(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 674 355**

(51) Int. CI.:

G06F 21/62 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **04.11.2008 E 08305772 (9)**

(97) Fecha y número de publicación de la concesión europea: **04.04.2018 EP 2182464**

(54) Título: **Método y sistema para almacenamiento y recuperación de información**

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
29.06.2018

(73) Titular/es:

AMADEUS S.A.S. (100.0%)
485 Route du Pin Montard, Sophia Antipolis
06410 Biot , FR

(72) Inventor/es:

GRANGEON, RAPHAËL;
LISIECKI, FABIEN;
AUJAMES, CÉLINE;
MONTEILLET, MÉLINDA;
ROY, SYLVAIN y
BARRETT, JEFFREY

(74) Agente/Representante:

SUGRAÑES MOLINÉ, Pedro

DESCRIPCIÓN

Método y sistema para almacenamiento y recuperación de información

Campo de la invención

La presente invención se refiere, en general, a un método y un sistema para almacenamiento y recuperación de información electrónica. El método y el sistema de la invención están particularmente dirigidos al almacenamiento de información electrónica que requiere ser asegurada y que debe estar disponible para su uso en diversas aplicaciones. El método se refiere también a un método para almacenamiento y recuperación de datos de aplicación que se refieren a dicha información.

Antecedentes de la invención

El aseguramiento del almacenamiento y la manipulación de información sensible es un problema fundamental en particular para organizaciones en las que muchas aplicaciones han de usar dicha información sensible.

El desarrollo de transacciones electrónicas incrementa el número de transacciones que requieren información sensible. Aparte, para facilitar las transacciones y para ser más atractivas para los usuarios, muchas organizaciones se esfuerzan en obviar la necesidad de reintroducir todos los datos necesarios para completar una transacción. Esto implica almacenar información sensible. También, el almacenamiento de información sensible puede difícilmente ser totalmente seguro. Realmente, las bases de datos que almacenan información sensible pueden posiblemente ser robadas, o pirateadas. Más aún, la información sensible puede recuperarse posiblemente de modo ilegal durante su transmisión desde la base de datos que la almacena a la aplicación que la procesa.

Para incrementar la seguridad del almacenamiento, algunos sistemas permiten dividir la información sensible en dos partes y almacenar cada parte en una base de datos respectiva.

Sin embargo estos sistemas han pasado a ser no totalmente satisfactorios, en particular en un entorno en el que diversas aplicaciones necesitan procesar la misma información sensible.

Estas diversas aplicaciones pueden ejecutarse mediante una única organización que proporciona muchos servicios. Los Sistemas de Distribución Global (GDS) tales como AMADEUS o SABRE son ejemplos típicos de dichas organizaciones que proporcionan muchos servicios implicando a diversas aplicaciones que requieren información sensible.

Diversas diferentes compañías pueden cooperar también para proporcionar servicios integrados a los clientes. Por ejemplo, el comercio electrónico y la banca pueden cooperar para proporcionar a los clientes soluciones sencillas para compra en línea. Diversos comerciantes pueden cooperar también para formar una organización y proporcionar a los clientes un intervalo más amplio de servicios y productos.

Es un objeto de la invención proporcionar un método eficiente y atractivo para el usuario para el almacenamiento y recuperación de información en un entorno en el que muchas aplicaciones pueden necesitar procesar la misma información sensible.

Sumario de la invención

La invención describe un método de almacenamiento y recuperación de información sensible que requiere ser asegurada en un entorno distribuido que comprende una pluralidad de sistemas de aplicación $ASi_{i=1..n}$ que pueden usar dicha información. Ejemplos típicos de información sensible son los números de tarjeta de crédito. El almacenamiento de dicha información comprende las etapas mencionadas a continuación.

Un sistema de aplicación dado AS_i de dicha pluralidad de sistemas de aplicación $ASi_{i=1..n}$ recibe dicha información y genera a partir de dicha información unos datos extraídos y unos datos complementarios. Los datos extraídos y los datos complementarios se generan de tal manera que:

- cuando se toman independientemente no es posible para ningún sistema de aplicación $ASi_{i=1..n}$ usar la información sensible y,
- cuando se toman juntos puede generarse y usarse dicha información.

Adicionalmente el sistema de aplicación dado AS_i genera una información codificada a partir de dicha información. A continuación, envía los datos extraídos y la información codificada a un sistema servidor SS.

El sistema servidor SS genera un índice ID y asigna este índice ID a la información codificada y a los datos extraídos. El sistema servidor SS almacena la información codificada, los datos extraídos, y el índice ID en una base de datos DB_{SS} asociada al sistema servidor SS. El sistema servidor SS envía además el índice ID al sistema de

aplicación AS_i dado de la pluralidad de sistemas de aplicación $ASi_{i=1..n}$.

A continuación, dicho sistema de aplicación AS_i dado asigna el índice ID al complementario. Finalmente, el sistema de aplicación AS_j dado almacena el índice ID junto con dichos datos complementarios en una base de datos DB_{ASj} asociada a dicho servidor de aplicación AS_i dado.

De acuerdo con un primer caso de uso, el sistema de aplicación dado AS_i solo envía los datos extraídos y la información codificada. No hay intercambio de información sensible. Aparte, dado que la base de datos DB_{ASj} del sistema de aplicación dado AS_j solo almacena datos complementarios y el ID y dado que la base de datos DB_{SS} del sistema servidor solo almacena los datos extraídos, la información codificada y el índice ID, entonces los datos extraídos y complementarios nunca se almacenan en la misma base de datos. Por lo tanto, el husmeo en la red, el robo, o pirateo de una cualquiera de las bases de datos del sistema de aplicación dado AS_i o de la base de datos del sistema servidor DB_{SS} no permite obtener ni los datos complementarios ni los datos extraídos o la información completa. Por lo tanto la información no puede reconstruirse y usarse ilegalmente.

Aparte, el índice ID se genera por el sistema servidor SS y no por el sistema de aplicación dado AS_i que recibe la información. Por ello, se asigna un índice ID a un único par formado de información codificada y datos extraídos. Por lo tanto, no hay índices ID simultáneos relacionados con la misma información. Para intercambiar datos relativos a una información dada, los diversos sistemas de aplicación $ASi_{i=1..n}$ pueden compartir el único índice ID asociado a dicha información. En consecuencia, la invención es particularmente conveniente para almacenar información sensible en un entorno distribuido en el que muchos sistemas de aplicación $ASi_{i=1..n}$ usan dicha información sensible.

La base de datos DB_{SS} del sistema servidor SS almacena los datos extraídos, junto con la información codificada y el índice ID. Así, un sistema de aplicación puede recibir o bien los datos extraídos cuando se proporciona a dicho sistema de aplicación el índice ID o bien recibir el índice ID cuando dicho sistema de aplicación accede a los datos extraídos y a la información codificada. Por lo tanto, dicho método permite proporcionar a la pluralidad de sistemas de aplicación $ASi_{i=1..n}$ diversos datos que estos sistemas de aplicación $ASi_{i=1..n}$ requieren para manejar usos predeterminados. En consecuencia, la invención permite un almacenamiento seguro para diversas operaciones que la pluralidad de sistemas de aplicación $ASi_{i=1..n}$ maneja.

Incluso si uno o muchos módulos de aplicaciones $ASi_{i=1..n}$ son robados, forzados o pirateados, la información no puede recuperarse. Realmente obtener el proceso de codificación no permite obtener la información dado que ni los datos extraídos ni la información codificada se almacenan en las bases de datos DB_{ASi} asociadas a los diversos sistemas de aplicación $ASi_{i=1..n}$. Por lo tanto la invención mejora la seguridad del almacenamiento de información.

De acuerdo con la invención, la recuperación de la información sensible en cualquier sistema de aplicación AS_k considerado de entre la pluralidad de sistemas de aplicación $ASi_{i=1..n}$ comprende las siguientes etapas. Dicho sistema de aplicación AS_k considerado recibe el índice ID, y envía el índice ID al sistema servidor SS. El sistema servidor SS recupera de la base de datos del sistema servidor SS los datos extraídos correspondientes a dicho ID. Envía además los datos extraídos a dicho sistema de aplicación AS_k considerado. A continuación, el sistema de aplicación AS_k considerado recibe los datos complementarios y reconstruye la información a partir de los datos extraídos y los complementarios. De ese modo, una aplicación del sistema de aplicación AS_k considerado puede usar la información.

La información sensible completa nunca se intercambia ni almacena. La información solo está disponible en el sistema de aplicación AS_k considerado cuando dicho sistema de aplicación AS_k considerado maneja un proceso que requiere dicha información sensible. Típicamente, la información se almacena en la memoria del proceso y en ningún otro lado. Una vez se completa el uso, la información se elimina. Aparte, los datos extraídos y complementarios no se almacenan en la misma base de datos. Por lo tanto, husmear en la red o robar una cualquiera de las bases de datos no permite obtener la información sensible.

Dado que la invención permite poner a disposición la información sensible a través de introducir un mero índice, la invención obvia la reintroducción adicional o manipulación de información sensible. Por lo tanto, la invención disminuye el riesgo de pérdida o robo de dicha información sensible cuando el usuario la manipula o introduce.

De acuerdo con un primer evento de este primer caso de uso, dicho sistema de aplicación AS_k considerado es el sistema de aplicación AS_i dado. De ese modo, la etapa de recibir los datos complementarios en dicho sistema de aplicación AS_k considerado comprende recibir el índice ID en dicho sistema de aplicación AS_j dado y recuperar gracias al índice ID los datos complementarios que se almacenan en la base de datos DB_{ASi} asociada al sistema de aplicación AS_j dado.

De ese modo, en caso de que el sistema de aplicación AS_k considerado que necesita la información ya haya sido provisto con esta información, los datos complementarios pueden recuperarse tras la recepción del índice ID directamente desde la base de datos DB_{ASk} del sistema de aplicación AS_k considerado. Más aún, el envío del índice al sistema servidor SS activa el envío de los datos extraídos desde el sistema servidor SS. A continuación el sistema de aplicación AS_k obtiene los datos extraídos y complementarios y puede obtener la información requerida.

De acuerdo con un segundo evento de este primer caso de uso, dicho sistema de aplicación AS_k considerado es diferente del sistema de aplicación AS_i dado. Aparte, la aplicación AS_k considerada no comprende una base de datos que indexe los datos complementarios junto con el índice ID. Por ello, la etapa anteriormente mencionada de recibir los datos complementarios en dicho sistema de aplicación AS_k considerado comprende recibir los datos complementarios del sistema de aplicación AS_i dado.

