(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 709 074**

(51) Int. CI.:

H04L 29/06 (2006.01)
G06F 21/62 (2013.01)
G06F 17/30 (2006.01)
H04L 29/08 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **15.03.2013 E 13001325 (3)**

(97) Fecha y número de publicación de la concesión europea: **31.10.2018 EP 2779016**

(54) Título: **Comparación de una lista de contactos automatizada con una mejora de la privacidad**

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
15.04.2019

(73) Titular/es:

ONECTIVE AG (100.0%)
Thurgauer Strasse 54
8050 Zürich, CH

(72) Inventor/es:

SINGLER, JOHANNES

(74) Agente/Representante:

LEHMANN NOVO, María Isabel

DESCRIPCIÓN

Comparación de una lista de contactos automatizada con una mejora de la privacidad

1. Antecedentes de la invención

1.1. Campo de la invención

La presente invención se encuentra en el campo de la tecnología de la información y está relacionada con un procedimiento y un sistema para comparar una primera pluralidad de conjuntos de datos privados, por ejemplo, una agenda de contactos privado que tiene típicamente unos pocos centenares de entradas de contactos, almacenadas en un dispositivo cliente de comunicación, con una segunda pluralidad de conjuntos de datos, por ejemplo, un gran repositorio de contactos que tiene miles o incluso millones de entradas de contacto, almacenadas en un sistema de comunicación basado en servidor.

1.2. Descripción y desventajas de la técnica anterior.

Los populares servicios de comunicación de la técnica anterior que se ejecutan en teléfonos inteligentes como WhatsApp utilizan el número de teléfono GSM del usuario como la identidad del usuario. Proporcionan servicios como mensajes de texto, llamadas a través de voz sobre IP y transferencia de archivos, a bajo coste, mediante la transferencia de datos a través de la conexión a Internet del teléfono. Para mostrar al usuario el subconjunto de sus contactos que también son miembros del mismo servicio de comunicación, comparan la agenda de contactos del usuario con el conjunto de miembros ya registrados del servicio de comunicación.

Hasta el momento, en la técnica anterior esto se hace enviando la lista de números de teléfono al proveedor de servicios sin cifrar y sin anonimizar, según se informa en la revista de "Stiftung Warentest", volumen 6/12, publicado el 24 de mayo de 2012, recibiendo en respuesta el subconjunto de contactos que ya son miembros. Por lo tanto, cualquier persona que pueda escuchar clandestinamente la comunicación entre el teléfono y el proveedor de servicios obtiene acceso a la agenda de contactos completa del usuario. Esto incluye al propio proveedor de servicios, por supuesto. Sin embargo, una agenda de contactos constituye información privada y es un secreto comercial para profesionales, que revela contactos valiosos y, posiblemente, incluso estrategias comerciales.

A fin de poder proporcionar el servicio solicitado, por ejemplo, para mostrar el estado en línea de los contactos, los contactos de los miembros del usuario deben ser revelados al proveedor del servicio. Esto está permitido porque esos contactos han aceptado los términos y servicios del servicio de comunicación. En cambio, los contactos que no participan en el servicio de comunicación no deben ser revelados públicamente, ni siquiera al proveedor del servicio de comunicación. De hecho, esto sería una violación del derecho de autodeterminación informativa ("Grundrecht auf informationelle Selbstbestimmung") del contacto, quien probablemente no ha aceptado que su información se comparta con dicho proveedor de servicios, como se informa en http://www.telemedicus.info/article/2222-LG-Berlin-Das-Facebook-Urteil-im-Detail.html.
según se publicó en Internet desde el 12 de marzo de 2012, o consulte la decisión publicada de Landgericht Berlin, LG Berlin, Urteil v. 06.03.2012, Az. 16 0 551 /10.

Una forma perfecta sensible con la privacidad sería cifrar asimétricamente la lista de números de teléfono, entrada por entrada a ambos lados con una clave aleatoria y única, y comparar las listas cifradas. Dado que el cifrado es biyectivo, es una bisección en el operador de intersección, por lo que el cliente puede transformar el resultado fácilmente hacia atrás. Sin embargo, cifrar todos los números de teléfono de un miembro en cada comparación es computacionalmente inviable, ya que podría haber millones, y se podrían producir múltiples consultas por segundo.

Otra forma sería enviar la lista completa de participantes del servicio al cliente y compararlos allí sin más interacción con el servidor. Sin embargo, esto también es inviable porque revelaría la información de contacto de todos los miembros del servicio a un usuario no fiable. Más aún, la larguísima lista de miembros tendría que enviarse al cliente, lo que daría lugar posiblemente a muchos megabytes de datos, lo cual no es factible, especialmente, para los clientes móviles.

1.3. Objetivos de la invención

El objetivo de la presente invención es proporcionar un procedimiento y un sistema eficaces para transmitir y comparar dichos datos privados, tal como se ha mencionado anteriormente, en el que los datos se procesan de manera que cualquier persona que escuche clandestinamente, incluido el proveedor de servicios central que realmente realiza la comparación, también tenga que dedicar un intensivo trabajo computacional si quiere saber acerca de los datos, y qué procedimiento no consume demasiada capacidad computacional en los dispositivos cliente y el servidor.

2. Resumen y ventajas de la invención.

Este objetivo de la invención se logra mediante las características establecidas en las reivindicaciones independientes adjuntas. Otras disposiciones y realizaciones ventajosas de la invención se exponen en las reivindicaciones dependientes respectivas. A continuación, se debe hacer referencia a las reivindicaciones anejas.

Según una característica básica de la presente invención, se describe un procedimiento ejecutado en un servidor para comparar una primera pluralidad de conjuntos de datos privados, como una agenda de contactos privada y almacenada en un dispositivo cliente de comunicación con una segunda pluralidad de conjuntos de datos, como un gran repositorio de contactos almacenado en dicho sistema de comunicación, que comprende las etapas de:

a) el ordenador servidor que calcula la longitud del valor hash s que representa el número de bits de un valor hash criptográfico de una parte única (por ejemplo, el número de teléfono) de cada uno de los conjuntos de datos privados que se transmitirán entre el cliente y el servidor, y que comunica s al cliente,

b) el sistema de servidor que recibe para cada una de dicha primera pluralidad de conjuntos de datos privados un valor hash reducido de dicho cliente, preferiblemente en un canal de transmisión cifrado, con una solicitud para comparar dichos conjuntos de datos privados con el segundo conjunto de datos almacenado en el servidor,

c) el servidor verifica si la longitud de los hash recibidos es igual a s,

d) si la longitud de los hash es igual a s, el servidor compara cada uno de los valores hash recibidos con la segunda pluralidad de conjuntos de datos almacenados en el servidor, encontrando los conjuntos de datos almacenados que tienen un valor hash reducido idéntico,

e) el servidor enlista cada uno de dichos valores hash encontrados con uno respectivo de dicho valor hash recibido, por ejemplo: 10 valores hash recibidos, 23 coincidencias encontradas

f) el servidor reduce los valores hash encontrados, es decir, los valores hash completos, a una longitud predeterminada de m bits, y

g) el servidor envía para cada uno de los valores hash reducidos y recibidos la lista de valores hash reducidos al dispositivo cliente, como respuesta a la petición anterior.

El procesamiento del parámetro s en la etapa a) es una etapa menos importante, una especie de preprocesamiento. El parámetro s también se puede configurar de forma manual o automática de forma general o individualmente para cada ejemplo de uso. El ordenador cliente y servidor realizan los controles de verosimilitud respectivos y acuerdan un valor específico s sin importar qué parte lo propone: cliente o servidor, o acuerdan una diferencia máxima permitida entre s según se envía desde el servidor, y s, según se envía desde el cliente.

