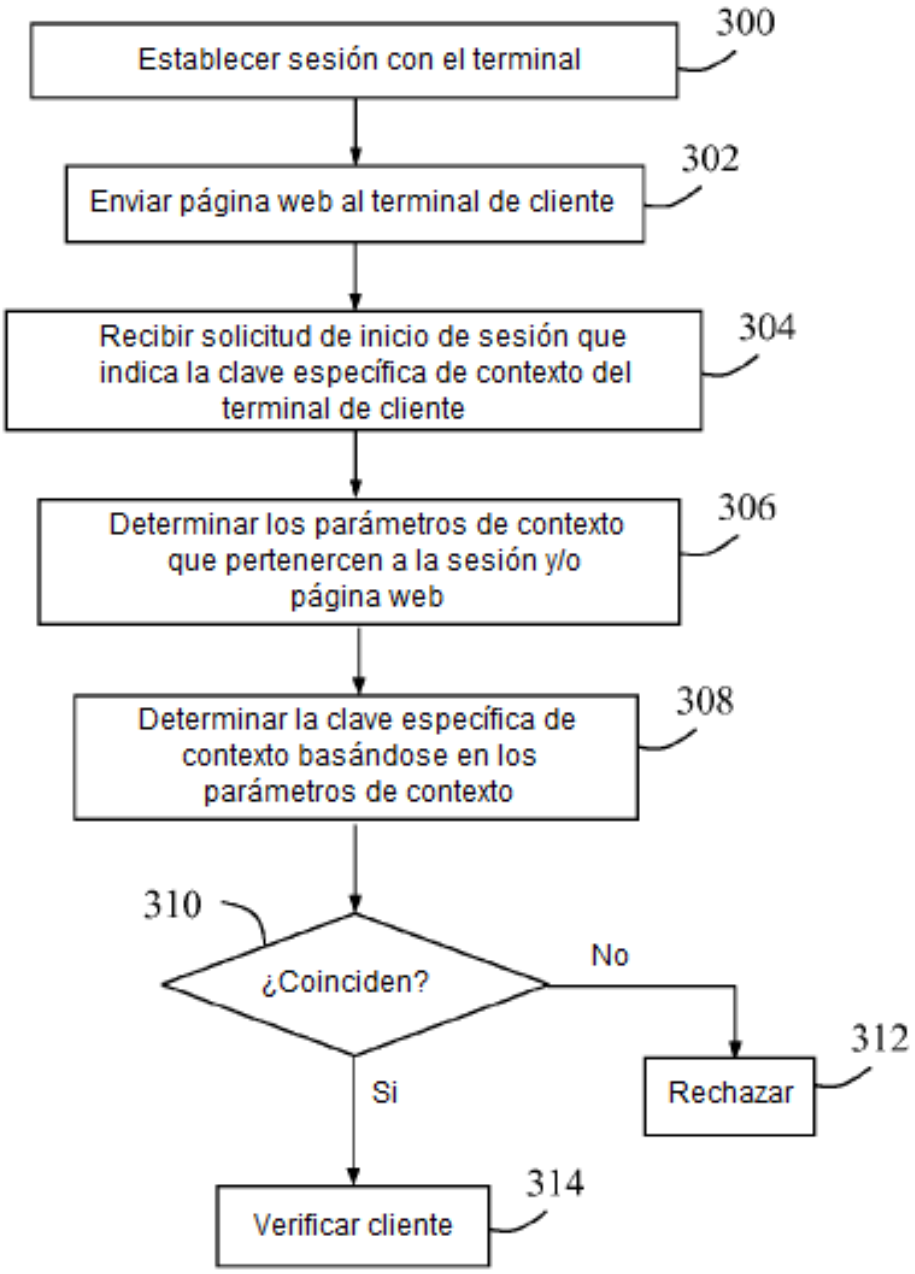
300
Establecer sesión con el terminal
302
Enviar página web al terminal de cliente
304
Recibir solicitud de inicio de sesión que indica la clave específica de contexto del terminal de cliente
306
Determinar los parámetros de contexto que pertenercen a la sesión y/o página web
308
Determinar la clave específica de contexto basándose en los parámetros de contexto
310
¿Coinciden?
No
312
Rechazar
Si
314
Verificar cliente

