(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 265 826**

(51) Int. Cl.:
H04L 9/30 (2006.01)
H04L 9/08 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Número de solicitud europea: **00107473 .1**
(86) Fecha de presentación : **06.04.2000**
(87) Número de publicación de la solicitud: **1043864**
(87) Fecha de publicación de la solicitud: **11.10.2000**

(54) Título: **Sistema y método para distribución de documentos.**

(30) Prioridad: **21.12.1999 US 469726**
06.04.1999 US 128164 P

(45) Fecha de publicación de la mención BOPI:
01.03.2007

(45) Fecha de la publicación del folleto de la patente:
01.03.2007

(73) Titular/es: **ContentGuard Holdings, Inc.**
103 Foulk Road, Suite 205-M
Wilmington, Delaware 19803, US

(72) Inventor/es: **Wang, Xin**

(74) Agente: **Ungría López, Javier**

DESCRIPCIÓN

Sistema y método para distribución de documentos.

La invención se refiere a métodos criptográficos, y más particularmente a sistemas y métodos para la distribución eficiente de documentos codificados a una gran cantidad de receptores.

Una de las más importantes cuestiones que impiden la distribución ampliamente generalizada de documentos digitales mediante comercio electrónico es la falta de protección actual de los derechos de protección intelectual de los propietarios de contenidos durante la distribución y uso de tales documentos digitales. Los esfuerzos para resolver este problema se han denominado "Gestión de los Derechos de Propiedad Intelectual" ("IPRM"), "Gestión de los Derechos de Propiedad Digital" ("DPRM"), "Gestión de Propiedad Intelectual" ("IPM"), "Gestión de Derechos" ("RM"), y "Gestión Electrónica de Derechos del Autor" ("ECM").

Un documento, tal como se denomina en este documento, es cualquier unidad de información sujeta a distribución o transferencia, incluyendo pero no limitando correspondencia, libros, revistas, diarios, periódicos, otros escritos, software, fotografías y otras imágenes, secuencias de audio y vídeo, y otras presentaciones multimedia. Un documento puede realizarse de forma impresa sobre papel, como datos digitales sobre un medio de almacenamiento, o de cualquier otra forma conocida de una variedad de medios.

En el mundo de los documentos impresos, un trabajo creado por un autor usualmente se proporciona a un editor, que da forma e imprime numerosas copias del trabajo. Las copias se envían por el distribuidor a las librerías u otras tiendas de distribución detallistas, en las cuales se compran las copias por los usuarios finales.

Mientras que la baja cualidad de las copias y el elevado costo de distribución del material impreso han servido como elementos disuasivos de la copia ilegal de la mayoría de los documentos impresos, es de lejos demasiado fácil copiar, modificar, y redistribuir documentos electrónicos no protegidos. Por consiguiente, son necesarios algunos métodos de protección de documentos electrónicos para dificultar la copia ilegal de los mismos. Esto servirá como un elemento disuasivo para la copia, incluso si es aun posible, por ejemplo, realizar copias de los documentos impresos y duplicarlos con el método antiguo.

Con documentos impresos, hay una etapa adicional de digitalizar los documentos antes de que puedan ser distribuidos electrónicamente; esto sirve como un elemento disuasivo. Desafortunadamente, se ha reconocido ampliamente que no hay método viable para impedir que la gente haga distribuciones no autorizadas de documentos electrónicos con los sistemas actuales de comunicación y computación de propósito general, tales como ordenadores personales, estaciones de trabajo, y otros dispositivos conectados sobre redes de área local (LAN), intranet, y la Internet. Muchos intentos de proporcionar soluciones basadas en hardware para impedir copias no autorizadas han probado ser un fracaso.

Se han empleado dos esquemas básicos para intentar resolver el problema de protección de documentos: contenedores seguros y sistemas de confianza.

Un "contenedor seguro" (o simplemente un documento cifrado) ofrece un método para guardar codificados los contenidos del documento hasta que se cumplen un conjunto de condiciones de autorización y se han guardado algunos términos de los derechos de autor (por ejemplo, pago por uso). Después de que se han verificado diversas condiciones y términos con el proveedor del documento, se libera el documento al usuario en forma clara. Productos comerciales tales como los Cryptolopes de IBM y los InterTrust's Digiboxes entran dentro de esta categoría. Claramente, el advenimiento de los contenedores seguros proporciona una solución para proteger el documento durante la distribución sobre canales inseguros, pero no proporciona ningún mecanismo para impedir a los legítimos usuarios obtener el documento descodificado y a continuación de usarlo redistribuirlo violando la propiedad intelectual de los propietarios del contenido.