Más particularmente cuando una aplicación AS_k considerada recibe un índice ID y requiere la información correspondiente a este índice ID, comprueba si este índice ID está almacenado en su base de datos DB_{ASk} asociada. En caso de que la base de datos no comprenda este índice ID, o no comprenda los datos complementarios correspondientes a dicho índice, dicho sistema de aplicación AS_k considerado envía una solicitud. Esta solicitud comprende el índice ID para el que se requiere la información. La solicitud llega al sistema de aplicación AS_i dado. El sistema de aplicación AS_i dado se asocia con una base de datos que indexa los datos complementarios con el índice ID.

En respuesta a dicha solicitud, el sistema de aplicación AS_i dado recupera de su base de datos DB_{ASi} los datos complementarios y los envía al sistema de aplicación AS_k considerado. Adicionalmente, el sistema de aplicación AS_k considerado envía el índice al sistema servidor SS para recibir los datos extraídos. De ese modo, el sistema de aplicación AS_k considerado puede combinar los datos complementarios y los datos extraídos para obtener y usar la información requerida.

El sistema de aplicación AS_k considerado puede almacenar además los datos complementarios en su base de datos e indexarlos con el índice ID. De ese modo, la siguiente vez el sistema de aplicación AS_k considerado será capaz de obtener la información sin requerir que el sistema de aplicación AS_i dado le envíe los datos complementarios.

Por lo tanto, diversos sistemas de aplicación distribuidos pueden intercambiar datos para permitir a una aplicación considerada obtener la información requerida aunque esta aplicación considerada nunca haya sido provista hasta el momento con dicha información.

En consecuencia, la invención proporciona un método seguro que obvia la necesidad de que los clientes reintroduzcan la información una vez que dicha información se ha introducido en uno cualquiera de los sistemas de aplicación del entorno distribuido.

Aparte, el intercambio de datos entre diversos sistemas de aplicación es totalmente transparente para el usuario.

De acuerdo con un segundo caso de uso, los datos de aplicación se pretende que se usen por al menos una aplicación de un sistema de aplicaciones $ASi_{i=1..n}$. Aparte, dichos datos de aplicación no requieren un alto nivel de seguridad. Por lo tanto los datos de aplicación pueden almacenarse como tales en la base de datos de una aplicación AS_i dada. Por ejemplo, los datos de aplicación pueden referirse a unos datos de perfil de usuario (programa de fidelidad del usuario, preferencias del usuario, salida y/o llegada del vuelo, solicitudes de servicios asociados con el vuelo, información de hotel o de alquiler de coches, datos de perfil del cliente, etc.).

La recuperación de dichos datos de aplicación en cualquier sistema de aplicación AS_k considerado comprende las siguientes etapas. Dicho sistema de aplicación AS_k considerado recibe la información sensible. Genera los datos extraídos y la información codificada a partir de dicha información. El sistema de aplicación AS_k considerado envía los datos extraídos y la información codificada al sistema servidor (SS).

A continuación, el sistema servidor (SS) recupera de la base de datos del sistema servidor DB_{SS} el índice ID correspondiente al par formado tanto por los datos extraídos como por la información codificada. Envía el índice ID a dicho sistema de aplicación AS_k considerado.

El sistema de aplicación AS_k considerado recibe el índice ID y recupera los datos de aplicación indexados con el índice ID. Así, una aplicación del sistema de aplicación AS_k considerado usa los datos de aplicación.

Como para el primer caso de uso, en este segundo caso de uso, el riesgo de robo de la tarjeta de crédito se reduce significativamente dado que ni el sistema de aplicación AS_k considerado, ni el sistema servidor mantienen la información completa. Aparte, nunca se transmite la información completa en una transmisión simple entre cualquier sistema de aplicación $ASi_{i=1..n}$ y el sistema servidor SS.

De acuerdo con un primer evento de este segundo caso de uso, el sistema de aplicación AS_k considerado es el sistema de aplicación AS_j dado. De ese modo, los datos de aplicación se recuperan de la base de datos DB_{ASj} asociada al sistema de aplicación AS_i dado.

Así, los datos de aplicación pueden recuperarse rápidamente una vez se introduce la información en cualquier sistema de aplicación. Esto permite el uso fácil y amigable de dicha aplicación.

Aparte, dicha recuperación de información es altamente segura. Realmente la base de datos del sistema de

aplicación AS_i dado incluso no ha de comprometer ninguno de los datos complementarios, los datos extraídos o la información. La base de datos del sistema servidor SS no incluye los datos complementarios o la información. Solo la información codificada y los datos extraídos y los datos extraídos se envían al sistema servidor.

Así, los datos complementarios y extraídos nunca se almacenan en la misma base de datos y la información solo está disponible en la memoria de proceso de la aplicación cuando genera los datos extraídos y la información codificada. En consecuencia, el husmeo en la red, el robo o pirateo de cualquier base de datos pasa a ser inútil.

De acuerdo con un segundo evento de este segundo caso de uso, el sistema de aplicación AS_k considerado es un sistema de aplicación que es diferente del sistema de aplicación AS_i dado. Dicho sistema de aplicación AS_k no comprende en su base de datos DB_{ASk} asociada los datos de aplicación indexados con el índice ID. Por ello, la recuperación de los datos de aplicación comprende las etapas indicadas a continuación. Tras la recepción del índice ID desde el sistema servidor, el sistema de aplicación AS_k considerado envía dicho índice ID al sistema de aplicación AS_i dado. A continuación el sistema de aplicación AS_i dado recibe el índice ID y recupera de su base de datos DB_{ASi} los datos de aplicación gracias el índice ID. Finalmente, el sistema de aplicación AS_i dado envía los datos de aplicación al sistema de aplicación AS_k considerado.

Este segundo evento del segundo caso destaca el hecho de que incluso en caso de que un sistema de aplicación AS_k considerado no almacene unos datos de aplicación requeridos, este sistema de aplicación AS_k considerado puede ser provisto con estos datos de aplicación mientras mantiene un alto nivel de seguridad para la información sensible. Realmente, esta información sensible nunca se almacena ni transmite en una única transmisión entre cualquiera de los sistemas de aplicación o entre cualquier sistema de aplicación y el sistema servidor.

En una realización preferida la etapa de generar datos complementarios y datos extraídos incluye dividir la información en una primera porción y una segunda porción.

La invención es particularmente conveniente para el método en el que la información es un número de tarjeta de crédito. Entonces, los datos complementarios pueden corresponder a las cifras visibles del número de la tarjeta de crédito y los datos extraídos pueden corresponder a las cifras ocultas del número de la tarjeta de crédito.

La etapa de generar una versión codificada de la información puede incluir calcular un valor de hash de la información a través de una función de hash. En una realización preferida, la función de hash es desconocida para el sistema servidor (SS). Por lo tanto, la invención permite limitar significativamente el riesgo de que una persona pueda obtener la información sensible a través del acceso a la versión codificada de la información.

La invención proporciona también un sistema para almacenamiento y recuperación de una información que se requiere sea asegurada en un entorno distribuido, comprendiendo una pluralidad de sistemas de aplicación $ASi_{i=1..n}$ que pueden usar dicha información. El sistema de la invención incluye un sistema servidor SS, y un sistema de aplicación AS_j de entre la pluralidad de sistemas de aplicación $ASi_{i=1..n}$. Dicho sistema de aplicación AS_j dado se dispone para:

- recibir dicha información,
- generar a partir de dicha información unos datos extraídos y unos datos complementarios, de modo que dichos datos extraídos y dichos datos complementarios tomados independientemente sean insuficientes para usar dicha información CC# por cualquier sistema de aplicación $ASi_{i=1..n}$ y de modo que dicha información pueda generarse y usarse a partir de dichos datos extraídos y complementarios tomados conjuntamente,
- generar una información codificada a partir de dicha información,
- enviar los datos extraídos y la información codificada a un sistema servidor SS.

El sistema servidor SS se dispone para:

- generar un índice ID y asignar este índice ID a la información codificada y a los datos extraídos,
- almacenar la información codificada, los datos extraídos y el índice ID en una base de datos DB_{SS} asociada al sistema servidor SS,
- enviar el índice ID a dicho sistema de aplicación AS_i dado de la pluralidad de sistemas de aplicación $ASi_{i=1..n}$.

El sistema de aplicación AS_i dado se dispone también para:

- asignar el índice ID a los datos complementarios,
- almacenar el índice ID junto con dichos datos complementarios en una base de datos DB_{ASj} asociada a dicho servidor de aplicación AS_i dado.

Más generalmente, el sistema de acuerdo con la invención comprende el sistema servidor y los sistemas de aplicación AS_i que se disponen para realizar el método descrito anteriormente.

El sistema de aplicación que manipula la información sensible comprende una memoria de proceso y se dispone de

modo que la información esté disponible solo en la memoria de proceso.

Una vez el sistema de aplicación ha usado la información, borra dicha información. Por lo tanto, la información sensible ya no está accesible después del uso. Esto limita el riesgo de robo.

En una realización preferida, el sistema comprende al menos un mecanismo de caché en los medios de procesamiento de un sistema de aplicación (AS_i). El mecanismo caché se dispone para almacenar la información durante el procesamiento de dicha información. Hay una instancia de caché por proceso.

En una realización preferida de la invención, el sistema comprende un componente proxy que es parte del sistema de almacenamiento seguro y que se incluye en el sistema de aplicación. El papel del componente proxy es manejar al menos una de las siguientes etapas que se pretende que tengan lugar en el sistema de aplicación que aloja el componente proxy:

- generar los datos complementarios y los datos extraídos a partir de la información,
- generar la información codificada a partir de dicha información,
- enviar los datos extraídos y la información codificada al sistema servidor SS,
- enviar los datos complementarios en la base de datos asociada al sistema de aplicación que aloja el componente proxy,
- generar mensajes que incluyen los datos a ser enviados desde el sistema de aplicación que aloja el componente proxy, estando dichos mensajes en un formato similar a EDIFACT,
- leer mensajes que incluyen los datos a ser recibidos en el sistema de aplicación que aloja el componente proxy, estando dichos mensajes en un formato similar a EDIFACT.

Típicamente, el componente proxy puede ser una librería middleware. Proporciona varias API (interfaces de programación de aplicación) que interrelacionan la aplicación del sistema de aplicación con el sistema servidor. El componente proxy puede comprender también un mecanismo caché.

A través del manejo del procesamiento de información e intercambios de datos, el componente proxy facilita significativamente la integración de cualquier aplicación en el sistema de la invención.

En otra realización del sistema de la invención, el sistema de aplicación no comprende un componente proxy y maneja todas las acciones requeridas por sí mismo. De ese modo, dicho sistema de aplicación puede formatear/leer por ejemplo mensajes desde el sistema servidor, calcular la versión codificada de la información, etc.