En la etapa b) anterior, en el caso de colisiones en la agenda de direcciones, es decir, cuando dos conjuntos de datos privados diferentes tienen accidentalmente el mismo valor hash, entonces solo se envía este valor hash único desde el cliente al servidor.

Una función hash criptográfica como SHA-1 (técnica anterior) proyecta cada número de teléfono en una cadena de bits de cierta longitud, llamada hash, por ejemplo, 160 bits. Una función hash criptográfica es una función unidireccional. Por lo tanto, determinar la imagen previa, es decir, el número de teléfono del hash, es computacionalmente difícil. En general, se supone que la única forma de obtener la imagen previa de un hash es probar la mayoría de las imágenes previas posibles. De hecho, se espera encontrar una coincidencia después de haber probado la mitad de ellas, pero en el peor de los casos, se tiene que intentar todas. En el siguiente párrafo, vamos a ignorar este pequeño factor constante de 2.

Recudir la longitud s en el sentido de la presente invención significa seleccionar un subconjunto ordenado arbitrario pero sistemático de longitud s del hash. Por ejemplo, se podrían seleccionar los primeros (como se muestra en los dibujos) o los últimos s bits, o bits arbitrarios, como el 1º, 4º, 3º, el 21 y así sucesivamente. La forma cómo se selecciona el subconjunto no tiene influencia en la calidad de la invención, ya que todos los bits tienen propiedades iguales, lo que es una característica de una función hash criptográfica. La única condición es que el servidor y el cliente acuerden el mismo subconjunto y que los dos bits de la misma posición de bit se comparen entre sí.

Las etapas a) y c) también pueden sustituirse por un procedimiento alternativo que produce parámetros de control utilizables en el procedimiento de la invención.

En cuanto a la etapa f), también se puede omitir una segunda reducción, cuando el parámetro m se establece en la longitud total del hash, por ejemplo, 160 bits en el SHA-1.

El procedimiento de la invención se puede aplicar ventajosamente cuando los conjuntos de datos privados son datos de contacto almacenados en un dispositivo informático móvil, como un teléfono inteligente o un teléfono móvil, ya que la capacidad computacional en dichos dispositivos es, aunque está en constante crecimiento, todavía bastante limitada.

Cuando adicionalmente, el ordenador servidor almacena un valor hash reducido y precalculado para cada uno de los conjuntos de datos privados, el cual se ha calculado con el mismo algoritmo que se utilizó en el ordenador cliente, no se debe realizar ningún cálculo de hash "sobre la marcha", es decir, cuando se recibe una solicitud de comparación de contactos en el servidor. Esto ahorra tiempo y permite unos tiempos de respuesta cortos para dichas solicitudes.

Además, el ordenador servidor calcula y almacena ventajosamente un valor hash de longitud media m que enviará de nuevo al cliente para cada conjunto de datos de la comparación.

A continuación, el cliente compara dicho valor hash de longitud media con el prefijo del valor hash completo, y para un par coincidente, el cliente puede calcular fácilmente el conjunto de datos privado respectivo que ha sido comparado por el servidor. Por lo tanto, dicho valor hash de tamaño medio utilizado en el servidor, también ahorra tiempo y permite unos tiempos de respuesta cortos para dichas solicitudes de comparación.

Cuando el ordenador servidor vuelve a calcular repetidamente los valores hash en base a un parámetro de entrada preferiblemente calculado de forma aleatoria, por ejemplo, lo que se conoce generalmente como "sal aleatoria", entonces se encuentra alguna medición contra el uso de las llamadas tablas del arco iris por parte de un posible atacante.

De una forma análoga según se describe para el lado del servidor, también para el dispositivo informático del cliente, preferiblemente un dispositivo móvil, se describe un procedimiento para comparar una primera pluralidad de conjuntos de datos privados, como una agenda de contactos privada y almacenada en un dispositivo cliente de comunicación con un segunda pluralidad de conjuntos de datos, como un gran repositorio de contactos almacenado en un sistema de comunicación basado en servidor, que comprende las etapas de:

a) el dispositivo cliente que procesa o calcula una longitud de valor hash predeterminada s que representa el número de bits de un valor hash criptográfico de una parte única, por ejemplo, el número de teléfono de cada conjunto de datos privados que se transmitirá entre el cliente y el servidor,

b) el dispositivo cliente que calcula el valor hash para cada uno de los conjuntos de datos privados,

c) el dispositivo cliente que reduce el valor hash para cada uno de los conjuntos de datos privados a una longitud predeterminada s,

d) el dispositivo cliente que envía, para cada uno de los conjuntos de datos privados dicho valor hash reducido al servidor, preferiblemente en un canal de transmisión cifrado, que solicita al servidor que compare los conjuntos de datos privados con la segunda pluralidad de conjuntos de datos almacenados en el servidor,

e) recibir un valor hash para cada comparación, cuyo valor hash se ha reducido a una pluralidad de un número predeterminado de m bits por el servidor,

f) el cliente compara los valores hash recibidos y reducidos a m con los conjuntos de datos almacenados en el cliente, revelando así información sobre en qué conjuntos de datos privados y almacenados tanto en el cliente como en dicho servidor, la porción única y respectiva es idéntica.

Ventajosamente, el dispositivo informático del cliente aplica de manera iterativa el cálculo del valor hash de forma repetida y múltiple a fin de proporcionar un mayor grado de privacidad.

El lector experto reconoce que, matemáticamente hablando, el problema de comparación resuelto aquí es encontrar la intersección del conjunto relativamente pequeño de contactos de la agenda de contactos de un usuario con el conjunto relativamente grande de un repositorio que almacena una gran pluralidad de datos de contacto de los miembros registrados de un servicio de comunicación respectivo, a través de una conexión de ancho de banda limitado. El requisito especial es proteger la privacidad. Los contactos y los miembros están identificados en este ejemplo de uso por su número de teléfono en el formato único y estandarizado internacionalmente, por ejemplo, +4199999999.

Así, un experto en la materia apreciará que se describe un procedimiento y un sistema ventajosos que ofrece un protocolo para comparar datos privados y, en particular, para comparar la agenda de contactos con un repositorio de contactos de un servicio de comunicación, en el que el protocolo no revela Información privada ni al proveedor de servicios ni a ningún tercero. Por lo tanto, el anonimato y la privacidad de los contactos del usuario se mantienen, y los contactos no están expuestos a escuchas clandestinas.

Según una característica opcional ventajosa adicional, todos los valores hash incluyen además una sal, es decir, que resulta de una cadena de bits aleatoria, pero constante, añadida al conjunto de datos privados, por ejemplo, añadida al número de teléfono, al nombre o cualquier parte del conjunto de datos privados, de manera clara que cambia determinísticamente el resultado hash. Después de intervalos de tiempo predeterminados, el proveedor de servicios vuelve a calcular todos los hash de los miembros en base a una nueva sal aleatoria, desaprobando una posible tabla hash inversa (tabla del arco iris) calculada por un ataque de fuerza bruta. Al decirle al usuario que utilice esta nueva

sal durante la comparación de contactos, el proveedor de servicios demuestra el gran esfuerzo que necesitaría para aplicar ingeniería inversa en la agenda telefónica del usuario, incluso para ellos mismos.