Los mecanismos criptográficos se usan típicamente para encriptar (o "cifrar") documentos que a continuación se distribuyen y se almacenan públicamente, y por ultimo se descifran por usuarios autorizados. Esto proporciona una forma básica de protección durante la distribución del documento desde el distribuidor del documento al pretendido usuario sobre una red pública, así como durante el almacenamiento del documento sobre medios inseguros.

En la aproximación a los "sistemas de confianza", el sistema entero es responsable de impedir el uso no autorizado y la distribución del documento. Construir un sistema de confianza usualmente conlleva introducir nuevo hardware tal como un procesador seguro, almacenamiento seguro, y dispositivos seguros de suministro. Esto requiere también que las aplicaciones software que corren sobre sistemas de confianza estén certificadas para ser de confianza. Mientras que la construcción de sistemas de confianza resistentes a sabotajes es aun un reto real para las tecnologías existentes, las tendencias actuales del mercado sugieren que los sistemas abiertos y no confiables tales como PC y estaciones de trabajo serán los sistemas predominantemente usados en el acceso a documentos con derechos de autor. En este sentido, los ámbitos de computación existentes tales como los PC y las estaciones de trabajo equipadas con sistemas operativos populares (por ejemplo, Windows y UNIX) y aplicaciones ejecutables (por ejemplo, Microsoft Word) no son sistemas de confianza y no pueden convertirse en confiables sin alterar significativamente sus arquitecturas.

Consecuentemente, aunque pueden desarrollarse ciertos componentes de confianza, debemos continuar basándonos en sistemas diversos desconocidos y no confiables. Sobre tales sistemas, incluso si se espera que sean seguros, se encuentran y se utilizan frecuentemente errores y debilidades imprevistas.

En el contexto de la distribución de documentos se presenta un asunto particular, como se describe a continuación de modo general. En el modelo tradicional de distribución de documentos, el autor del contenido y el editor típicamente no manejan la distribución; una parte separada con expertos en distribución se encarga de esa responsabilidad. Además, mientras que es posible cifrar un documento (usando técnicas estándar) de modo que puedan descifrarlo múltiples receptores, no se sabe usualmente en el instante en que se crea el trabajo quienes será el usuario final. Tiene más sentido para el distribuidor determinar quienes serán los usuarios finales, y distribuirles el documento como deseen. Si, como en el modelo tradicional, el trabajo original del autor se envía al editor y el distribuidor sin cifrar, ese es el punto de vulnerabilidad para el trabajo.

Un problema similar se presenta en el escenario de oficina, por ejemplo, en el que es frecuentemente deseable designar lo que se llama de forma diversa un agente del documento, sustituto o delegado. En esta situación, es frecuentemente útil dar a un auxiliar administrativo o secretario el derecho de descifrar cierto documento no pretendido directamente para esa persona.

Considerando el problema más ampliamente, en ámbitos de red, se pasan frecuentemente mensajes a receptores distintos de los que inicialmente se pretende. Cuando la confidencialidad de los mensajes es una preocupación y se envían mensajes cifrados, es muy deseable permitir a uno descifrar esos mensajes en representación de otro. Para concretar, supongamos que Bob es quien necesita leer algún mensaje que está inicialmente cifrado para Alice. Una solución trivial es que Alice revele simplemente su clave de descifrado a Bob para que Bob pueda usarla para descifrar él mismo el mensaje. Esto requiere de Alice que confíe en Bob totalmente, lo cual puede no ser aceptable para Alice. Otro modo de cumplir este cometido es dejar que Alice primero descifre el mensaje, y luego lo re-cifre para Bob y finalmente envíe el mensaje nuevamente cifrado a Bob de modo que el pueda descifrarlo. Aunque el mensaje se comunica con seguridad, esta solución es menos eficiente ya que requiere dos operaciones de descifrado y una de cifrado para que Bob obtenga el mensaje. Más importante, es que en algunas situaciones tal solución de volver a cifrar no es incluso aplicable o deseable. Por ejemplo, Alice puede que no tenga acceso al mensaje cifrado ya que puede enviarse directamente por el remitente a Bob para eficiencia en la comunicación y otras consideraciones. También, el descifrado de un mensaje cifrado a una versión clara, incluso sólo por un corto periodo de tiempo, puede ser una vulnerabilidad sustancial.