En una realización particular de la invención, el método comprende las siguientes etapas en cualquier sistema de aplicación AS_k considerado. El sistema de aplicación AS_k considerado genera un mensaje de solicitud que comprende diversos índices ID. A continuación, envía al sistema servidor dicho mensaje de solicitud. Además, el sistema servidor busca en su base de datos y recupera cada dato extraído asociado a un índice ID comprendido en el mensaje de solicitud. A continuación el sistema servidor envía un mensaje de respuesta que comprende los datos extraídos recuperados.

A continuación, el sistema de aplicación AS_k considerado recibe el mensaje de respuesta desde el sistema servidor SS. Finalmente, el sistema de aplicación AS_k considerado puede reconstruir toda la información para la que se han recibido los datos extraídos desde el sistema servidor.

En consecuencia, con solo una transacción un sistema de aplicación puede enviar una lista de índices al sistema servidor para recibir todos los datos extraídos correspondientes a la lista de índices.

Dichos procesos por lotes pueden usarse también para recuperar datos de aplicación. Realmente, un sistema de aplicación considerado puede enviar al sistema servidor SS un mensaje de solicitud que contenga una lista de datos extraídos y versiones codificadas de información sensible. El sistema servidor recupera de su base de datos cada índice que está asignado a un par de datos extraídos y versión codificada incluidos en el mensaje de solicitud. A continuación envía al sistema de aplicación considerado un mensaje de respuesta que comprende los índices que se han recuperado. Por ello, el sistema de aplicación considerado puede recuperar los datos de aplicación que están indexados con los índices del mensaje de respuesta.

Por lo tanto, dichos procesos por lotes permiten simplificar y acelerar sustancialmente la recuperación de información para muchos usuarios. En consecuencia, la invención proporciona un método que mejora los servicios ofrecidos a los usuarios de aplicaciones. Dichos procesos por lotes son particularmente útiles cuando nuevas aplicaciones están migrando al sistema de almacenamiento de la presente invención. Realmente cuando se llevan a cabo migraciones, han de almacenarse rápida y fácilmente grandes cantidades de datos.

De acuerdo con la realización anteriormente mencionada, tanto los datos extraídos como la información codificada se envían al sistema servidor para recuperar el índice. El envío tanto de los datos extraídos como de la información codificada permite reducir significativamente el riesgo de recuperar un índice erróneo.

De acuerdo con una realización alternativa, solo se envía la información codificada al sistema servidor para recibir el índice en el sistema de aplicación. Aunque el riesgo de obtener un índice erróneo con esta realización es más alto que cuando están en la solicitud tanto los datos extraídos como la información codificada, dicho riesgo sigue siendo muy bajo. Dicha realización alternativa es particularmente útil cuando se manejan grandes cantidades de datos. Realmente evita manipular y enviar pesados datos extraídos al sistema servidor.

La información puede estar constituida por números, letras, símbolos o combinaciones de los tres. Como se designa en la presente invención, la información no está limitada a números. Los datos extraídos y complementarios pueden incluir también números, letras, símbolos o combinaciones de los tres. Los datos de aplicación pueden comprender cualquier naturaleza de datos y cualquier clase de archivo tales como números, letras, símbolos, imágenes, vídeos, etc.

El sistema puede comprender también medios de seguridad adicionales. Estos medios de seguridad se disponen para reforzar la seguridad de los intercambios entre los diversos sistemas de aplicación y entre los sistemas de aplicación y que el sistema servidor. Estos medios de seguridad pueden comprobar si el remitente de cada mensaje está realmente autorizado. Puede descartar mensajes enviados por remitentes no autorizados. Por ejemplo, el acceso al sistema servidor puede limitarse a un número limitado de sistemas de aplicaciones autorizadas.

En la práctica, los medios de seguridad realizan el cifrado y descifrado de los mensajes intercambiados entre los diversos componentes del sistema. Los medios de seguridad pueden comprender también medios para activar un aviso cuando se intenta una operación anormal. Pueden incluir también medios para supervisar y registrar intercambios, transacciones, y procesamiento de datos.

Breve descripción de los dibujos

Serán más claramente evidentes otras características, objetos y ventajas de la invención a partir de la descripción que sigue, ilustrada por las siguientes figuras:

- La FIG. 1 es un diagrama de bloques de alto nivel de un ejemplo que comprende los componentes principales de un sistema de acuerdo con la invención.
- La FIG. 2 es un diagrama de bloques de alto nivel que ilustra un caso de uso para el almacenamiento de información sensible.
- La FIG. 3 es un diagrama de bloques de alto nivel que ilustra un primer caso de uso de recuperación de información sensible.
- La FIG. 4 es un diagrama de bloques de alto nivel que ilustra un segundo caso de uso de recuperación de información sensible.
- La FIG. 5 es un diagrama de bloques de alto nivel que ilustra un primer caso de uso de recuperación de datos de aplicación.
- La FIG. 6 es un diagrama de bloques de alto nivel que ilustra un segundo caso de uso de recuperación de datos de aplicación.

Descripción detallada de realizaciones particulares

La siguiente descripción detallada de la invención se refiere a los dibujos adjuntos. Aunque la invención incluye realizaciones de ejemplo, son posibles otras realizaciones, y pueden hacerse cambios a las realizaciones descritas sin apartarse del espíritu y alcance de la invención.

La Figura 1 ilustra un sistema de acuerdo con la invención para almacenamiento y recuperación de información.

El sistema comprende n sistemas de aplicación que se referencian como AS_1, AS_2.., AS_i.., AS_n. Cada sistema de aplicación se asocia a una base de datos DB_{AS1}, DB_{AS2}.., DB_{ASi}.., DB_{ASn}. Los sistemas de aplicación comprenden aplicaciones que están dirigidas al uso de información sensible. Se ejecutan mediante una organización constituida por diversas compañías que cooperan o mediante una organización tal como una GDS. Para mejorar la eficiencia de los servicios proporcionados a los usuarios, la organización se esfuerza en obviar la necesidad de que un usuario haya de reintroducir los mismos datos cada vez que se lleva a cabo una transacción.

Esto también disminuye el riesgo de pérdida o robo de información sensible cuando el usuario manipula o introduce dicha información sensible.

El sistema incluye también un sistema de almacenamiento seguro que comprende un sistema servidor SS y una base de datos DB_{SS} asociada con este sistema servidor SS.

El sistema comprende también una red de comunicación, tal como Internet que interconecta cada sistema de aplicación AS_i con el sistema servidor SS. La red de comunicación permite también que los sistemas de aplicaciones AS_i intercambien información conjuntamente en un entorno distribuido. Ventajosamente, los diversos componentes del sistema se localizan remotamente.

Los componentes del sistema de almacenamiento de información ejecutan procesos que proporcionan un almacenamiento y recuperación seguros de información sensible o valiosa.

El almacenamiento y recuperación seguros de información sensible se detalla a continuación a través de casos de uso ilustrativos. En estos casos de uso la información sensible es un número de tarjeta de crédito CC#. Normalmente, el número de tarjeta de crédito está formado por dieciséis cifras.

La Figura 2 ilustra cómo la invención permite el almacenamiento de información sensible.

En la etapa 21, un sistema de aplicación AS_i de entre la pluralidad de sistemas de aplicación $ASi_{i=1..n}$ recibe un número de tarjeta de crédito CC#. Típicamente, este número de tarjeta de crédito es recibido después de que el usuario lo haya introducido a través de una interfaz convencional tal como un teclado por ejemplo. Este número de tarjeta de crédito debe estar fácilmente disponible para su uso en una fase posterior sin requerir que el usuario lo reintroduzca. Por lo tanto, este número de tarjeta de crédito debe almacenarse. El sistema de aplicación AS_i que maneja el almacenamiento del número de tarjeta de crédito se designa como el primer sistema de aplicación AS_i en lo que sigue.

El primer sistema de aplicación AS_i divide el número de la tarjeta de crédito CC# en una primera porción y una segunda porción. La primera porción corresponde, en este ejemplo ilustrativo, a las primeras seis cifras y a las cuatro últimas cifras del número de tarjeta de crédito. Esta primera porción continuará estando disponible en la aplicación AS_i. La segunda porción corresponde a las seis cifras restantes. Esta segunda porción no continuará estando disponible para el sistema de aplicación AS_i. En lo que sigue, la primera y segunda porciones se designan respectivamente cifras visibles A(CC#) y cifras ocultas C(CC#).

Las cifras visibles A(CC#) y las cifras ocultas C(CC#) se generan de tal manera que:

- cuando se toman independientemente no es posible para ningún sistema de aplicación $ASi_{i=1..n}$ usar el número de tarjeta de crédito CC# y,
- cuando se toman juntas puede reconstruirse y usarse dicho número de tarjeta de crédito CC#.

El primer sistema de aplicación AS_i también genera una versión codificada H(CC#) del número de la tarjeta de crédito CC#. Más particularmente, el primer sistema de aplicación AS_i calcula un valor de hash del número de la tarjeta de crédito CC#.

En la etapa 22, el primer sistema de aplicación AS_j envía las cifras ocultas C(CC#) y el número de la tarjeta de crédito codificado H(CC#) al sistema servidor SS.

El sistema servidor SS recibe las cifras ocultas C(CC#) y el número de la tarjeta de crédito codificado H(CC#). Genera un índice ID y asigna este índice ID al número de la tarjeta de crédito codificado H(CC#) y a las cifras ocultas C(CC#). A continuación el sistema servidor SS almacena el número de la tarjeta de crédito codificado H(CC#), las cifras ocultas C(CC#) y el índice ID en una base de datos DB_{SS} asociada al sistema servidor SS (etapa 23). El sistema servidor SS genera el índice ID una vez ha comprobado en su base de datos DB_{SS} que el índice ID está disponible (etapa 24) si la dupla (H(CC#), C(CC#)) ya está almacenada en el sistema servidor SS, entonces el índice ID asignado a esta dupla (H(CC#), C(CC#)) se recupera y se devuelve al sistema de aplicación. De ese modo, la comprobación realizada por el sistema servidor SS no se limita a la comprobación de disponibilidad del índice ID.

En la etapa 25, el sistema servidor SS envía adicionalmente el índice ID al primer sistema de aplicación AS_j de la pluralidad de sistemas de aplicación ($ASi_{i=1..n}$).

A continuación, el primer sistema de aplicación AS_j asigna el índice ID a las cifras visibles A(CC#). Finalmente, el primer sistema de aplicación AS_i almacena el índice ID junto con las cifras visibles A(CC#) en su base de datos DB_{ASi} (etapa 26).