Fig. 3

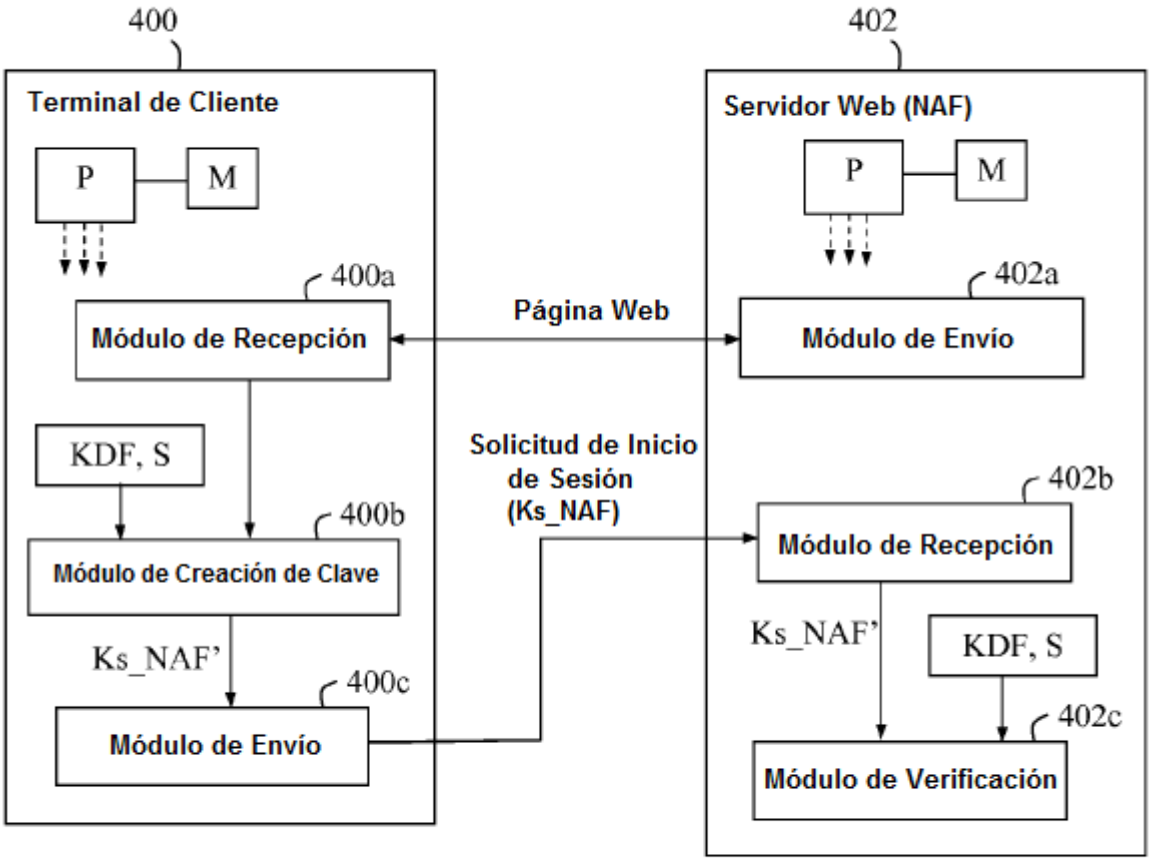
400
Terminal de Cliente
P
M
400a
Módulo de Recepción
KDF, S
400b
Módulo de Creación de Clave
Ks_NAF'
400c
Módulo de Envío
Página Web
Solicitud de Inicio de Sesión (Ks_NAF)
402
Servidor Web (NAF)
P
M
402a
Módulo de Envío
402b
Módulo de Recepción
Ks_NAF'
KDF, S
402c
Módulo de Verificación