Por consiguiente, sería deseable tener un entramado de cifrado/descifrado que soporte la posibilidad de transferir el derecho de decodificar los mensajes. Tal entramado permitiría una delegación, esencialmente, autorizar el re-cifrado de un mensaje para el uso por otra parte sin descifrar el mensaje original. Sería útil también para esto, que fuese posible sin que el delegado tuviese nunca el mensaje cifrado en su posesión.

Cómo transferir el derecho de descifrar desde un titular de clave a otro de modo seguro y eficiente es el objeto del cifrado de delegación. Recientemente se han propuesto esquemas de cifrado de delegación para convertir mensajes cifrados por una clave en mensajes cifrados para otro sin revelar al público el secreto de la claves de descifrado ni el mensaje original. Mambo y Okamoto han introducido varios esquemas de cifrado de delegación privados, no conmutativos, independientes del mensaje. Blaze y Strauss han introducido un esquema de cifrado de delegación, público, conmutativo, independiente del mensaje.

En esta divulgación, se dirige inicialmente el mismo problema general, pero en un contexto más general de esquemas de codificación. Los esquemas de codificación considerados en esta divulgación difieren de los esquemas de cifrado o sistemas de criptografía en que no tienen necesariamente ningún requisito relacionado con la seguridad. Para que un esquema de codificación sea un esquema de cifrado, es necesario que un curioso, mirando por encima un mensaje codificado sea incapaz de determinar ni el mensaje original ni la clave usada para decodificar el mensaje. Trabajar con esquemas de codificación hace posible construir aplicaciones seguridad ligera pero con una eficacia en la implementación elevada, tal como una eficiente distribución de documentos masiva y actualización del texto cifrado con nuevas claves para proteger mensajes cifrados a largo plazo. En esta divulgación, se define una clase de esquemas de codificación, y se dan varios ejemplos de esquemas. En este documento se ofrece también un proceso por el cual pueden construirse nuevos esquemas usando los esquemas existentes.

Se presentan a continuación varios esquemas más formales de cifrado de delegación. Un esquema de cifrado de delegación es un esquema de cifrado que permite a un titular designado de clave descifrar mensajes en representación de otro titular de clave. Esta divulgación introduce dos nuevos esquemas de cifrado de delegación basado en el conocido esquema de ElGamal, que mejora las funcionalidades sobre los esquemas de cifrado de delegación existentes. Son públicos en el sentido de que la información relativa a la delegación y las transformaciones pueden hacerse al público con seguridad, y al mismo tiempo no conmutativos en términos de relación de confianza entre los titulares de claves involucrados. También se presentan las aplicaciones de estos nuevos esquemas a la distribución masiva de documentos y la protección de ficheros.

La idea básica en los métodos presentes en esta divulgación es la que sigue: para que Alice transfiera el derecho de decodificar a Bob, Alice genera una clave de transferencia t para Bob. Con la clave de transferencia t, Bob puede cifrar de nuevo el mensaje inicialmente codificado para Alice y posteriormente descifrarlo usando su propia clave.

Como en el cifrado de delegación, la transferencia se realiza de tal modo que la clave de la transferencia no revela explícitamente las claves de decodificación ni de Alice ni de Bob, ni el mensaje original.