De ese modo, el primer sistema de aplicación AS_i solo envía las cifras ocultas C(CC#) y el número de la tarjeta de crédito codificado H(CC#). No hay intercambio de todo el número de la tarjeta de crédito CC#. Aparte, dado que la base de datos DB_{ASi} del primer sistema de aplicación AS_i solo almacena las cifras visibles A(CC#) y el ID, y dado que la base de datos DB_{SS} del sistema servidor solo almacena las cifras ocultas C(CC#), el número de la tarjeta de crédito codificado H(CC#) y el índice ID, entonces las cifras ocultas C(CC#) y visibles A(CC#) nunca se almacenan en la misma base de datos. Por lo tanto, husmeando en la red, robando, o pirateando una cualquiera de las bases de datos del primer sistema de aplicación AS_j o la base de datos del sistema servidor DB_{SS} no se puede obtener tanto las cifras visibles A(CC#) como las ocultas C(CC#) o el número de la tarjeta de crédito CC# completo. Por lo tanto, el número de la tarjeta de crédito CC# no puede reconstruirse y usarse ilegalmente.

Aparte, el procesamiento de hash se realiza en la aplicación de servidor AS_i. De ese modo, la función continúa siendo desconocida para el sistema servidor SS. Por lo tanto, la invención permite limitar significativamente el riesgo de que una persona obtenga el número de tarjeta de crédito CC# a través del acceso a la versión codificada del

número de tarjeta de crédito CC#.

Más aún, dado que no se almacenan datos enviados por el sistema de aplicación AS_i en la base de datos DB_{ASi} de dicho sistema de aplicación AS_j, entonces, los datos enviados no pueden hacerse cuadrar con los datos almacenados. Por lo tanto, es también inútil tanto piratear la base de datos DB_{ASi} del sistema de aplicación AS_i, como husmear en el mensaje transmitido por este sistema de aplicación AS_j, para obtener la información (por ejemplo el número de la tarjeta de crédito CC#).

Las cifras ocultas C(CC#) generadas por cada sistema de aplicación AS_i se almacenan todas juntas en la base de datos DB_{SS} del sistema servidor. Además, estas cifras ocultas son obligatorias para reconstruir el número de la tarjeta de crédito CC# completo. Por ello, los recursos asignados a asegurar la información sensible pueden concentrarse sobre el sistema servidor SS y su base de datos dedicada DB_{SS}. La invención obvia la necesidad de dispersar estos recursos entre diversos sistemas de aplicación $ASi_{i=1..n}$. Por lo tanto, la seguridad puede mejorarse significativamente en el sistema servidor SS y en su base de datos asociada DB_{SS} para impedir cualquier clase de robo. Este aspecto es particularmente ventajoso en un entorno distribuido en el que los diversos sistemas de aplicación $ASi_{i=1..n}$ y las diversas bases de datos DB_{ASi} asociadas a los sistemas de aplicación se localizan todas remotamente, y/o en un entorno distribuido en el que los diversos sistemas de aplicación $ASi_{i=1..n}$ y las diversas bases de datos DB_{ASi} se localizan remotamente respecto al sistema servidor SS y su base de datos DB_{SS}.

Aparte, el índice ID se genera por el sistema servidor SS y no por cualquier sistema de aplicación AS_i. De ese modo, se asigna un índice ID a un único número de tarjeta de crédito CC#. Por lo tanto, no hay índices simultáneos ID para el mismo número de tarjeta de crédito CC#. Para intercambiar datos con relación a un número de tarjeta de crédito CC# dado, los diversos sistemas de aplicación $ASi_{i=1..n}$ pueden compartir el índice ID único asociado a dicho número de tarjeta de crédito CC#. En consecuencia, la invención es particularmente conveniente para almacenar números de tarjeta de crédito CC# en un entorno distribuido en el que muchos sistemas de aplicación $ASi_{i=1..n}$ usan dicho número de tarjeta de crédito CC#.

Por ejemplo, diversos sistemas de aplicación de una organización pueden compartir datos para obviar la necesidad de que un usuario reintroduzca su número de tarjeta de crédito en cualquier sistema de aplicación AS_k una vez ya se ha introducido el número completo de la tarjeta de crédito en el primer sistema de aplicación AS_i. Este caso de uso se detalla a continuación con referencia a la figura 4.

Por lo tanto, la invención contribuye a mejorar la eficiencia de los servicios proporcionados a usuarios de aplicación.

Con referencia a las figuras 3 y 4 se detalla a continuación un caso de uso que ilustra la recuperación del número de la tarjeta de crédito CC#.

La figura 3 ilustra un evento en el que el primer sistema de aplicación AS_i o cualquier sistema de aplicación que tenga las cifras visibles A(CC#) almacenadas en su base de datos DB_{ASj} necesita recuperar el número de la tarjeta de crédito CC#.

Con este fin, en la etapa 31 el primer sistema de aplicación AS_i recibe el índice ID de su base de datos. A continuación, en la etapa 32 envía el índice ID al sistema servidor SS. El sistema servidor SS recupera de su base de datos DB_{SS} las cifras ocultas C(CC#) indexadas con dicho índice ID (etapas 33 y 34). El sistema servidor SS envía adicionalmente las cifras ocultas C(CC#) al primer sistema de aplicación AS_j (etapa 35).

El primer sistema de aplicación AS_j recupera de su base de datos DB_{ASj} las cifras visibles A(CC#) que están indexadas con el índice ID. Finalmente, el primer sistema de aplicación AS_i combina las cifras visibles A(CC#) y las cifras ocultas C(CC#) recibidas del sistema servidor SS para reconstruir el número de la tarjeta de crédito CC# (etapa 36).

La figura 4 ilustra un segundo evento en el que se necesita recuperar el número de tarjeta de crédito CC# en algún sistema de aplicación AS_k que no ha generado previamente y almacenado el número de la tarjeta de crédito A(CC#) visible indexado con el índice ID. Dicho algún sistema de aplicación AS_k es por lo tanto diferente del primer sistema de aplicación AS_i y se designa como segundo sistema de aplicación AS_k en lo que sigue.

Los diversos sistemas de aplicaciones $ASi_{i=1..n}$ se clasifican dependiendo de la clase de servicio que proporcionan. Estos sistemas de aplicación $ASi_{i=1..n}$ se conocen entre sí y son capaces de identificar el tipo de datos que requieren respectivamente. Cuando un primer sistema de aplicación AS_i recibe un dato que es útil para otros sistemas de aplicación, entonces el primer sistema de aplicación AS_j envía estos datos a dichos otros sistemas de aplicación.

La transmisión de datos útiles puede hacerse funcionar automáticamente una vez que dicho dato está disponible en el primer sistema de aplicación AS_i.

Por ejemplo, una vez se generan cifras visibles A(CC#) por el AS_i y una vez que se recibe el índice ID en AS_i desde el sistema servidor SS, entonces el AS_j envía tanto las cifras visibles A(CC#) como el índice ID a todos los sistemas

de aplicación que están clasificados como que requieren las cifras visibles A(CC#) y el índice ID (etapas 41, 42).

Entonces cada sistema de aplicación que se provee con dichos datos útiles puede usarlos. Por ejemplo, tan pronto como el segundo sistema de aplicación AS_k recibe tanto las cifras visibles A(CC#) como el índice ID (etapa 42) desde el AS_i puede obtener el número de la tarjeta de crédito CC#. Más precisamente, tras la recepción del índice ID, el segundo sistema de aplicación AS_k envía este índice ID al sistema servidor SS (etapa 43). El sistema servidor SS recupera de su base de datos DB_{SS} las cifras ocultas C(CC#) indexadas con dicho índice ID (etapas 44 y 45). El sistema servidor SS envía adicionalmente las cifras ocultas C(CC#) al segundo sistema de aplicación AS_k (etapa 46). A continuación, el segundo sistema de aplicación AS_k combina las cifras visibles A(CC#) recibidas del primer sistema de aplicación AS_i y las cifras ocultas C(CC#) recibidas desde el sistema servidor SS para reconstruir el número de la tarjeta de crédito CC# (etapa 47). Finalmente, la aplicación del segundo sistema de aplicación AS_k puede usar el número de tarjeta CC#.

Ventajosamente, el segundo sistema de aplicación AS_k almacena en su base de datos DB_{ASk} el número de la tarjeta de crédito visible A(CC#) indexado con el índice ID (etapa 48). Así, la siguiente vez el segundo sistema de aplicación AS_k será capaz de obtener el número de la tarjeta de crédito CC# sin requerir que el primer sistema de aplicación AS_j envíe las cifras visibles A(CC#).

Esta realización implica que las cifras visibles A(CC#) y el índice ID se transmiten juntos. Esta realización es particularmente ventajosa en un entorno en el que todas las aplicaciones se ejecutan por una única organización y en la que las interacciones entre las diversas aplicaciones son totalmente transparentes para el usuario. En particular, el usuario final (poseedor de la tarjeta) se supone que no accede al índice ID asignado a su número de tarjeta de crédito CC#.

Como se ha ilustrado en los casos de uso de las figuras 3 y 4, nunca se intercambia ni almacena el número de la tarjeta de crédito CC# completo. El número de la tarjeta de crédito CC# solo está disponible durante su uso en un sistema de aplicación AS_k considerado cuando dicho sistema de aplicación AS_k considerado maneja un proceso que requiere dicho número de tarjeta de crédito CC#. Típicamente, el número de la tarjeta de crédito CC# se almacena en la memoria del proceso y en ningún otro lado. Una vez se completa el uso, el número de la tarjeta de crédito CC# se elimina. Aparte, las cifras visibles A(CC#) y las cifras ocultas C(CC#) no se almacenan en la misma base de datos. Por lo tanto, el husmeo en la red o el robo de una cualquiera de las bases de datos no permite obtener el número de la tarjeta de crédito CC#.

Los casos de uso de las figuras 3 y 4 también ilustran que diversos sistemas de aplicación distribuidos pueden intercambiar datos para permitir a una aplicación AS_k considerada obtener el número de la tarjeta de crédito CC# requerido aunque esta aplicación dada nunca haya sido provista hasta el momento con dicho número de tarjeta de crédito CC# o con las cifras visibles A(CC#). En consecuencia, la invención proporciona un método seguro que obvia la necesidad de que los clientes reintroduzcan el número de la tarjeta de crédito CC# una vez que dicho número de tarjeta de crédito CC# ya se haya introducido en uno cualquiera de los sistemas de aplicación del entorno distribuido. Aparte, el intercambio de datos entre diversos sistemas de aplicación es totalmente transparente para el usuario.

Con referencia a las figuras 5 y 6 se ilustra a continuación un caso de uso que ilustra la recuperación de un dato de aplicación relativo al número de la tarjeta de crédito CC#. Estos datos de aplicación están dirigidos a ser usados por al menos un sistema de aplicación. Aparte, dichos datos de aplicación no requieren un alto nivel de seguridad. Por lo tanto los datos de aplicación pueden almacenarse como tales en una base de datos de la aplicación dada. Por ejemplo, los datos de aplicación pueden ser relativos a un dato de perfil de usuario (programa de fidelidad del usuario, preferencias del usuario, foto del usuario, etc.).