Con este procedimiento de la invención, no es prácticamente viable que el proveedor de servicios ni ninguna escucha clandestina realicen una ingeniería inversa de la agenda de contactos. Aquí, prácticamente inviable significa que requiere un esfuerzo computacional prohibitivo. E incluso si la escucha clandestina dedicara este esfuerzo, todavía existe la incertidumbre de si un contacto reproducido por ingeniería inversa es realmente un contacto, por ejemplo, del 50%, en función de ciertos parámetros, que pueden configurarse y modificarse mediante parámetros predeterminados adaptables a cualquier ejemplo de uso individual.

Además, se preserva el anonimato y la privacidad de los contactos que no son miembros del servicio de comunicación, ya que los contactos que no participan no pueden ser conocidos con exactitud, ni siquiera por el proveedor de servicios. El proveedor de servicios conoce solo a los contactos que están registrados para el servicio, es decir, que han aceptado los términos y condiciones del proveedor de servicios en cualquier caso.

Por lo tanto, se describe un procedimiento de la invención que incluye la característica técnica de aplicar hashing a las entradas de la agenda de contactos e impedir los ataques de fuerza bruta al no transmitir el hash completo, sino solamente una parte bien elegida. Esto introduce incertidumbre en cualquier ataque de fuerza bruta pero mantiene la tasa de aciertos esperada a la hora de proporcionar una comparación de contactos fiable. Los parámetros se eligen para optimizar la privacidad frente a los requisitos de ancho de banda.

Sin embargo, el procedimiento y el sistema respectivo son eficaces en términos de cálculos y comunicación incluso para un gran número de miembros, para posiblemente todos los números de teléfono utilizados en la Tierra.

3. Breve descripción de los dibujos

La presente invención se ilustra a modo de ejemplo y no está limitada por la forma de las figuras de los dibujos en los que:

La Figura 1 es una descripción general esquemática del sistema implicado en el procedimiento de la invención.

La Figura 2 es un diagrama de interacción que ilustra la interacción entre el cliente y el servidor durante una realización preferida del procedimiento de la invención.

La Figura 3 es una vista con zoom en un detalle de una etapa de la Figura 2 que es ejecutada por el cliente.

La Figura 4 es una representación esquemática y exhaustiva que ilustra un valor hash reducido, un valor hash de tamaño medio y la referencia clara del usuario, en este caso el nombre de una persona, para tres entradas distintas de la agenda de contactos.

4. Descripción detallada de la realización preferida

Con referencia general a las figuras y, de nuevo, con referencia especial a la Figura 1, el procedimiento de la invención se implementa en una arquitectura cliente-servidor tal como se representa en la Figura 1.

Un dispositivo de teléfono inteligente móvil cliente 12 está conectado a través de una conexión a Internet 10 a un ordenador servidor 14 operado por un proveedor de servicios. El ordenador servidor almacena una gran pluralidad de, por ejemplo, varios millones de registros de contactos en un gran repositorio físico, organizado en una base de datos indexada, por ejemplo, una base de datos relacional, en la que cada conjunto de datos está asociado con una persona individual que se ha suscrito al servicio antes de que sus datos de contacto sean almacenados. Los datos de contacto comprenden una o más direcciones físicas de su puesto, números de teléfono, direcciones de correo electrónico y, opcionalmente, otros datos como sexo, edad, profesión, fecha de nacimiento, salario y otros datos que puedan ser relevantes para fines de marketing, como aficiones e intereses, tamaño del cuerpo, etc. El dispositivo cliente y el ordenador servidor se comunican con una conexión cifrada a través de Internet, tal como el protocolo TLS como se conoce en la técnica anterior o, adicionalmente, con un protocolo, por ejemplo, tal como se describe en un número de solicitud de patente europea en tramitación 12007079.2 presentada por el solicitante.

Además, se representa simbólicamente a un atacante 19 que intenta atacar la comunicación en su camino a través de Internet, con medios técnicos de escucha clandestina de la técnica anterior.

Además, se supone que un segundo atacante trabaja en el sitio del proveedor de servicios, tal vez un miembro del personal, por ejemplo, un operador de la base de datos malicioso 20 que tiene acceso de lectura y/o escritura al repositorio.

El procedimiento de comparación de la invención se refiere en la presente memoria también como *Comparación de Contactos Privados* y se abrevia como *CCP.* Deje que en esta realización ejemplar del procedimiento de la invención, el "cliente" sea el dispositivo informático móvil en el que el usuario ejecuta una aplicación de comunicación, por ejemplo, "Facebook" o "Twitter" o "WhatsApp", implementando el procedimiento CCP de la invención incorporado en

una App respectiva. Se comunica con el "servidor" del proveedor de servicios de comunicación, implementando el procedimiento CCP equivalente y, por lo tanto, interactuando con el cliente.

La Figura 2 es un diagrama de interacción que ilustra la interacción entre el cliente 12 y el servidor 14 durante una realización preferida del procedimiento de la invención.

La Figura 3 es una vista con zoom en un detalle de las etapas 225 y 230 de la Figura 2, ejecutado por el cliente. Como primera medida de seguridad, se prefiere cifrar la conexión de Internet 10 entre el cliente y el servidor utilizando la seguridad de la capa de transporte denominada TLS, según se hace en la técnica anterior. Al verificar que el certificado del servidor pertenece al proveedor de servicios, se garantiza que solo el proveedor de servicios conocido y deseado puede escuchar la comunicación, de manera que el protocolo no puede ser escuchado fácilmente de forma clandestina por un tercero, como puede ser el atacante 18 de la Figura 1.

En segundo lugar, según una característica preferida de la presente invención, los números de teléfono 22 de los contactos del cliente no se transmiten en texto simple, sino que se procesarán mediante un procedimiento criptográfico específico, en particular, mediante una función hash criptográfica, tal como se conoce de la técnica anterior, la función MD-5 o SHA-1, en la etapa 225, el procedimiento de la invención proyecta cada número de teléfono en una cadena de bits, denominada hash (valor), que se reduce en la etapa 230 y posteriormente se transfiere al servidor en la etapa 235.

Más particularmente y, preferiblemente, antes de realizar las etapas 225 a 235 mencionadas anteriormente, al iniciar sesión durante la etapa de solicitud del cliente 210 y la etapa de concesión de inicio de sesión del servidor 215, el servidor calcula el valor s = floor(log_2(n)) - u como una indicación de cuántos bits por contacto se deben enviar en un posible procedimiento de comparación en una etapa 216 y se lo comunica al cliente en una etapa 219. El cliente recibe dicho parámetro s y verifica si este valor es razonablemente pequeño. De forma alternativa, el cliente puede decidir qué valores usar en base a cualquier otro criterio.

A continuación, en una etapa 225 y con una referencia adicional especial a la Figura 3, para cada contacto de la agenda de contactos que debe coincidir con el repositorio de contactos del servidor, el cliente calcula un valor hash criptográfico del número de teléfono 22. Preferiblemente y, a modo de ejemplo, se usa el algoritmo SHA-1 y se añade una sal en una etapa 310 antes de aplicar la función de hash al número de teléfono con sal añadida 24 en una etapa 320. Preferiblemente, el hashing se repite un gran número de veces, por ejemplo 1000 veces (no representado en el dibujo); cuantas más veces, más difícil es (directamente proporcional) calcular la imagen previa del valor hash.

En tercer lugar, los valores hash de los números de teléfono de la agenda de direcciones se reducen a un cierto número de bits en una etapa 330 adicional, lo que da como resultado un valor hash reducido 28, a fin de introducir incertidumbre en el caso de que el atacante aplique fuerza bruta.

En referencia a los problemas especiales de este uso de la invención, solo el proveedor de servicios puede atacar debido a la transmisión segura de TLS, pero se debe observarse que los contactos que no son miembros en el repositorio del servidor no se deben comparar, ni siquiera estar expuestos al proveedor de servicios.