El objetivo del cifrado de delegación es cómo delegar el derecho de descifrar desde un titular de clave a otro de forma segura y eficiente. Muy recientemente, se han propuesto algunos esquemas de cifrado de delegación específicos para convertir mensajes cifrados con una clave en mensajes cifrados para otro sin revelar al público las claves secretas de descifrado ni el mensaje original. Manbo y Okamoto han descrito tres esquemas de cifrado de delegación para los esquemas de cifrado ElGamal y RSA. M. Mambo y E. Okamoto, "Proxy cryptosystems: Delegation of the power to decrypt ciphertexts", *IEICE Trans. on Fundamentals*, Vol. E80-A, Nº 1, pág. 54-63 (1997). Para la situación que se ha mencionado anteriormente, sus esquemas tienen mejor funcionamiento computacional sobre el esquema de re-cifrado, pero por razones de seguridad requiere la presencia del titular de clave original Alice en la conversión del mensaje. Además, los propios esquemas no ayudan a especificar quién es el titular de la clave al que Alice quiere delegar los derechos de descifrado. El esquema propuesto por Blaze y Strauss, por el contrario, no tienen estos defectos. Es una modificación del esquema de cifrado de ElGamal. M. Blaze y M. Strauss, "Proxy Cryptography", Borrador, AT&T Research Labs, ftp://ftp.research.att.com/dist/mab/proxy.ps de (Mayo de 1997). Una característica muy aparente del esquema de Blaze y Strauss es que permite comunicar la información relativa a la delegación y realizar la conversión del mensaje en público. Pero introduce un problema más serio: es conmutativo en el sentido de que Bob es capaz de obtener la clave de descifrado de Alice. Este tipo de conmutatividad convierte el esquema de descifrado de delegación en obsoleto, y todo el esquema puede simplificarse dando la clave de Alice a Bob y dejando a Bob descifrar. Otra cuestión (no necesariamente un problema) creada por este esquema es que una vez que se ha conferido a Bob la clave de descifrado por Alice, el puede descifrar todos los mensajes que originalmente son para Alice. Esta independencia del mensaje puede ser útil en algunos casos tales como una auto-delegación pero no es deseable en muchas aplicaciones prácticas en las que el titular de la clave original quiere ser selectivo o en qué mensajes está permitido el descifrado de delegación.

El objetivo de la presente invención es proporcionar un método mejorado para cifrar documentos para su distribución a los receptores seleccionados que soporta la capacidad de transferir el derecho de decodificar el documento.

El objetivo se resuelve por la cuestión sujeto de la reivindicación independiente 1.

Las realizaciones preferidas se definen por las cuestiones sujeto de las reivindicaciones dependientes.

Por consiguiente, los esquemas de cifrado de delegación de acuerdo con la presente invención que son públicos y no conmutativos, eliminan algunas desventajas de los sistemas de cifrado conocidos.

En esta divulgación, se introducen dos nuevos esquemas de cifrado. Están ambos basados en el esquema de cifrado de clave pública de ElGamal y tienen un funcionamiento computacional comparable. Esencialmente, mantienen las siguientes características deseables de los esquemas existentes: (i) público: no se requiere la presencia del titular de la clave original después de que se genera la información de delegación, y la información relacionada con la delegación y las operaciones pueden comunicarse y dirigirse en público; (2) no-conmutativo: los titulares de las claves no tiene que confiar cada uno en el otro en el cuidado de sus claves de cifrado privadas; y (iii) restringido: el titular de clave al cual se delega el derecho de descifrado se especifica, y la información de delegación (clave) es dependiente del mensaje.

Finalmente, se describe luego la delegación del derecho de descifrar mensajes en el contexto de los sistemas de cifrado de Cramer-Shoup, que conllevan algunas ventajas sobre otros sistemas.

Estas y otras características y ventajas de la presente invención son aparentes mediante las Figuras y la descripción completa en la Descripción Detallada de la Invención.

La Figura 1 es un diagrama de bloques de un sistema de distribución de documentos electrónico capaz de operar de acuerdo con la invención;

La Figura 2 es un diagrama de bloques que ilustra las operaciones de codificación realizadas cuando se delega la autoridad de descifrar un mensaje en un método útil para el entendimiento de la invención;

La Figura 3 es un diagrama de flujo que ilustra las etapas generales realizadas en la transformación de un mensaje codificado para su decodificación por otro;

La Figura 4 es un diagrama de bloques que ilustra esquemáticamente las partes involucradas en un sistema adaptado para la delegación de la autoridad para descifrar mensajes;

La Figura 5 es un diagrama de flujo que ilustra las etapas realizadas en un esquema de cifrado de delegación genérico;

La Figura 6 es un diagrama de flujo que ilustra las etapas realizadas en el cifrado y descifrado de un mensaje de acuerdo con los sistemas de criptografía de ElGamal;

La Figura 7 es un diagrama de flujo que ilustra las etapas realizadas en un esquema de cifrado y descifrado de delegación conocido basado en el ElGamal propuesto por Mambo y Okamoto;

La Figura 8 es un diagrama de flujo que ilustra las etapas realizadas en un esquema de cifrado y descifrado de delegación conocido basado en el ElGamal propuesto por Blaze y Strauss;

La Figura 9 es un diagrama de flujo que ilustra las etapas realizadas en una primera realización de un esquema de cifrado y descifrado de delegación basado en ElGamal útil para el entendimiento de la invención;