La figura 5 ilustra un evento en el que el sistema de aplicación que necesita usar los datos de aplicación ya los ha almacenado e indexado en su base de datos. Dicho sistema de aplicación se designa en lo que sigue primer sistema de aplicación AS_i.

En la etapa 51 el primer sistema de aplicación AS_i recibe el número de tarjeta de crédito CC#. A continuación, genera las cifras ocultas C(CC#) y el número de la tarjeta de crédito H(CC#) codificado a partir de dicho número de tarjeta de crédito CC#. En la etapa 52, envía las cifras ocultas C(CC#) y el número de la tarjeta de crédito H(CC#) codificado al sistema servidor SS. A continuación, el sistema servidor SS recupera de su base de datos DB_{SS} el índice ID correspondiente tanto a las cifras ocultas C(CC#) como al número de la tarjeta de crédito codificado H(CC#) (etapas 53 y 54). A continuación el sistema servidor SS envía el índice ID al primer sistema de aplicación AS_i (etapa 55). Tras la recepción del índice ID, el primer sistema de aplicación AS_i recupera los datos de aplicación indexados con el índice ID (etapa 56 y 57). Finalmente, la aplicación del primer sistema de aplicación AS_j puede usar los datos de aplicación.

La figura 6 ilustra un evento en el que la base de datos del sistema de aplicación que necesita usar los datos de aplicación no almacena o indexa dichos datos de aplicación en su base de datos. Dicho sistema de aplicación se designa en lo que sigue segundo sistema de aplicación AS_k.

En la etapa 61 el segundo sistema de aplicación AS_k recibe un número de tarjeta de crédito CC#. Las etapas 62, 63, 64, 65 son sustancialmente idénticas a las etapas 52, 53, 54, 55 detalladas anteriormente. En la etapa 65 el segundo sistema de aplicación AS_k recibe el índice ID desde el sistema servidor SS. Después de haber comprobado que su base de datos DB_{ASk} no almacena unos datos de aplicación indexados con el índice recibido, el segundo sistema de aplicación AS_k envía este índice ID al primer sistema de aplicación AS_j (etapa 66). El primer sistema de aplicación AS_j busca en su base de datos DB_{ASj} y recupera los datos de aplicación indexados con el índice ID (etapas 67 y 68). A continuación el primer sistema de aplicación AS_j envía los datos de aplicación al segundo sistema de aplicación AS_k (etapa 69). Finalmente los datos de aplicación están disponibles para su uso en una aplicación del segundo sistema de aplicación AS_k.

Ventajosamente, el segundo sistema de aplicación AS_k almacena en su base de datos DB_{ASk} los datos de aplicación indexados con el índice ID.

De ese modo, los datos de aplicación pueden recuperarse rápidamente siempre que los datos de aplicación ya se hayan indexado y almacenado en cualquier base de datos y siempre que el número de la tarjeta de crédito ya se haya introducido una vez.

Esto permite un uso fácil y amigable de los datos de aplicación convirtiendo por lo tanto las aplicaciones en más atractivas para los usuarios. Aparte, el intercambio de datos entre diversos sistemas de aplicación es totalmente transparente para el usuario.

Como para los casos de uso que se refieren a la recuperación del número de tarjeta de crédito, en los casos de uso que se dirigen a la recuperación de datos de aplicación el riesgo de robo de la tarjeta de crédito es significativamente reducido dado que ni el sistema de aplicación AS_i, AS_k, ni el sistema servidor SS mantienen el número de la tarjeta de crédito CC# completo. Aparte, no se transmite nunca el número de tarjeta de crédito CC# completo en una única transmisión entre cualquier sistema de aplicación $ASi_{i=1..n}$ y el sistema servidor SS o entre dos sistemas de aplicación AS_j, AS_k.

En una realización preferida, el sistema de almacenamiento comprende al menos un módulo de procesamiento y transacción en un sistema de aplicación. Como se ilustra en las realizaciones particulares de las figuras 2 a 6, el sistema comprende un componente proxy en cada sistema de aplicación. De acuerdo con otra realización solo algunos de los sistemas de aplicación pueden asociarse a un componente proxy. El componente proxy es parte del sistema de almacenamiento seguro. El proxy maneja al menos alguna de las siguientes etapas que se pretende tengan lugar en un sistema de aplicación en el que se incluye dicho componente proxy:

- generar las cifras visibles A(CC#) y las cifras ocultas C(CC#) a partir del número de tarjeta de crédito CC#,
- generar el número de la tarjeta de crédito codificado H(CC#) a partir de dicho número de tarjeta de crédito CC#,
- enviar las cifras ocultas C(CC#) y el número de tarjeta de crédito codificado H(CC#) al sistema servidor SS,
- enviar el índice ID al sistema servidor SS para recuperación del número de tarjeta de crédito CC# adicional,
- enviar las cifras visibles A(CC#) a la base de datos asociada al sistema de aplicación,
- recuperar las cifras visibles A(CC#) de la base de datos asociada al sistema de aplicación,
- generar mensajes que incluyen los datos a ser enviados desde el sistema de aplicación al sistema servidor o a otros sistemas de aplicación, estando dichos mensajes en un formato similar al EDIFACT,
- leer los mensajes recibidos desde el sistema servidor o desde otros sistemas de aplicación, estando dichos mensajes en un formato similar al EDIFACT.

Típicamente, el componente proxy puede ser una librería middleware. Proporciona varias API (interfaces de programación de aplicación) que interrelacionan el software de aplicación del sistema de aplicación con el sistema servidor. El componente proxy puede comprender también un mecanismo de caché. El mecanismo de caché se dispone para almacenar el número de la tarjeta de crédito CC# durante el procesamiento de dicho número de tarjeta de crédito CC#. Hay una instancia de caché por proceso de modo que el número de la tarjeta de crédito CC# ya no está disponible una vez se ha usado por la aplicación del sistema de aplicación.

A través del manejo del procesamiento e intercambios de datos del número de tarjeta de crédito CC#, el componente proxy facilita la integración de cualquier aplicación en el sistema de la invención.

REIVINDICACIONES

1. Método de almacenamiento y recuperación de una información sensible (CC#), requiriéndose que dicha información (CC#) sea asegurada en un entorno que comprende una pluralidad de sistemas de aplicación ($ASi_{i=1..n}$) que pueden usar dicha información (CC#), **caracterizado por que** almacenar dicha información (CC#) comprende las siguientes etapas:

• en un sistema de aplicación (AS_j) dado de dicha pluralidad de sistemas de aplicación ($ASi_{i=1..n}$):

- recibir dicha información (CC#) (21),
- generar a partir de dicha información (CC#) unos datos extraídos (C(CC#)) y unos datos complementarios (A(CC#)), de modo que dichos datos extraídos (C(CC#)) y dichos datos complementarios (A(CC#)) tomados independientemente sean insuficientes para usar dicha información (CC#) y de modo que dicha información (CC#) pueda generarse a partir de dichos datos extraídos (C(CC#)) y datos complementarios (A(CC#)) tomados conjuntamente,
- generar una información codificada (H(CC#)) a partir de dicha información (CC#), incluyendo la etapa de generar una información codificada (H(CC#)) el cálculo de un valor de hash de la información (CC#) a través de una función de hash,
- enviar los datos extraídos (C(CC#)) y la información codificada (H(CC#)) a un sistema servidor (SS) (22),

• en el sistema servidor (SS):

- generar un índice (ID) y asignar este índice (ID) a la información codificada (H(CC#)) y a los datos extraídos (C(CC#)),
- almacenar la información codificada (H(CC#)), los datos extraídos (C(CC#)) y el índice (ID) en una base de datos (DB_{SS}) asociada al sistema servidor (SS) (23, 24),
- enviar el índice (ID) a dicho sistema de aplicación (AS_j) dado de la pluralidad de sistemas de aplicación ($ASi_{i=1..n}$) (25),

• en dicho sistema de aplicación (AS_i):

- asignar el índice (ID) a los datos complementarios (A(CC#)),
- almacenar el índice (ID) junto con dichos datos complementarios (A(CC#)) en una base de datos (DB_{ASj}) asociada a dicho sistema de aplicación (AS_i) dado (26).

2. Método de la reivindicación 1 en el que recuperar dicha información (CC#) en cualquier sistema de aplicación (AS_k) considerado de entre la pluralidad de sistemas de aplicación ($ASi_{i=1..n}$) comprende las siguientes etapas:

• en dicho sistema de aplicación (AS_k) considerado:

- recibir el índice (ID) (31),
- enviar el índice (ID) al sistema servidor (SS) (32, 43),

• en el sistema servidor (SS):

- recuperar de la base de datos (DB_{SS}) del sistema servidor (SS) los datos extraídos (C(CC#)) correspondientes a dicho índice (ID) (33, 34, 44, 45),
- enviar los datos extraídos (C(CC#)) a dicho sistema de aplicación (AS_k) considerado (35, 46),

• en dicho sistema de aplicación (AS_k) considerado:

- recibir los datos complementarios (A(CC#)) (36, 42),
- reconstruir la información (CC#) a partir de los datos extraídos y los complementarios (A(CC#)) (37, 47).

3. Método de la reivindicación 2 en el que dicho sistema de aplicación (AS_k) considerado es el sistema de aplicación (AS_i) dado y en el que recibir los datos complementarios (A(CC#)) en dicho sistema de aplicación (AS_k) considerado comprende recuperar gracias al índice (ID) los datos complementarios (A(CC#)) que se almacenan en la base de datos (DB_{ASi}) asociada al sistema de aplicación (AS_j) dado (36).

4. Método de la reivindicación 2 en el que dicho sistema de aplicación (AS_k) considerado es un sistema de aplicación que es diferente del sistema de aplicación (AS_i) dado y que no comprende una base de datos que almacena el índice (ID) junto con los datos complementarios (A(CC#)), y en el que recibir los datos complementarios (A(CC#)) en dicho sistema de aplicación (AS_k) considerado comprende recibir los datos complementarios (A(CC#)) desde el sistema de aplicación (AS_i) dado (41, 42).

5. Método de la reivindicación 1 en el que los datos complementarios son datos de aplicación (programa de

fidelidad) dirigidos a ser usados por al menos un sistema de aplicación ($ASi_{i=1..n}$) y que no requieren un alto nivel de seguridad.