Dado que el servicio de comunicación tiene n, por ejemplo, n = 10.000.000 miembros. Deje que L=log_2(n), por lo tanto, L = 23,xxx. Si se comparan L bits de hash, se espera que un hash aleatorio coincida con aproximadamente 1 contacto. El cliente corta los hash en una etapa 330 a s=floor(L)-u bits, donde u es un parámetro predeterminado y configurable por el usuario. Cuanto mayor sea u, mayor es la incertidumbre, ya que se espera que más miembros que no sean contactos coincidan con el hash reducido. Supongamos u=1, entonces

s = floor(23,xxx - 1) = floor(22,xxx) = 22 = s. Por lo tanto, el cliente reduce el valor hash de, por ejemplo, 160 bits (en el caso de usar SHA-1) a 22 bits, lo que resulta en un valor hash reducido indicado con el signo de referencia 28 en la Figura 3. A continuación, el cliente envía la lista de hash reducidos al servidor a través de la conexión cifrada TLS, véase la etapa 235 en la Figura 2.

Después de recibir dichos valores hash reducidos, a fin de evitar exponer los números de teléfono a demasiadas colisiones de hash con falsos positivos, el programa CCP del servidor verifica si la longitud s de los hash reducidos es lo suficientemente larga según se ha solicitado, y verifica en la etapa 238, si el número de valores hash enviados no es inverosímilmente grande, para evitar revelar una parte inadecuadamente grande del conjunto de miembros de datos de contacto al cliente. En este sentido, se propone definir un límite máximo del número de hash enviados al cliente a 5000, como un tamaño máximo posible para la agenda de contactos.

Posteriormente, el servidor compara los hash reducidos recibidos con los miembros de su repositorio del sistema en una etapa 240. Más particularmente, el servidor encuentra los miembros para los cuales el prefijo del hash del número de teléfono coincide con uno de los hash cortos recibidos mediante un procedimiento de comparación respectivo. Para cada comparación, en una etapa 245, el servidor enlista el valor hash correspondiente en una longitud de tamaño medio m, preferiblemente más larga, por ejemplo, de 64 bits, del mismo tipo, lo que hace que los resultados de la comparación sean muy probablemente exactos, pero aún así requiere un alto esfuerzo computacional para descifrar y revelar los números telefónicos de la imagen previa a la comparación.

A continuación, como respuesta a la solicitud de comparación, en lugar de devolver los números de teléfono en texto simple, el servidor envía dichos valores hash de tamaño medio, más largos (por ejemplo, m 64 bits) al cliente, etapa 250.

Después de que el cliente ha recibido dichos valores hash de tamaño medio, en una etapa 260 el cliente compara los valores hash de tamaño medio recibidos con los propios datos de contacto, es decir, encuentra los contactos para los cuales el prefijo del hash del número de teléfono coincide con uno de los hash de tamaño medio recibidos. Cada coincidencia es un contacto, que también es miembro del servicio de comunicación, que forma la intersección resultante de conjuntos. A continuación, el cliente muestra a su usuario que el procedimiento de comparación se ha completado con éxito, etapa 270.

A fin de responder de manera eficaz a las consultas de comparación, el servidor mantiene preferiblemente el valor hash corto con cada conjunto de datos miembros de forma indexada en la base de datos, por lo que no es necesario calcular los valores hash sobre la marcha. Además, el valor hash de tamaño medio para el resultado se almacena preferiblemente en la base de datos para una construcción rápida de la respuesta.

La Figura 4 es una representación esquemática y exhaustiva de la tabla que ilustra en una columna de la izquierda un valor hash reducido indicado en un código hexadecimal ejemplar y que corresponde a 16 bits, en la columna central un valor hash de tamaño medio en un código hexadecimal ejemplar que corresponde a 64 bits, y la columna de la derecha la referencia clara del usuario, en este caso el nombre de una persona, para tres entradas distintas de la agenda de direcciones. Las cadenas de bits idénticas de los hash se denotan dentro de los círculos en líneas discontinuas.

En una implementación eficaz preferida del sistema en la parte del servidor, el servidor mantiene asociado con cada miembro del repositorio de contactos, el hash corto y el hash de tamaño medio, incorporando la sal actual. Almacena el hash corto en formato hexadecimal de modo que se puede realizar una búsqueda indexada eficaz. Si el número de bits no es divisible por cuatro, los bits 0 se rellenan tanto en el hash corto como en la consulta.

Para permitir el cambio de las propiedades de hash como la longitud y la sal mientras se continúa con el servicio, el nuevo cálculo que consume mucho tiempo se debe realizar como tarea de fondo. Con este propósito, el servidor conserva dos copias de la tabla. Mientras que una de ellos se usa activamente, la otra se puede volver a calcular utilizando nuevos parámetros como tarea de fondo. Cuando finaliza el cálculo, el servidor intercambia ambas tablas y empieza a comunicar los nuevos parámetros a los clientes.

Además, a continuación se ofrece una breve prueba de la eficacia del procedimiento de la invención según se ha descrito anteriormente, junto con otras mejoras del mismo:

La conexión entre el cliente y el servidor se cifra y autentifica mediante un certificado de servidor bien conocido por el cliente. El cliente no se autentifica en el servidor, ya que se supone que el servicio está abierto al público en cualquier caso. Interceptar la comunicación cifrada requeriría acceso de escritura al cliente e instalar una versión falsificada o manipulada del software cliente. Teniendo en cuenta este supuesto, el atacante podría simplemente leer la agenda de contactos en texto simple directamente desde el teléfono del cliente y enviarlo a la parte atacante. Así que podemos suponer que la conexión es segura.

De todos modos, el proveedor de servicios de comunicación puede leer la comunicación completa, y el personal no puede considerarse como completamente fiable debido al reglamento de privacidad mencionado en el capítulo introductorio. Por otro lado, el proveedor de servicios de comunicación puede demostrar su credibilidad de privacidad al público mediante el procedimiento de la invención. Por lo tanto, el lector experto entiende que incluso el proveedor de servicios podría reconstruir la agenda de contactos solo con un esfuerzo extremo y, aún así, solo con un grado significativo de incertidumbre, como se justifica a continuación.

Calculamos que existen aproximadamente 10^12 (un billón) de números de teléfono en todo el mundo (incluidos los códigos de país). A pesar de que el conjunto real de números de teléfono en el mundo es, de hecho, mucho más pequeño, no hay una lista completa legible por máquina y disponible públicamente que pueda ser de ayuda. Dado que el procedimiento de la invención preferiblemente aplica iterativamente la función hash del orden de magnitud 10^3 veces, atacar la agenda de contactos requiere un esfuerzo computacional de aproximadamente 10^15 (1000 trillones) cálculos de hash, lo que es muy costoso incluso en ordenadores grandes. Para el cliente, el cálculo de los hash requiere bastante capacidad computacional, pero esto es aceptable (10^6, un millón de cálculos de hash para 1000 contactos), ya que esto se puede producir como tarea de fondo. A medida que la capacidad computacional de los dispositivos móviles aumenta generalmente con el tiempo, tanto en los clientes como en las máquinas disponibles para el atacante, el número de iteraciones puede aumentar.

Para el parámetro u=0 mencionado anteriormente, la tasa de aciertos esperada para cada hash es 1, para u>0, es 2^u (es decir, 2 exp u). El inconveniente es que la u más grande, la lista de resultados devueltos crece, y también lo hacen los requisitos de ancho de banda. Por lo tanto, seleccionar un valor para u es un compromiso entre la privacidad y los requisitos de ancho de banda. Un valor preferido es u=1, por lo que la probabilidad de que un contacto sospechoso sea realmente un contacto es del 50%.