La Figura 10 es un diagrama de flujo que ilustra las etapas realizadas en una segunda realización de un esquema de cifrado y descifrado de delegación basado en ElGamal útil para el entendimiento de la invención;

La Figura 11 es un diagrama de flujo que ilustra las etapas realizadas en un esquema de distribución de documentos de acuerdo con la invención;

La Figura 12 es un diagrama de flujo que ilustra las etapas realizadas en un esquema de protección de ficheros útil para el entendimiento de la invención;

La Figura 13 es un diagrama de flujo que ilustra las etapas realizadas en el cifrado y descifrado de un mensaje de acuerdo con el sistema de criptografía de Cramer-Shoup; y

La Figura 14 es un diagrama de flujo que ilustra las etapas realizadas en una realización de un esquema de cifrado y descifrado de delegación basado en Cramer-Shoup de acuerdo con la invención;

Las Figuras se explican más enteramente en la siguiente Descripción Detallada de la Invención.

La invención se describe a continuación, con referencia a las realizaciones ilustrativas detalladas. Será aparente que la invención puede realizarse de una amplia variedad de formas, algunas de las cuales puede ser bastante diferente de las de las realizaciones reveladas.

Por consiguiente, los detalles de la estructura específica y funcional revelados en este documento son meramente representativos y no limitan el alcance de la invención.

La Figura 1 representa un modelo funcional de máximo nivel para un sistema de distribución electrónica de documentos, que como se ha definido anteriormente, puede incluir correspondencia, libros, revistas, diarios, periódicos, otros documentos escritos, software, secuencias de audio y vídeo, y otras presentaciones multimedia.

Un autor (o editor) 110 crea un contenido original del documento 112 y lo pasa al distribuidor 114 para su distribución. Aunque se contempla que el autor puede distribuir también los documentos directamente, sin involucrar a otra parte como el editor, la división de tareas mostrada en la Figura 1 es más eficiente, ya que permite al autor/editor 110 concentrarse en la creación de contenidos, y no las funciones mecánicas y mundanas tomadas por el distribuidor 114. Además, tal descomposición permite al distribuidor 114 realizar economías de escala por asociación con un número de autores y editores (incluyendo el autor/editor ilustrado 110).

El distribuidor 114 pasa a continuación el contenido modificado 116 a un usuario 118. En un modelo de distribución electrónica típica, el contenido modificado 116 representa una versión re-cifrada del contenido original cifrado 112; el distribuidor 114 primero descifra el contenido original 112 y luego lo cifra de nuevo con la clave pública del usuario 118; ese contenido modificado 112 se hace a medida solamente para el usuario único 118. El usuario 118 está capacitado para usar su clave privada para descifrar el contenido modificado 116 y ver el contenido original 112.

El pago 120 por el contenido 112 se pasa desde el usuario 118 al distribuidor 114 por medio de un banco de liquidación 122. El banco de liquidación 122 colecta peticiones desde el usuario 118 y desde otros usuarios que quieren ver un documento en particular. El banco de liquidación 122 también colecta información del pago, tal como transacciones de débito, transacciones con tarjeta de crédito, u otros esquemas de pago electrónico conocidos, y conducen los pagos colectados de los usuarios como una hornada de pagos 124 al distribuidor 114. Por supuesto, se espera que el banco de liquidación retenga una participación del pago del usuario 120. A su vez, el distribuidor 114 retiene una parte de la hornada de pagos 124 y dirige el pago 126 (incluyendo los pagos por derechos de autor) al autor y editor 110. En una realización de este esquema, el distribuidor 114 espera un manojo de peticiones de usuario por un documento único antes de enviar ninguno. Cuando se hace esto, puede generarse un único documento con el contenido modificado 116 para descifrado por todos los usuarios peticionarios. Esta técnica es bien conocida en la técnica.

Entretanto, cada vez que el usuario 118 pide (o usa) un documento, se envía un mensaje de contabilidad a un servidor auditor 130. El servidor auditor 130 asegura que cada petición por el usuario 118 coincide con un documento enviado por el distribuidor 114; la información de contabilidad 131 se recibe por el servidor auditor directamente desde el distribuidor 114. Cualesquiera contradicciones se transmiten al banco de liquidación 122 mediante un informe 132, que pueden luego ajustar las hornadas de pagos 124 hechas al distribuidor 114. Este esquema de contabilidad está presente para reducir la posibilidad de fraude en este modelo de distribución electrónica de documentos, así como para manejar cualesquiera permisos de uso dependientes del tiempo que pueden resultar en cargas que varían, dependiendo de la duración u otra extensión de uso.