6. Método de la reivindicación 5 que comprende adicionalmente que recuperar dichos datos de aplicación (programa de fidelidad) en cualquier sistema de aplicación (AS_k) considerado comprende las siguientes etapas:

- en dicho sistema de aplicación (AS_k) considerado:

  - recibir dicha información (CC#) (51, 61),
  - generar los datos extraídos (C(CC#)) a partir de dicha información (CC#) y la información codificada (H(CC#)) a partir de dicha información (CC#),
  - enviar los datos extraídos (C(CC#)) y la información codificada (H(CC#)) al sistema servidor (SS) (52, 62),

- en el sistema servidor (SS):

  - recuperar de la base de datos (DB_{SS}) del sistema servidor (SS) el índice (ID) correspondiente tanto a los datos extraídos (C(CC#)) como a la información codificada (H(CC#)) (53, 54, 63, 64),
  - enviar el índice (ID) a dicho sistema de aplicación (AS_k) considerado (55, 65),

- en dicho sistema de aplicación (AS_k) considerado,

  - recibir el índice (ID),
  - recuperar los datos de aplicación (programa de fidelidad) indexados con el índice (ID) (56, 57, 66, 67, 68, 69).

7. Método de la reivindicación 6 en el que dicho sistema de aplicación (AS_k) considerado es el sistema de aplicación (AS_j) dado y en el que los datos de aplicación (programa de fidelidad) se recuperan de la base de datos (DB_{ASj}) asociada al sistema de aplicación (AS_j) dado (56).

8. Método de la reivindicación 6 en el que dicho sistema de aplicación (AS_k) considerado es un sistema de aplicación que es diferente del sistema de aplicación (AS_j) dado y que no comprende en su base de datos asociada los datos de aplicación (programa de fidelidad) indexados con el índice (ID) y en el que recuperar los datos de aplicación (programa de fidelidad) comprende las siguientes etapas:

- en el sistema de aplicación (AS_j) dado,

  - recibir el índice (ID) correspondiente a los datos de aplicación (programa de fidelidad) que se quiere recuperar (66),
  - recuperar en la base de datos (DB_{ASj}) del sistema de aplicación (AS_j) dado los datos de aplicación (programa de fidelidad) gracias al índice (ID) (67, 68),
  - enviar los datos de aplicación al sistema de aplicación (AS_k) considerado (69).

9. Método de cualquiera de las reivindicaciones 1 a 4 en el que la etapa de generar datos complementarios (A(CC#)) y datos extraídos (C(CC#)) incluye dividir la información (CC#) en una primera porción (A(CC#)) y una segunda porción (C(CC#)).

10. Método de la reivindicación 9 en el que la información (CC#) es un número de tarjeta de crédito (CC#) y en el que los datos complementarios (A(CC#)) corresponden a cifras visibles del número de tarjeta de crédito (CC#) y los datos extraídos (C(CC#)) corresponden a cifras ocultas del número de tarjeta de crédito (CC#).

11. Método de cualquiera de las reivindicaciones 1 a 10 en el que la función de hash es desconocida para el sistema servidor (SS).

12. Método de cualquiera de las reivindicaciones 1 a 11 en el que al menos una de las siguientes etapas realizadas en cualquier sistema de aplicación (AS_i) es manejada por un componente proxy:

- generar los datos complementarios (A(CC#)) y los datos extraídos (C(CC#)) a partir de la información (CC#),
- generar la información codificada (CC#) a partir de dicha información (CC#) (H(CC#)),
- enviar los datos extraídos (C(CC#)) y la información codificada (H(CC#)) al sistema servidor (SS),
- enviar los datos complementarios (A(CC#)) en la base de datos (DB_{ASj}) asociada al sistema de aplicación (AS_j) dado,
- generar mensajes que incluyen los datos a ser enviados desde dicho cualquier sistema de aplicación (AS_i), estando dichos mensajes en un formato similar a EDIFACT,
- leer mensajes que incluyen los datos a ser recibidos en el sistema de aplicación, estando dichos mensajes en un formato similar a EDIFACT.

13. Método de cualquiera de las reivindicaciones 1 a 12 en el que en cualquier momento, la información (CC#) está disponible solo en una memoria de proceso de un sistema de aplicación (AS_i) que procesa dicha información (CC#).

14. Sistema para almacenamiento y recuperación de una información sensible (CC#), requiriéndose que dicha información (CC#) sea asegurada en un entorno que comprende una pluralidad de sistemas de aplicación ($ASi_{i=1..n}$) que pueden usar dicha información (CC#), **caracterizado por que** el sistema incluye:

• un sistema servidor (SS) y
• un sistema de aplicación (AS_j) dado de entre dicha pluralidad de sistemas de aplicación ($ASi_{i=1..n}$), disponiéndose dicho sistema de aplicación (AS_j) dado para:

- recibir dicha información (CC#) (21),
- generar a partir de dicha información (CC#) unos datos extraídos (C(CC#)) y unos datos complementarios (A(CC#)), de modo que dichos datos extraídos (C(CC#)) y dichos datos complementarios (A(CC#)) tomados independientemente sean insuficientes para usar dicha información (CC#) por cualquier sistema de aplicación ($ASi_{i=1..n}$) y de modo que dicha información (CC#) pueda generarse a partir de dichos datos extraídos y complementarios (A(CC#)) tomados conjuntamente,
- generar una información codificada (H(CC#)) a partir de dicha información (CC#), generando una información codificada (H(CC#)) que incluye el cálculo de un valor de hash de la información (CC#) a través de una función de hash,
- enviar los datos extraídos (C(CC#)) y la información codificada (H(CC#)) a un sistema servidor (SS) (22),

• disponiéndose el sistema servidor (SS) para:

- generar un índice (ID) y asignar este índice (ID) a la información codificada (H(CC#)) y a los datos extraídos (C(CC#)),
- almacenar la información codificada (H(CC#)), los datos extraídos (C(CC#)) y el índice (ID) en una base de datos (DB_{SS}) asociada al sistema servidor (SS) (23, 24),
- enviar el índice (ID) a dicho sistema de aplicación (AS_j) dado de la pluralidad de sistemas de aplicación ($ASi_{i=1..n}$) (25),

• disponiéndose también dicho sistema de aplicación (AS_i) dado para:

- asignar el índice (ID) a los datos complementarios (A(CC#)) relacionados con la información (CC#),
- almacenar el índice (ID) junto con dichos datos complementarios (A(CC#)) en una base de datos (DB_{ASj}) asociada a dicho sistema de aplicación (AS_i) dado (26).

15. El sistema de la reivindicación 14 que está dispuesto para realizar el método de cualquiera de las reivindicaciones 2 a 13.

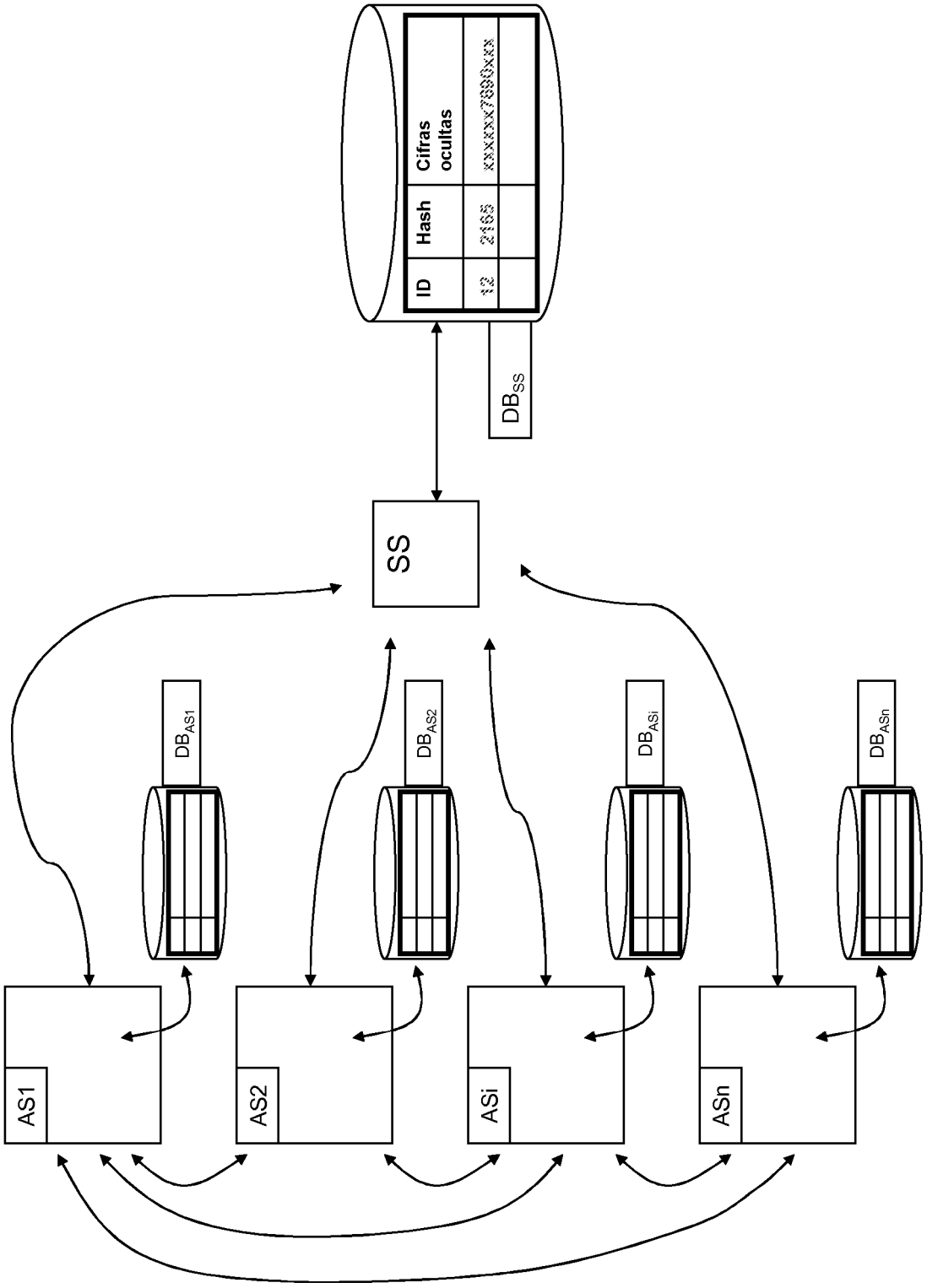
ID
Hash
Cifras ocultas
12
2165
xxxxx7890xxx
DBSS
SS
DBAS1
DBAS2
DBASi
DBASn
AS1
AS2
ASi
ASn