Además, el número de bits enviados al servidor en el procedimiento de la invención será asintóticamente menor que la entropía de bits contenida en la agenda de contactos, por lo que a una parte independiente se le da una pista de que la agenda de contactos no se envía por completo, con solo saber la longitud del mensaje cifrado.

Por ejemplo, para n=10^6 y u=1, una realización de la invención enviaría solo 18 bits por contacto, lo que equivale a aproximadamente 6 dígitos decimales, que es incluso más corto que la parte individual habitual del número de teléfono (es decir, excluido el código de área y de país).

Dado que los hash enviados por el cliente están diseñados para producir colisiones de hash, el proveedor de servicios revela algunos miembros que no son contactos hacia el cliente, como hash de tamaño medio. Nuevamente, el cálculo de la imagen previa todavía está limitado por un esfuerzo prohibitivo, e incluso si se invierte ese esfuerzo, el cliente solo conoce miembros aleatorios. Sin embargo, es posible verificar si hay un miembro con un número de teléfono específico, ya que podría añadirse fácilmente a la agenda de contactos de la parte interesada y esa parte podría convertirse en miembro. Después de todo, exponer a un miembro que un contacto también es miembro del servicio, es el propósito original de todo el sistema de la invención.

El procedimiento y el sistema de la invención se pueden modificar y ampliar de diversas maneras:

El procedimiento de Comparación de Contactos Privados de la invención también es ventajoso cuando se utiliza con dispositivos estacionarios con una conexión a Internet de mayor ancho de banda. La alternativa desventajosa, como se ha mencionado en el capítulo introductorio, de transferir todo el conjunto de miembros del servicio de comunicación, aún no es factible en este caso, ya que produciría mucho tráfico inesperado por parte del usuario. Gracias a una mayor capacidad computacional disponible en este entorno, el número de iteraciones de hash podría incrementarse para una privacidad aún mejor.

Además, en caso de que el servidor desee proporcionar múltiples niveles de privacidad utilizando diferentes conjuntos de parámetros, si se equilibra rendimiento informático y ancho de banda, el servidor solo puede mantener múltiples conjuntos de parámetros con el respectivo grado de equilibrio tal como se ha mencionado anteriormente.

El procedimiento de la invención también se puede utilizar con medios de identificación de contacto distintos a los números de teléfono mencionados anteriormente, por ejemplo, direcciones de correo electrónico. Funciona incluso mejor, ya que el universo de posibles direcciones de correo electrónico es mucho mayor en número y es más fácil de administrar gracias a la ausencia de interfaces de comunicación análogas que el universo de números de teléfono. Por lo tanto, es menos probable que los ataques tengan éxito.

Mientras que en la descripción anterior, un usuario suele ser identificado por un solo número de teléfono, el procedimiento de la invención también puede variar para admitir múltiples números de teléfono por contacto. Los diferentes números de teléfono se tratan como contactos individuales, lo cual también es útil para comunicarse con el destinatario en el dispositivo deseado (p. ej., teléfono del trabajo frente al teléfono de casa).

Para confundir todavía más la agenda de contactos del cliente, el cliente puede añadir contactos virtuales generados aleatoriamente, es decir, conjuntos de datos respectivos, detrás de los cuales no hay una persona física, e incluir los hash respectivos en la solicitud de comparación. Sin embargo, esto no añade mucho más al truncamiento de los hash, excepto que está fuera de la influencia del proveedor de servicios.

Si bien la característica antes mencionada está manchada de incertidumbre para el proveedor de servicios en que la coincidencia de un contacto sea efectivamente un contacto real en el teléfono del usuario, es posible decir con alta probabilidad si un contacto sospechoso a priori es realmente el contacto de un cliente, después de ejecutar el procedimiento de la invención, debido a que es improbable a posteriori que otro contacto del usuario produzca el mismo hash corto.

Para hacer más probable una coincidencia fortuita, se propone, según una característica adicional de la invención, elegir un parámetro u más grande, es decir, aceptar más tráfico, o hacer que la función hash conserve una gran parte de la información original, por ejemplo, combinando los primeros dígitos del número de teléfono (código de país y área, lo cual es probable que vuelva a aparecer y no incluya información privada) con un hash del número de teléfono correspondiente más corto. Esto amplía dicho parámetro u más eficazmente.

La presente invención se puede realizar en hardware, software o una combinación de hardware y software. Una herramienta según la presente invención se puede realizar de manera centralizada en un sistema informático, o de manera distribuida en la que diferentes elementos se extienden a través de varios sistemas informáticos interconectados. Es adecuado cualquier tipo de sistema informático u otro aparato adaptado para llevar a cabo los procedimientos descritos en la presente memoria. Una combinación típica de hardware y software podría ser un sistema informático de propósito general con un programa informático que, al ser cargado y ejecutado, controla el sistema informático de manera que lleve a cabo los procedimientos descritos en la presente memoria.

Los recursos de hardware de almacenamiento permanente mencionados en la presente memoria, como las unidades de disco duro, o los dispositivos ópticos o magneto-ópticos, como los discos compactos o los discos versátiles digitales (DVD), los dispositivos Flash ROM, son en general intercambiables en uso, a menos que se

especifique explícitamente de manera diferente. Por lo tanto, para la mayoría de los propósitos de almacenamiento, se puede utilizar un dispositivo de almacenamiento permanente, no volátil, como se ha mencionado anteriormente, cuando el sistema informático en cuestión no tenga una dedicación especial o relación con ser utilizado como un sistema de mano o adecuado para un bolsillo. Naturalmente, en ausencia de una unidad de disco duro, se prefiere un dispositivo de almacenamiento FLASH-ROM.

La memoria volátil se utiliza generalmente para almacenar temporalmente programas o datos y, en particular, para cargar programas o submódulos de programas, y para el intercambio inmediato de datos con la unidad central de procesamiento (CPU) de un ordenador.

La presente invención también puede integrarse en un producto de programa informático que comprende todas las características que permiten la implementación de los procedimientos descritos en la presente memoria, y que, cuando se carga en un sistema informático, es capaz de ejecutar estos procedimientos.

Medios de programas informáticos o programas informáticos en el contexto presente significan cualquier expresión, en cualquier lenguaje, código o notación, de un conjunto de instrucciones destinadas a hacer que, un sistema que tiene una capacidad de procesamiento de información, realice una función particular ya sea directamente o después de uno o ambos de los siguientes aspectos

a) conversión a otro idioma, código o notación;

b) reproducción en una forma material diferente.