El modelo anterior para el comercio electrónico de documentos, mostrado en la Figura 1, se usa comúnmente hoy. Como se describirá en detalle más adelante, es igualmente aplicable al sistema y al método mostrado en este documento para la distribución de documentos autoprotegidos.

Esquemas de Codificación Delegados

Por simplicidad, inicialmente consideramos esquemas de codificación del siguiente tipo. Un sistema de codificación consiste de cuatro componentes: (i) un espacio de mensajes X que es una colección de los posibles mensajes, (ii) un espacio de claves K que es un conjunto de claves posibles, (iii) una transformación de codificación eficiente desde el punto de vista computacional E: KxX → X y (iv) una transformación de decodificación eficiente desde el punto de vista computacional D: KxX → X. Para cada k ∈ K, la transformación de codificación E_k: X → X y cada transformación de decodificación D_k: X → X son aplicaciones inyectivas (uno a uno) sobre X, y satisfacen que, por cada mensaje x ∈ X,

$$D_k\ (E_k(x)) = x.$$

Ciertamente, tales esquemas de codificación pueden variarse de varios modos para cubrir un rango más amplio de casos. Uno es diferenciar el espacio de mensajes codificados del espacio de mensajes original, y otro es considerar que las claves usadas para codificar y decodificar son diferentes. En términos de criptografía, los esquemas de codificación considerados a continuación son esencialmente claves privadas (o, más precisamente, simétricas), sistemas de criptografía endomórficos.

Tales sistemas de codificación definidos tienen algunas propiedades ventajosas. Dado un esquema de codificación (X, K, E, D), cada transformación de codificación y su correspondiente transformación de decodificación son transformaciones inversas entre sí; esto es, para cada k∈K,

$$D_k = (E_k)^{-1} \qquad \text{y} \qquad E_k = (D_k)^{-1}$$

Si X es un conjunto finito, cada transformación de codificación y decodificación es sólo una permutación sobre X.

Clásicos, esquemas de cifrado con clave simétrica son esquemas de codificación. Aquí están algunos de ellos.

Esquema XOR, X. En este esquema, el espacio de mensajes X es el conjunto B_n de todas las cadenas binarias de n-bit para un entero n > 0, y tal es el espacio de claves K. El número de posibles mensajes y el número de posibles claves son ambos 2^n. Para cada mensaje x y cada clave k, la codificación es

$$y = E_k(x) = x \oplus k$$

y la decodificación del mensaje y es

$$x = D_k(y) = y \oplus k;$$

donde ⊕ representa la operación XOR orientada al bit (or exclusiva).

Esquema Multiplicativo M. Un mensaje en este esquema es un elemento en $X = Z_n = \{0,1, \ldots n-1\}$ para algún número entero n > 0. Una clave es también un elemento a de Zn pero que satisface gcd (a,n) = 1, donde "gcd" es la función que especifica el máximo común divisor entero de los dos argumentos. Esto es, el espacio de llaves K consiste en los elementos en el grupo multiplicativo Zn = {a ∈ Zn | gcd(a,n) = 1}. La codificación de un mensaje x con una clave a es

$$y = E_a(x) = ax(\text{mod } n)$$

y la decodificación de un mensaje y con una clave a es

$$x = D_a(y) = a^{-1}\ y\ (\text{mod } n),$$

donde a^{-1} es la inversa multiplicativa de a en módulo n; es decir, a^{-1} es un elemento en Zn tal que aa^{-1} (mod n) = a^{-1} a(mod n) = 1. Obsérvese que la condición sobre a, gcd (a,n) = 1, se usa para garantizar que a tiene la inversa a^{-1}. Es sabido que el número de tales a es igual a la función ϕ de Euler

$$\phi(n) = \prod_{i=1}^{n}(p_i^{ei} - p_i^{ei-1})$$

donde

$$n = \prod_{i=1}^{n} p_i^{ei}$$

es la descomposición en números primos de n. De modo que el número de claves en el esquema M es ϕ (n).