Fig. 4

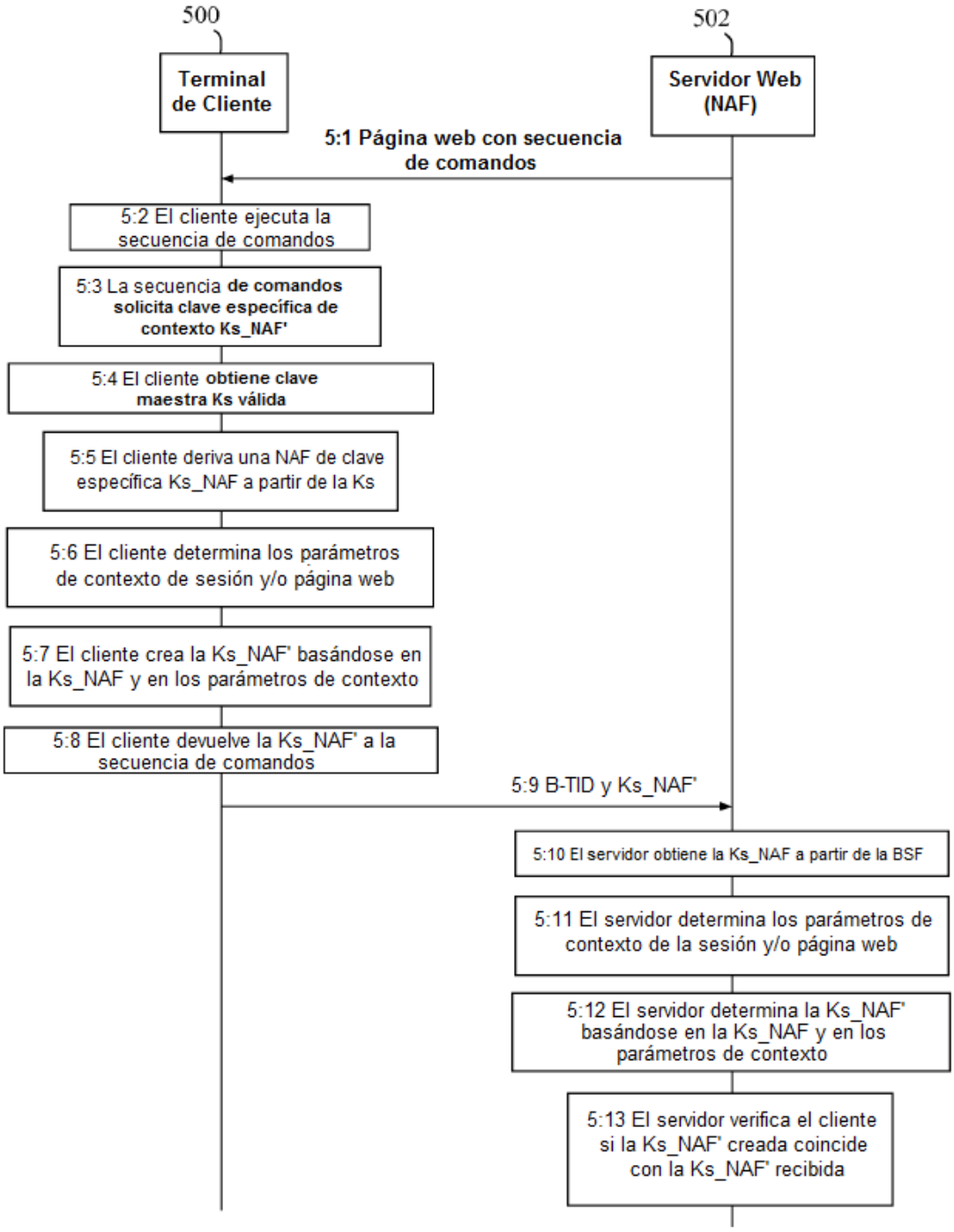
500
Terminal de Cliente
502
Servidor Web (NAF)
5:1 Página web con secuencia de comandos
5:2 El cliente ejecuta la secuencia de comandos
5:3 La secuencia de comandos solicita clave específica de contexto Ks_NAF'
5:4 El cliente obtiene clave maestra Ks válida
5:5 El cliente deriva una NAF de clave específica Ks_NAF a partir de la Ks
5:6 El cliente determina los parámetros de contexto de sesión y/o página web
5:7 El cliente crea la Ks_NAF' basándose en la Ks_NAF y en los parámetros de contexto
5:8 El cliente devuelve la Ks_NAF' a la secuencia de comandos
5:9 B-TID y Ks_NAF'
5:10 El servidor obtiene la Ks_NAF a partir de la BSF
5:11 El servidor determina los parámetros de contexto de la sesión y/o página web
5:12 El servidor determina la Ks_NAF' basándose en la Ks_NAF y en los parámetros de contexto
5:13 El servidor verifica el cliente si la Ks_NAF' creada coincide con la Ks_NAF' recibida

Fig. 5