LISTA DE SIGNOS DE REFERENCIA

10 conexión a Internet

12 dispositivo cliente móvil

14 ordenador servidor proveedor de servicios

16 repositorio

18 atacante dentro de Internet

20 operador de servidor malicioso

22 número de teléfono del cliente

24 número de teléfono con sal añadida

26 valor hash de 24

28 valor hash reducido de 26

30 hash de tamaño medio de 26

210 a 330: etapas del procedimiento de la invención

REIVINDICACIONES

1. Un procedimiento para comparar una primera pluralidad de conjuntos de datos privados, como una agenda de contactos privada y almacenada en un dispositivo cliente de comunicación (12) con una segunda pluralidad de conjuntos de datos, como un gran repositorio de contactos almacenado en un sistema de comunicación basado en servidor (14), que comprende las etapas de:

a) dicho sistema de servidor que recibe para cada una de dicha primera pluralidad de conjuntos de datos privados un valor hash reducido (28) de dicho cliente, preferiblemente en un canal de transmisión cifrado, con una solicitud para comparar dichos conjuntos de datos privados con dicha segunda pluralidad de conjuntos de datos almacenados en dicho servidor, en el que dicho valor hash reducido tiene una longitud s que representa el número de bits de un valor hash criptográfico de una parte única, preferiblemente un número de teléfono de cada uno de los conjuntos de datos privados que se transmitirán entre el cliente y el servidor, en el que s se elige de manera que se producirán colisiones de hash en dicho sistema de comunicación basado en servidor (14),

b) dicho servidor compara (240) cada uno de dichos valores hash reducidos y recibidos (28) con dicha segunda pluralidad de conjuntos de datos almacenados en dicho servidor, y encuentra los conjuntos de datos almacenados entre dicha segunda pluralidad de conjuntos de datos que tienen un valor hash reducido idéntico,

c) dicho servidor enlista (245) cada uno de dichos valores hash encontrados con uno respectivo de dichos valores hash reducidos y recibidos (28),

d) dicho servidor envía (250) dichos valores hash encontrados (28) a dicho dispositivo cliente.

2. El procedimiento según la reivindicación 1 en el que

a) para cada comparación, dicho servidor enlista (245) el valor hash correspondiente en una longitud de bits de m bits, en el que dicha longitud de bits de m bits hace que dichos resultados de la comparación sean muy probablemente exactos, y

d) dicho servidor envía (250) dichos valores hash (30) con longitud de m bits a dicho dispositivo cliente.

3. El procedimiento según la reivindicación 1 en el que dicho servidor comprueba (238) si la longitud s de los hash reducidos es lo suficientemente larga.

4. El procedimiento según la reivindicación 1 en el que dichos conjuntos de datos privados son datos de contacto de usuarios almacenados en un dispositivo informático móvil (12).

5. El procedimiento según la reivindicación 1 en el que el ordenador servidor (14) determina un valor s que representa el número de bits de un valor hash criptográfico (28) de una parte única de cada uno de dichos conjuntos de datos privados que se transmitirán al cliente y

transmitir dicho valor s al cliente.

6. El procedimiento según la reivindicación 1 en el que dicho ordenador servidor (14) almacena un valor hash reducido precalculado (28) asociado con un conjunto de datos privados respectivos.

7. El procedimiento según la reivindicación 1 en el que dicho ordenador servidor (14) calcula y almacena un valor hash más largo (30) que el recibido, asociado con un conjunto de datos privado respectivo y envía (250) dicho valor hash más largo de nuevo al cliente para cada conjunto de datos de la comparación.

8. El procedimiento según la reivindicación 1 en el que dicho ordenador servidor (14) vuelve a calcular dichos valores hash (28) en base a un parámetro de entrada predeterminado, preferiblemente calculado aleatoriamente.

9. Un procedimiento para comparar una primera pluralidad de conjuntos de datos privados, como una agenda de contactos privada y almacenada en un dispositivo cliente de comunicación (12) con una segunda pluralidad de conjuntos de datos, como un gran repositorio de contactos almacenado en un sistema de comunicación basado en servidor (14), que comprende las etapas de:

a) para cada uno de dichos conjuntos de datos privados el dispositivo cliente calcula (225) los valores hash criptográficos,

b) dicho dispositivo cliente reduce (230) dichos valores hash a una longitud de valor hash predeterminada s que representa el número de bits de una porción única de dicho valor hash de cada uno de los conjuntos de datos privados que se transmitirán entre el cliente y el servidor, en el que s es elegido de manera que se producirán colisiones de hash en dicho sistema de comunicación basado en servidor (14),

c) dicho dispositivo cliente envía para cada uno de dichos conjuntos de datos privados dicho valor hash reducido (28) al servidor, preferiblemente en un canal de transmisión cifrado, solicitando que el servidor compare, en base a dicho

valor hash reducido, dichos conjuntos de datos privados con dicha segunda pluralidad de conjuntos de datos almacenados en dicho servidor,

d) recibir un valor hash para cada comparación,

e) el cliente compara (260) los valores hash recibidos con los conjuntos de datos almacenados en el cliente, revelando así información sobre en qué conjuntos de datos privados y almacenados tanto en el cliente como en dicho servidor, la porción única y respectiva es idéntica.

10. El procedimiento según la reivindicación 9, en el que dicho valor hash recibido de dicho servidor se ha reducido a una pluralidad de m bits por el servidor, en el que dicha longitud de bits de m bits hace que dichos resultados de la comparación sean muy probablemente exactos.

11. El procedimiento según la reivindicación 9 en el que dicho ordenador cliente (12) aplica iterativamente dicho cálculo de valor hash de forma repetida y múltiple.

12. Un sistema informático que tiene medios para realizar las etapas de un procedimiento según una de las reivindicaciones anteriores 1 a 11.

13. Un programa informático para ejecutar en un sistema de procesamiento de datos basado en servidor (14) e implementado en una red cliente-servidor, la comparación de una primera pluralidad de conjuntos de datos privados, como una agenda de contactos privada y almacenada en un dispositivo cliente de comunicación (12) con una segunda pluralidad de conjuntos de datos, como un gran repositorio de contactos almacenado en un sistema de comunicación basado en servidor (14), que comprende un componente funcional para realizar las etapas respectivas de:

a) dicho sistema de servidor que recibe para cada una de dicha primera pluralidad de conjuntos de datos privados un valor hash reducido (28) de dicho cliente, preferiblemente en un canal de transmisión cifrado, con una solicitud para comparar dichos conjuntos de datos privados con dicha segunda pluralidad de conjuntos de datos almacenados en dicho servidor, en el que dicho valor hash reducido tiene una longitud s que representa el número de bits de un valor hash criptográfico de una parte única, preferiblemente un número de teléfono de cada uno de los conjuntos de datos privados que se transmitirán entre el cliente y el servidor, en el que s se elige de manera que se producirán colisiones de hash en dicho sistema de comunicación basado en servidor (14),

b) dicho sistema de servidor compara (240) cada uno de dichos valores hash reducidos y recibidos (28) con dicha segunda pluralidad de conjuntos de datos almacenados en dicho servidor, y encuentra los conjuntos de datos almacenados entre dicha segunda pluralidad de conjuntos de datos que tienen un valor hash reducido idéntico,

c) dicho sistema de servidor enlista (245) cada uno de dichos valores hash encontrados con uno respectivo de dichos valores hash reducidos y recibidos (28),

d) dicho sistema de servidor envía (250) dichos valores hash encontrados (28) a dicho dispositivo cliente,

cuando dicho programa informático se ejecuta en un ordenador.