Esquema de desplazamiento S. Los mensajes y las claves en el esquema de desplazamiento son elementos de Z_n = {0,1, ..., n-1} para cualquier número entero n > 0; esto es, X = K = Zn. De este modo, el número de mensajes y el número de claves en el esquema de desplazamiento son ambos iguales a n. Para codificar un mensaje x con una clave b, se calcula

$$y = E_b(x) = x + b(\text{mod } n)$$

y para decodificar un mensaje con b, se computa

$$x = D_b(y) = y - b(\text{mod } n).$$

Esquema de Sustitución P. Este esquema se define también sobre $X = Z_n$. No obstante, el espacio de claves $K = \Pi_n$ consiste en todas las permutaciones de los elementos en Z_n. De este modo, el número total de claves es n!. Por cada permutación $p \in \Pi_n$, la codificación es

$$y = E_p(x) = p(x),$$

y la decodificación es

$$x = D_p(y) = p^{-1}(y),$$

donde p^{-1} es la permutación inversa de p.

Debe observarse que los esquemas multiplicativos y de desplazamiento son casos especiales del esquema de sustitución que incluye sólo ϕ(n) y n respectivamente de las n ! posibles permutaciones de los n elementos.

Los nuevos esquemas de codificación se construyen por combinación de los existentes. Un modo es formar su "producto". Supongamos que S y S' son dos esquemas de codificación con el mismo espacio de mensajes X. El producto de S y S', denominado como SxS', tiene el mismo espacio de mensajes X. Una clave del esquema producto tiene la forma (k, k'), donde k y k' son las claves de S y S' respectivamente. Las transformaciones de codificación y decodificación del esquema producto se definen como sigue: para cada clave $(k, k') \in K$,

$$E_{(k,k')}(x) = E_{k'}(E_k(x))$$

y

$$D_{(k,k')}(x) = D_k(D'_{k'}(c))$$

Esto es, el mensaje x primero se codifica con E_k, y el mensaje resultante se "re-codifica" a continuación con $E_{k'}$. La decodificación es similar, pero se hace en el orden inverso.

Es directo comprobar que la construcción de producto es siempre asociativa; (SxS')xS" = Sx(S'xS"). Si un esquema de codificación S se toma para formar el producto con el mismo, se obtiene el esquema SxS, denominado S^2. Si se toma el producto enésimo, el esquema resultante, denominado S^n, se llama un esquema de codificación iterado.

Un simple ejemplo para ilustrar la definición de producto de esquemas de codificación es como sigue.

Esquema Afín A. Este esquema se define también sobre X = Zn. Una clave de un esquema afín es un par de números enteros (a,b) en Zn, donde gcd(a,n) =1. La transformación de codificación es

$$y = E_{(a,b)}(x) = (ax + b)(\text{mod } n)$$

y la transformación de decodificación es

$$x = D_{(a,b)} = a^{-1}(y - b)(\text{mod } n)$$

donde a^{-1} es la inversa modular de a en módulo n. Estas transformaciones del tipo ax + b se llaman usualmente transformaciones afines, de aquí el nombre de esquema afín. Obsérvese que el esquema A se reduce al esquema multiplicativo cuando b = 0 y al esquema de desplazamiento S cuando a = 1. De este modo, M y S son casos especiales de A. Por el contrario, A es su producto M x S. Como se ha visto anteriormente, una clave en el esquema multiplicativo es un elemento $a \in Z_n$'; la correspondiente transformación de codificación es $E_a(x) = ax \pmod n$. Una clave en el esquema de desplazamiento es un elemento $b \in Zn$, y la correspondiente transformación de codificación es $E_b(x) = x + b \pmod n$. De aquí que, una clave en el esquema producto M x S tiene la forma $(a,b) \in Z_{n'} \times Z_n$, y su codificación es

$$E_{(a,b)}(x) = E_b(E_a(x)) = ax + b \pmod n$$

Esta es precisamente la definición de la transformación de codificación en el esquema afín. Similarmente, la transformación de decodificación en el esquema afín es la composición de las transformaciones de decodificación de los esquemas de desplazamiento y multiplicativo.

El objetivo de transferir el derecho a decodificar mensajes en cualquier esquema de codificación dado (X, K, E, D) puede establecerse como sigue: para cualquier mensaje dado $x \in X$ y claves k, k' $\in$ K, convertir de un modo eficiente el mensaje codificado $y = E_k(x)$ usando la clave k en el mensaje codificado $y' = E_{k'}(x)$ usando la clave k' de modo que el mensaje y' puede decodificarse correctamente usando la clave k'. Si esto puede conseguirse, se dice que el derecho de decodificar el mensaje y se ha transferido o delegado desde el titular de la clave k al titular de la clave k'.