Figura 1

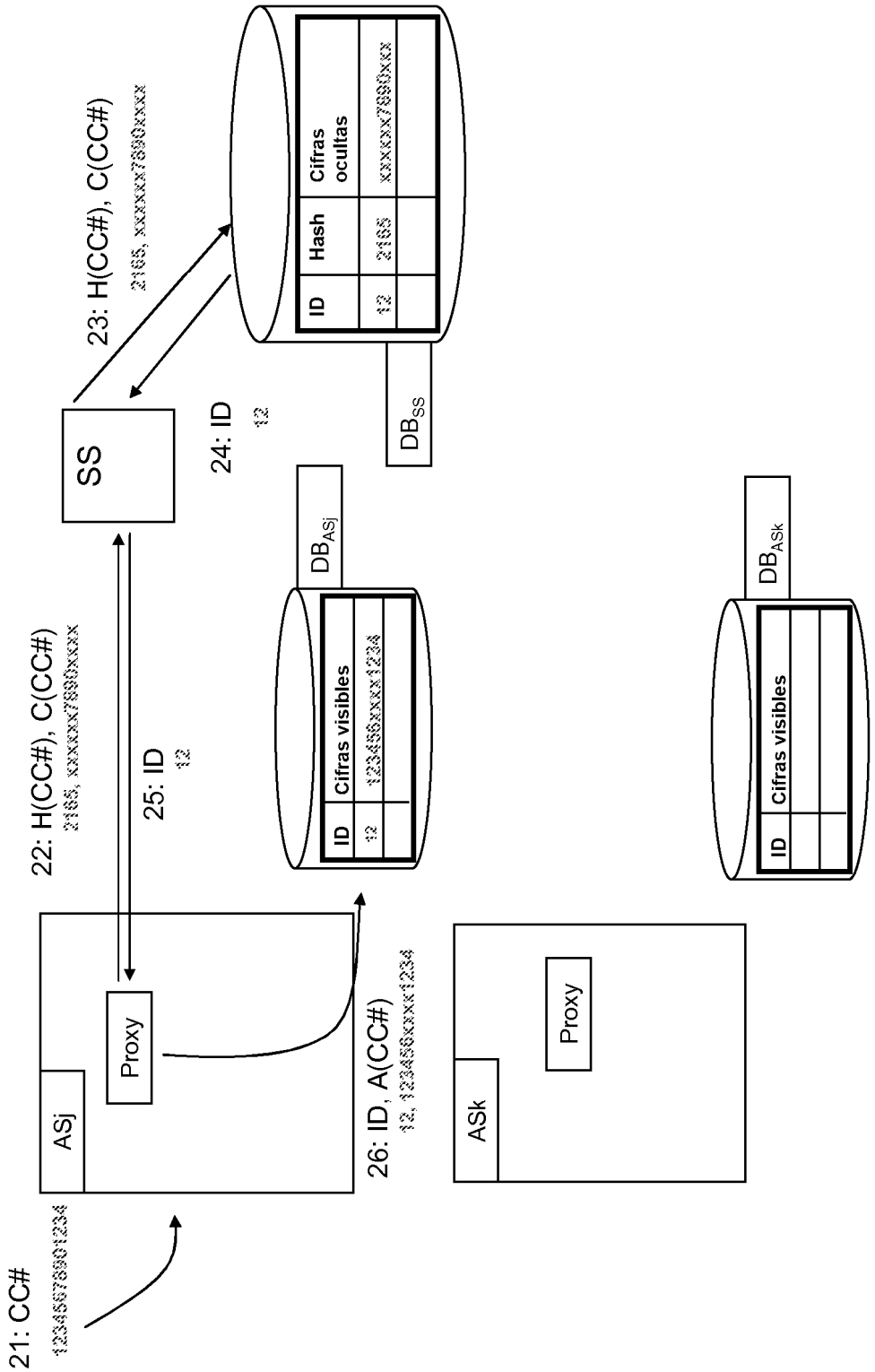
21: CC#
1234567890123456
ASj
Proxy
22: H(CC#), C(CC#)
2165, xxxxxx7890xxxx
25: ID
12
SS
23: H(CC#), C(CC#)
2165, xxxxxx7890xxxx
24: ID
12
DBSS
ID
Hash
Cifras ocultas
12
2165
xxxxxx7890xxxx
DBASj
ID
Cifras visibles
12
123456xxxx1234
26: ID, A(CC#)
12, 123456xxxx1234
ASk
Proxy
DBASk
ID
Cifras visibles

Figura 2

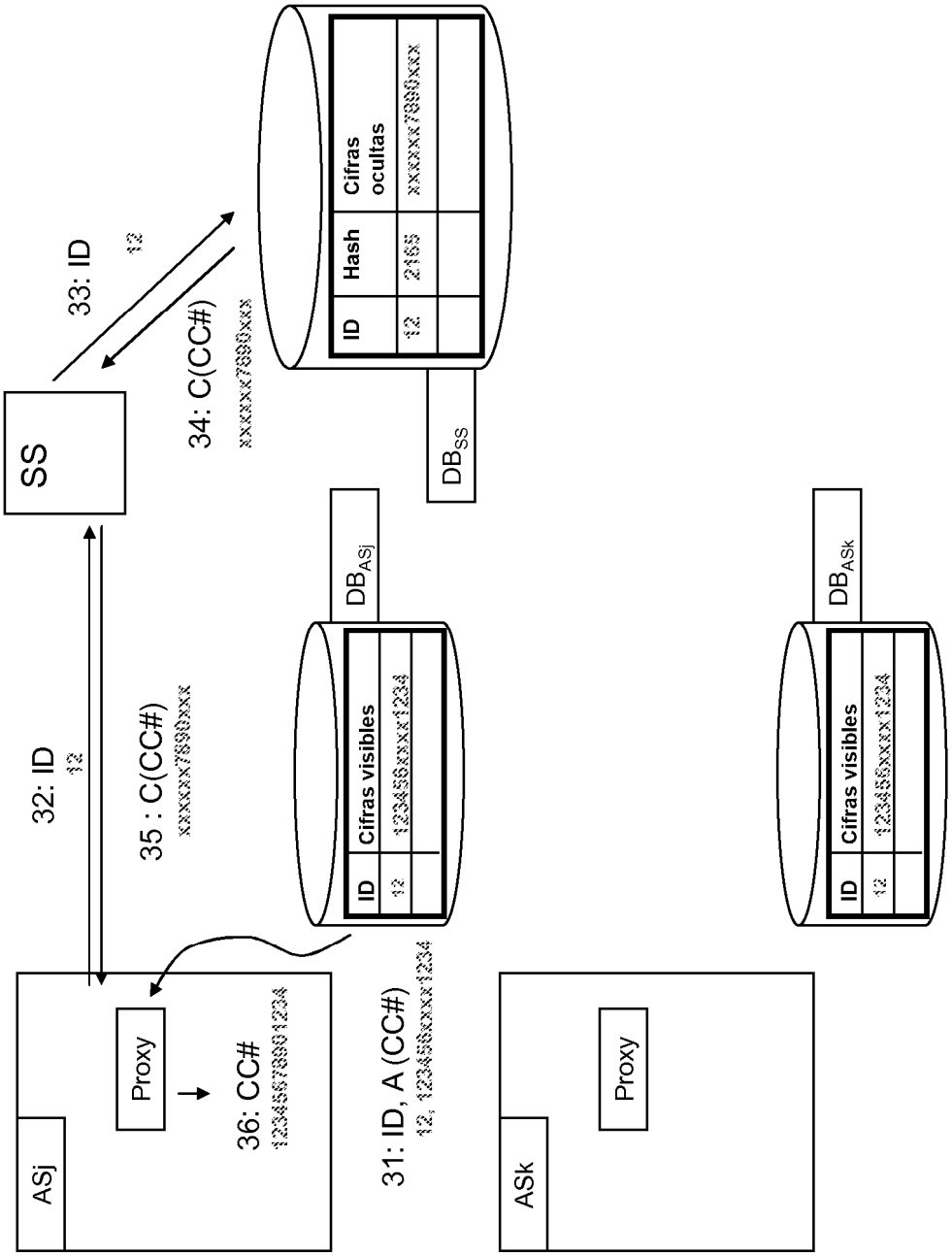
SS
33: ID
12
34: C(CC#)
xxxxxx7890xxx
ID
Hash
Cifras ocultas
12
2165
xxxxxx7890xxx
DBSS
32: ID
12
35 : C(CC#)
xxxxxx7890xxx
DBASj
ID
Cifras visibles
12
123456xxxx1234
DBASk
ID
Cifras visibles
12
123456xxxx1234
ASj
Proxy
36: CC#
1234567890123A
31: ID, A (CC#)
12, 123456xxxx1234
ASk
Proxy

Figura 3

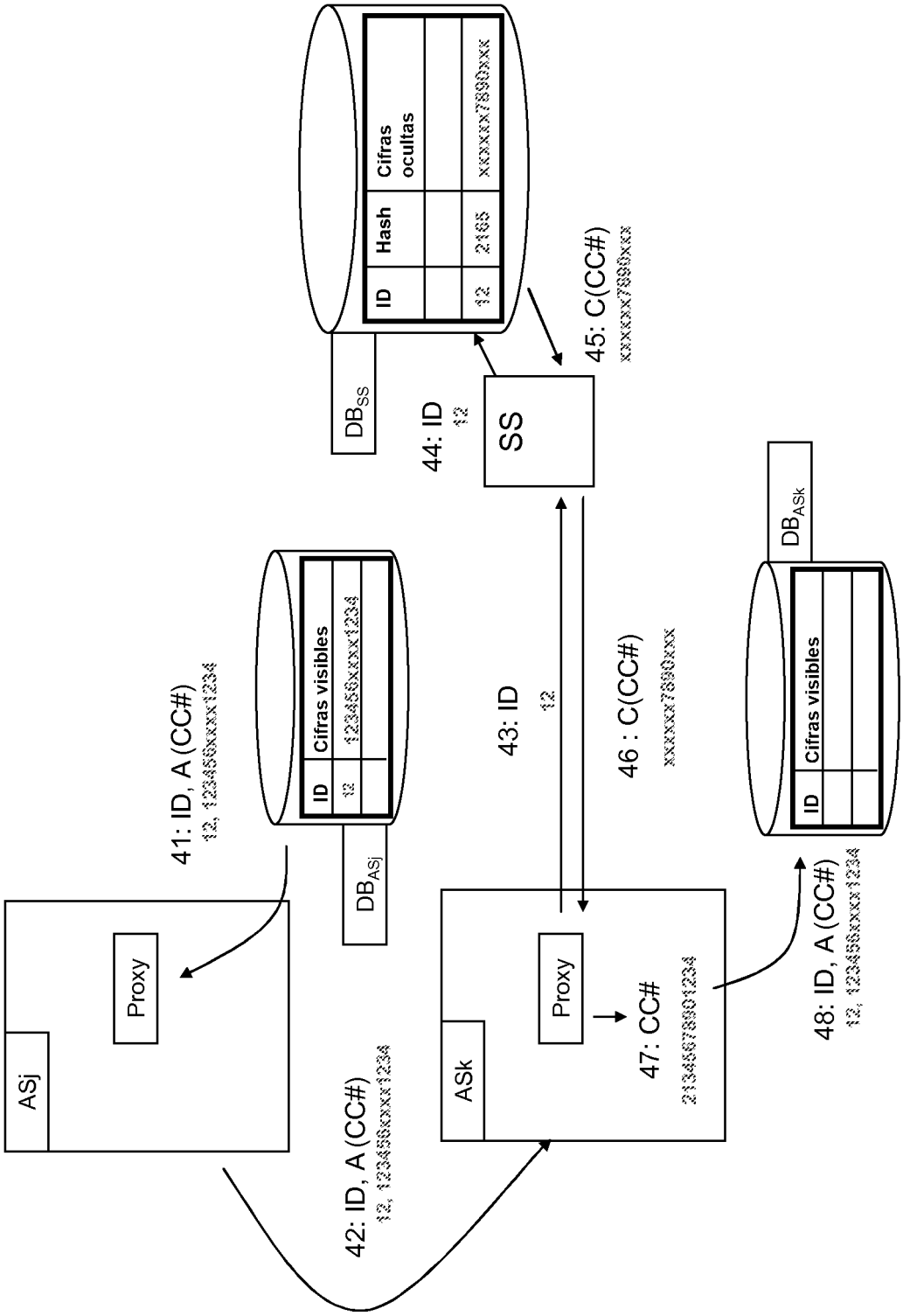
ID
Hash
Cifras ocultas
12
2165
xxxxx7890xxx
DBSS
44: ID
12
SS
45: C(CC#)
xxxxx7890xxx
DBASk
ID
Cifras visibles
41: ID, A(CC#)
12, 123456xxxx1234
ASj
Proxy
DBASj
ID
Cifras visibles
12
123456xxxx1234
43: ID
12
46 : C(CC#)
xxxxx7890xxx
ASk
Proxy
47: CC#
2134567890123 4
48: ID, A(CC#)
12, 123456xxxx1234
42: ID, A(CC#)
12, 123456xxxx1234