14. Un producto de programa informático basado en servidor almacenado en un medio utilizable por ordenador, implementable en una red cliente-servidor, para comparar una primera pluralidad de conjuntos de datos privados, como una agenda de contactos privado almacenado en un dispositivo cliente de comunicación (12) con una segunda pluralidad de conjuntos de datos, como un gran repositorio de contactos almacenado en un sistema de comunicación basado en servidor (14), que comprende medios de programa legibles por ordenador que incluyen un componente funcional para hacer que un ordenador realice las etapas de:

a) dicho sistema de servidor que recibe para cada una de dicha primera pluralidad de conjuntos de datos privados un valor hash reducido (28) de dicho cliente, preferiblemente en un canal de transmisión cifrado, con una solicitud para comparar dichos conjuntos de datos privados con dicha segunda pluralidad de conjuntos de datos almacenados en dicho servidor, en el que dicho valor hash reducido tiene una longitud s que representa el número de bits de un valor hash criptográfico de una parte única, preferiblemente un número de teléfono de cada uno de los conjuntos de datos privados que se transmitirán entre el cliente y el servidor, en el que s se elige de manera que se producirán colisiones de hash en dicho sistema de comunicación basado en servidor (14),

b) dicho sistema de servidor compara (240) cada uno de dichos valores hash reducidos y recibidos (28) con dicha segunda pluralidad de conjuntos de datos almacenados en dicho servidor, y encuentra los conjuntos de datos almacenados entre dicha segunda pluralidad de conjuntos de datos que tienen un valor hash reducido idéntico,

c) dicho sistema de servidor enlista (245) cada uno de dichos valores hash encontrados con uno respectivo de dichos valores hash reducidos y recibidos (28),

d) dicho sistema de servidor envía (250) dichos valores hash encontrados (28) a dicho dispositivo cliente,

cuando dicho producto de programa informático se ejecuta en un ordenador.

15. Un programa informático para ejecutar en un sistema de procesamiento de datos basado en cliente (12) e implementado en una red cliente-servidor, la comparación de una primera pluralidad de conjuntos de datos privados, como una agenda de contactos privada y almacenada en un dispositivo cliente de comunicación (12) con una segunda pluralidad de conjuntos de datos, como un gran repositorio de contactos almacenado en un sistema de comunicación basado en servidor (14), que comprende un componente funcional para realizar las etapas respectivas de:

a) para cada uno de dichos conjuntos de datos privados el dispositivo cliente calcula (225) los valores hash criptográficos,

b) dicho dispositivo cliente reduce (230) dichos valores hash a una longitud de valor hash predeterminada s que representa el número de bits de una porción única de dicho valor hash de cada uno de los conjuntos de datos privados que se transmitirán entre el cliente y el servidor, en el que s es elegido de manera que se producirán colisiones de hash en dicho sistema de comunicación basado en servidor (14),

c) dicho dispositivo cliente envía para cada uno de dichos conjuntos de datos privados dicho valor hash reducido (28) al servidor, preferiblemente en un canal de transmisión cifrado, solicitando que el servidor compare dichos conjuntos de datos privados con dicha segunda pluralidad de conjuntos de datos almacenados en dicho servidor,

d) dicho dispositivo cliente recibe un valor hash para cada comparación,

e) el dispositivo cliente compara (260) los valores hash recibidos con los conjuntos de datos almacenados en el cliente, revelando así información sobre en qué conjuntos de datos privados y almacenados tanto en el cliente como en dicho servidor, la porción única y respectiva es idéntica,

cuando dicho programa informático se ejecuta en un ordenador.

16. Un producto de programa informático cliente almacenado en un medio utilizable por ordenador, implementable en una red cliente-servidor, para comparar una primera pluralidad de conjuntos de datos privados, como una agenda de contactos privada y almacenada en un dispositivo cliente de comunicación (12) con una segunda pluralidad de conjuntos de datos, como un gran repositorio de contactos almacenado en un sistema de comunicación basado en servidor (14),

que tiene un componente funcional para ejecutar las etapas de:

a) para cada uno de dichos conjuntos de datos privados el dispositivo cliente calcula (225) los valores hash criptográficos,

b) dicho dispositivo cliente reduce (230) dichos valores hash a una longitud de valor hash predeterminada s que representa el número de bits de una porción única de dicho valor hash de cada uno de los conjuntos de datos privados que se transmitirán entre el cliente y el servidor, en el que s es elegido de manera que se producirán colisiones de hash en dicho sistema de comunicación basado en servidor (14),

c) dicho dispositivo cliente envía, para cada uno de dichos conjuntos de datos privados dicho valor hash reducido (28) al servidor, preferiblemente en un canal de transmisión cifrado, que solicita al servidor que compare dichos conjuntos de datos privados con dicha segunda pluralidad de conjuntos de datos almacenados en dicho servidor,

d) dicho dispositivo cliente recibe un valor hash para cada comparación,

e) el dispositivo cliente compara (260) los valores hash recibidos con los conjuntos de datos almacenados en el cliente, revelando así información sobre en qué conjuntos de datos privados y almacenados tanto en el cliente como en dicho servidor, la porción única y respectiva es idéntica,

cuando dicho producto de programa informático se ejecuta en un ordenador cliente.

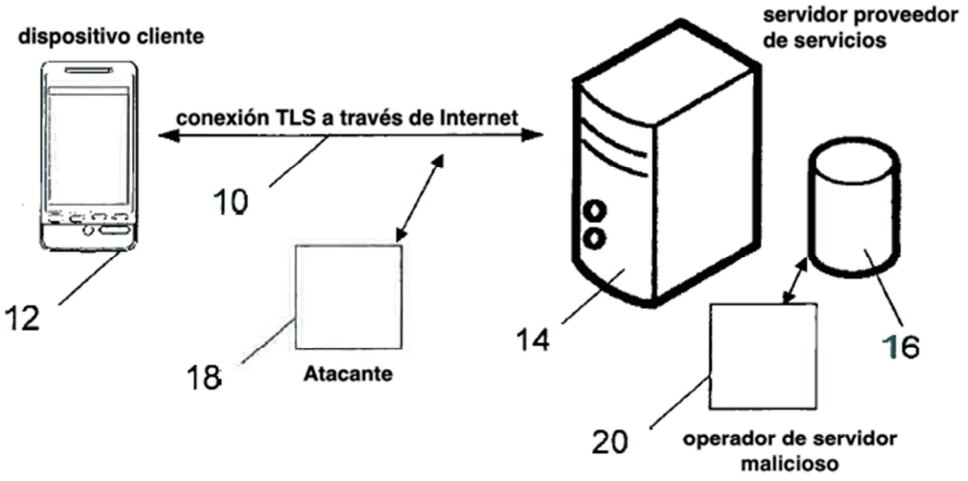
dispositivo cliente
servidor proveedor de servicios
conexión TLS a través de Internet
10
12
14
16
18
Atacante
20
operador de servidor malicioso

FIG. 1

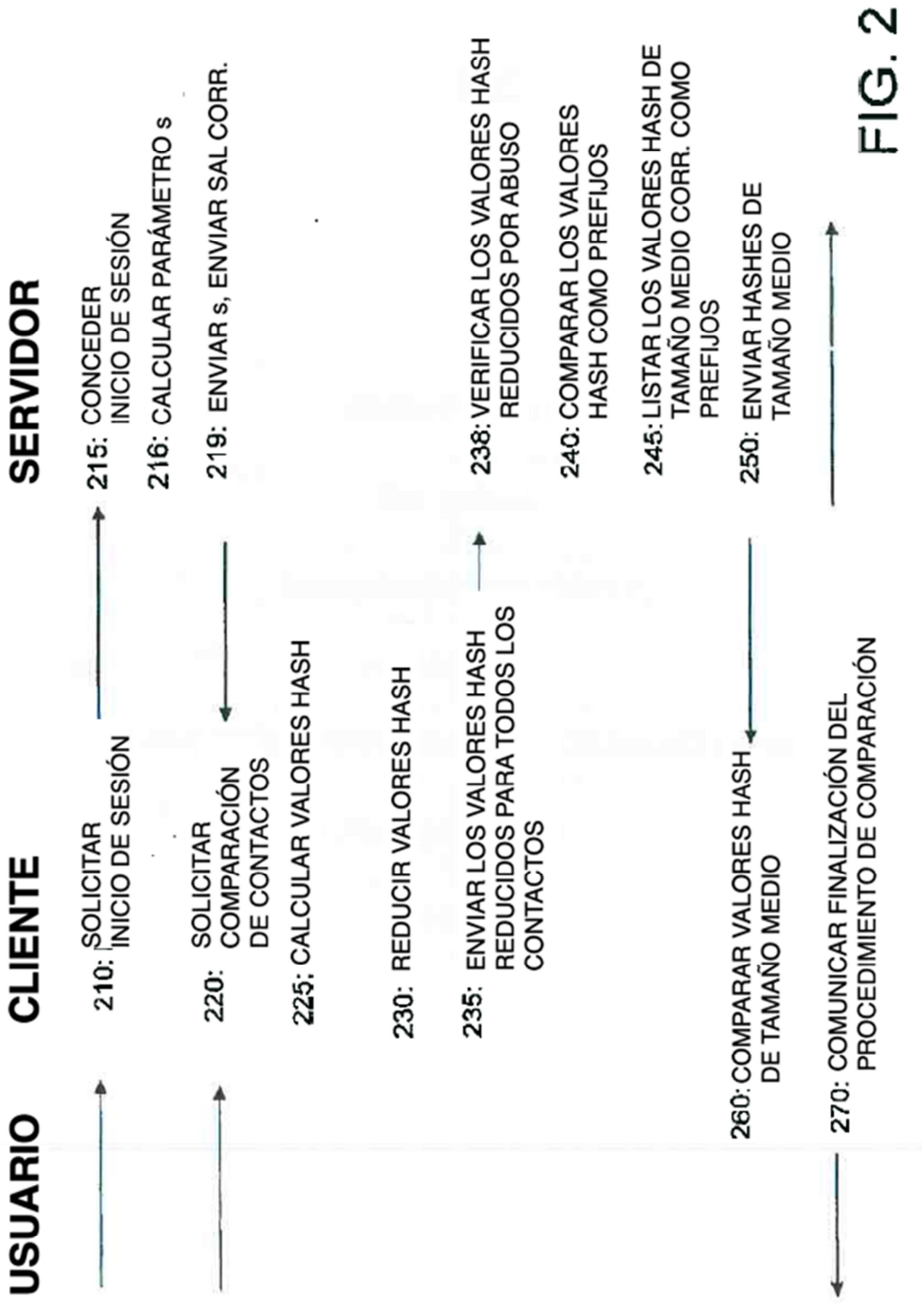
USUARIO
CLIENTE
SERVIDOR
210: SOLICITAR INICIO DE SESIÓN
215: CONCEDER INICIO DE SESIÓN
216: CALCULAR PARÁMETRO s
219: ENVIAR s, ENVIAR SAL CORR.
220: SOLICITAR COMPARACIÓN DE CONTACTOS
225: CALCULAR VALORES HASH
230: REDUCIR VALORES HASH
235: ENVIAR LOS VALORES HASH REDUCIDOS PARA TODOS LOS CONTACTOS
238: VERIFICAR LOS VALORES HASH REDUCIDOS POR ABUSO
240: COMPARAR LOS VALORES HASH COMO PREFIJOS
245: LISTAR LOS VALORES HASH DE TAMAÑO MEDIO CORR. COMO PREFIJOS
250: ENVIAR HASHES DE TAMAÑO MEDIO
260: COMPARAR VALORES HASH DE TAMAÑO MEDIO
270: COMUNICAR FINALIZACIÓN DEL PROCEDIMIENTO DE COMPARACIÓN

FIG. 2

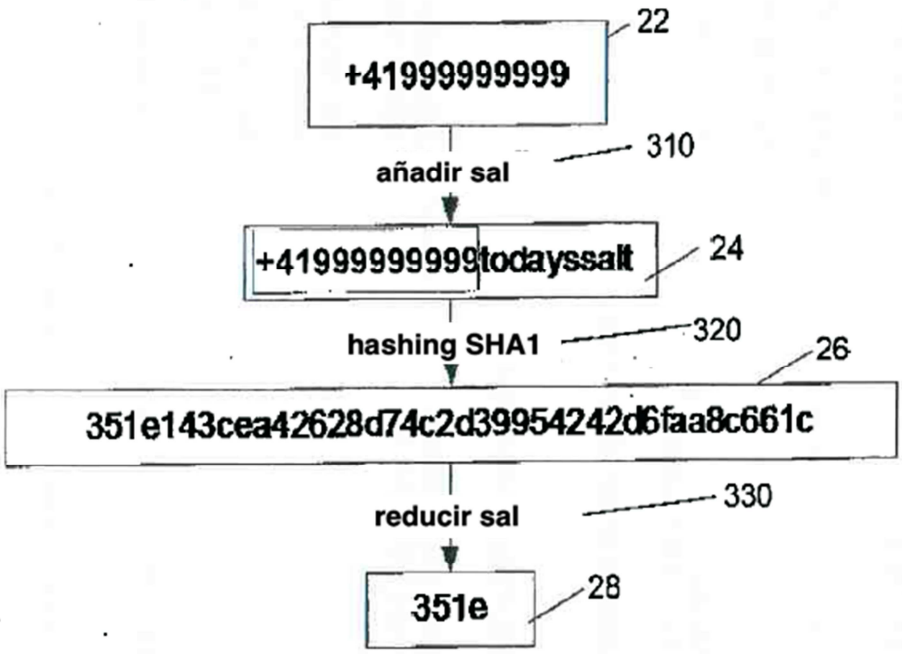
22
+41999999999
310
añadir sal
+41999999999todayssalt
24
hashing SHA1
320
26
351e143cea42628d74c2d39954242d6faa8c661c
330
reducir sal
351e
28

FIG. 3

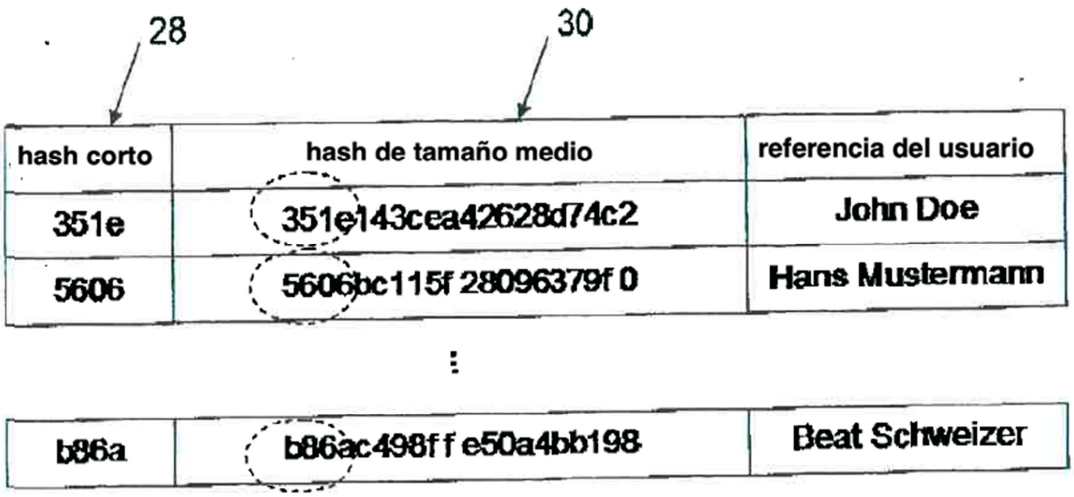
28
30
hash corto
hash de tamaño medio
referencia del usuario
351e
351e143cea42628d74c2
John Doe
5606
5606bc115f28096379f0
Hans Mustermann
b86a
b86ac498ff e50a4bb198
Beat Schweizer

FIG. 4