Figura 4

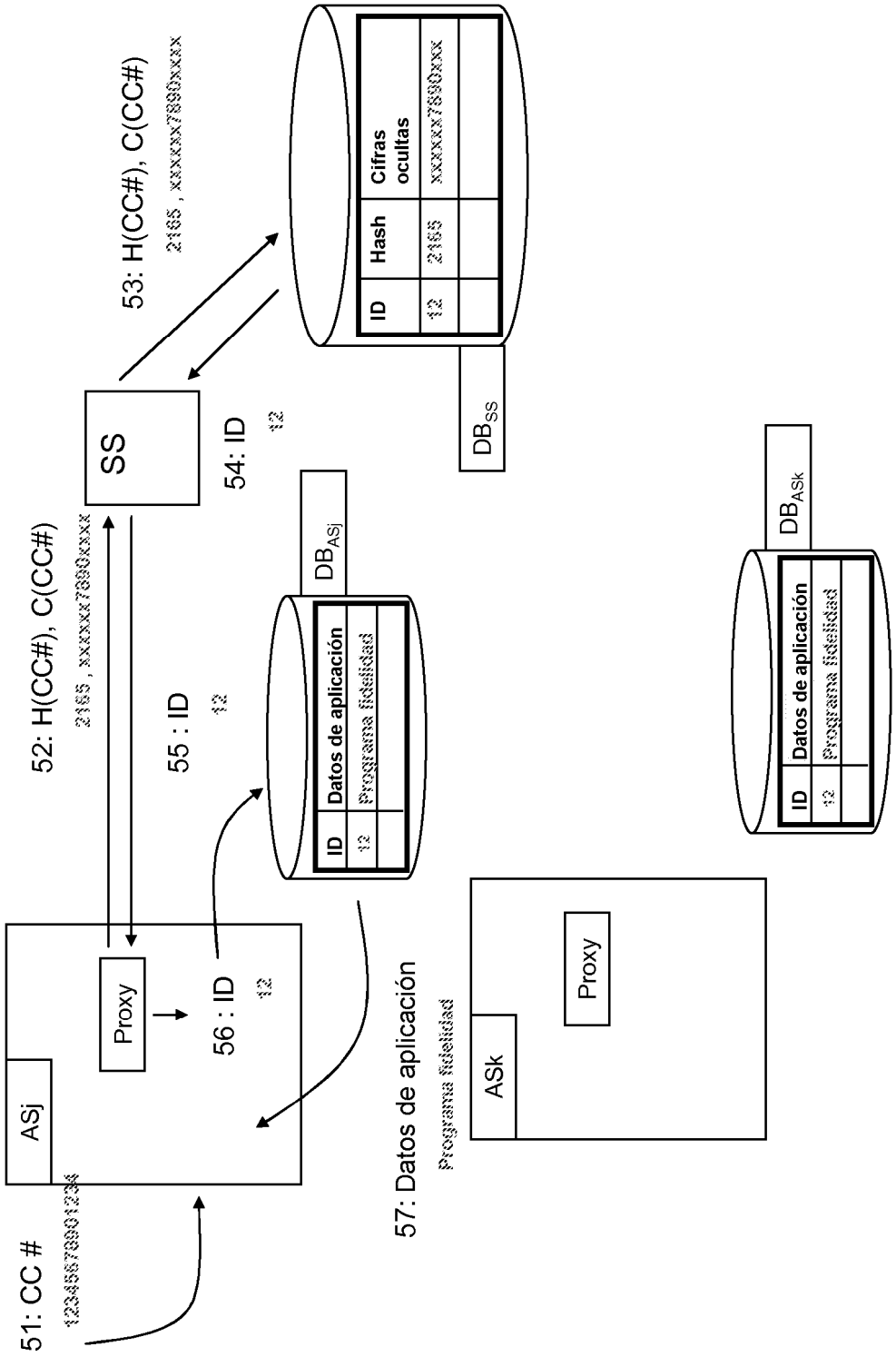
51: CC #
1234567890123456
ASj
Proxy
52: H(CC#), C(CC#)
2165 , xxxxxx7890xxxx
SS
53: H(CC#), C(CC#)
2165 , xxxxxx7890xxxx
54: ID
12
55 : ID
12
56 : ID
12
57: Datos de aplicación
Programa fidelidad
ID
Hash
Cifras ocultas
12
2165
xxxxxx7890xxx
DBSS
DBASj
ID
Datos de aplicación
Programa fidelidad
12
ASk
Proxy
DBASk
ID
Datos de aplicación
12
Programa fidelidad

Figura 5

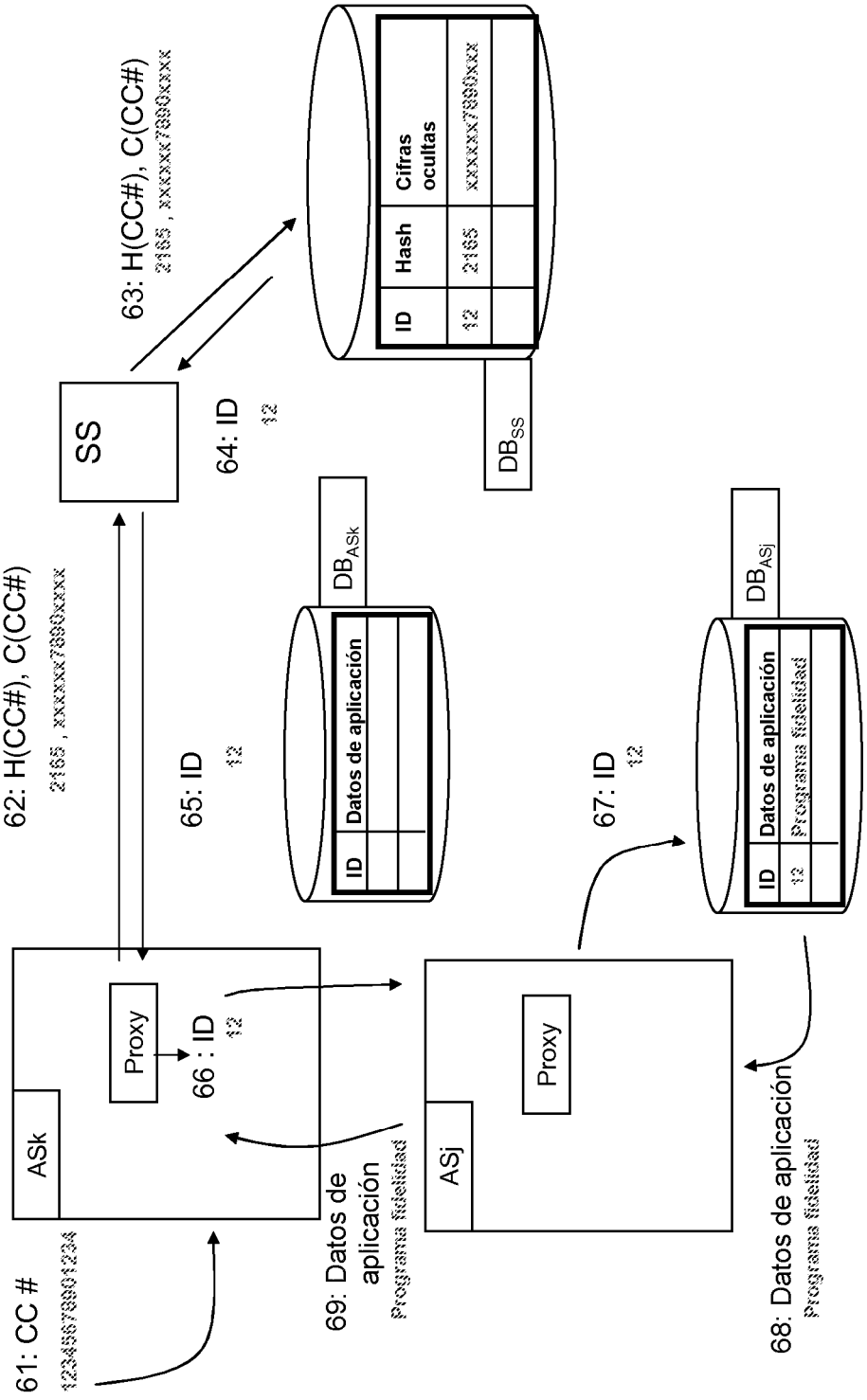
61: CC #
1234567890123x
ASk
Proxy
66 : ID
12
62: H(CC#), C(CC#)
2165 , xxxxx7890xxx
65: ID
12
SS
63: H(CC#), C(CC#)
2165 , xxxxx7890xxx
64: ID
12
ID
Hash
Cifras ocultas
12
2165
xxxxx7890xxx
DBSS
DBASk
ID
Datos de aplicación
69: Datos de aplicación
Programa fidelidad
ASj
Proxy
67: ID
12
DBASj
ID
Datos de aplicación
12
Programa fidelidad
68: Datos de aplicación
Programa fidelidad

Figura 6