La Figura 2 ilustra la transformación π 210 que se necesita para conseguir el objetivo. Las líneas finas 212, 214, y 216 representan transformaciones E_k, π, y D_k, respectivamente, a partir de una secuencia de etapas que codifica un mensaje x con una clave k, convierte el mensaje codificado en el otro codificado con la otra clave k' y decodifica el mensaje usando la llave k'. Las líneas finas 218 y 220, representan las transformaciones E_k y D_k, respectivamente, muestran otras posibles operaciones de codificación y decodificación que pueden realizarse.

En muchos casos, el espacio de claves K de un esquema de codificación no es simplemente un conjunto. Equipada con alguna operación " . ", K puede poseer una estructura matemática. Por ejemplo, los espacios de claves de todos los esquemas de ejemplos dados en la sección previa pueden equiparse con algunas operaciones para convertirse en grupos matemáticos. La tabla 1, a continuación muestra algunas de estas operaciones, donde º permanece para el operador de composición de permutaciones y

$$* : (Z_n' \times Z_n) \times (Z_n' \times Z_n) \rightarrow Z_n' \times Z_n$$

se define como

$$(a, b) * (a', b') = (a'a \pmod n, a'b + b' \pmod n).$$

TABLA 1

Esquema	Espacio de Claves "k"	Operación " . "
$\oplus$ X	B_n	(XOR)
M	Z'_n	x (mod n)
S	Z_n	+ (mod n)
P	Π_n	º (composición)
A	$Z_n \times Z_n$	*(definida anteriormente)

Cuando el espacio de claves K de un esquema de codificación (X, K, E, D) es un grupo con alguna operación " . ", las transformaciones de codificación y decodificación pueden determinarse únicamente por las claves. Esto pasa cuando el espacio de claves es isomorfo, como un grupo, a los grupos de transformación $E = \{E_k \mid k \in K\}$ y $D = \{D_k \mid k \in K\}$ formados por las transformaciones de codificación y decodificación sobre el espacio de mensajes X; que es, para cualquier k, k' $\in$ K,

$$D_k = (E_k)^{-1} = E_k - 1 \qquad \text{y} \qquad E_k{}^{o}E_{k'} = E_{k.k'}.$$

y

$$E_k = (D_k)^{-1} = D_k - 1 \qquad y \qquad D_k{}^{o}D_{k'} = D_{kk'},$$

donde º es el operador composición de la transformación que se define como, por ejemplo,

$$E_k{}^{o}E_{k'}(x) = E_{k'}(E_k(x))$$

para todo $x \in X$.

Puede comprobarse fácilmente que todos los esquemas dados en la Tabla 1 anterior se determinan por claves. Los esquemas de codificación determinados por clave permiten de modo sistemático transferir el derecho de decodificar mensajes desde un titular de clave a otro. Con el isomorfismo entre el espacio de claves y los grupos de transformación, la composición de la transformación de decodificación con una clave k y la transformación de codificación con otra llave k' puede luego verse como la transformación de codificación determinada con la clave compuesta $k^{-1}.k$. Sea (X, K, E, D) un esquema de codificación determinado por clave. Supongamos que $y = E_k(x)$ es la versión codificada del mensaje $x \in X$ con la clave $k \in K$. El derecho a decodificar el mensaje codificado de x puede transferirse desde el titular de la clave k al titular de la clave k' en el algoritmo de dos etapas mostrado en la Figura 3.

En primer lugar, genera una clave de transferencia $t = k^{-1}.k$ (etapa 310). A continuación codifica el mensaje con la clave t de acuerdo a $y' = E_t(y)$ (etapa 312),

El algoritmo es correcto gracias a la propiedad de que el espacio de claves es isomorfo a los grupos de transformación de codificación y decodificación. La corrección puede verificarse como sigue:

$$\begin{aligned} D_{k'}(y') &= D_{k'}(E_t(y)) \\ &= D_{k'}(E_k\text{-}1_{.k'}(y)) \\ &= D_{k'}(E_{k'}(E_k\text{-}1(y))) \\ &= E_k\text{-}1(y) \\ &= D_k(y) \\ &= D_k(E_k(x)) \\ &= x \end{aligned}$$

La generalización del algoritmo lo hace inmediato para derivar las etapas de transferencia para los esquemas de ejemplo que se han mostrado anteriormente. Refiriéndonos a la Figura 3, para el Esquema X, XOR sobre B_n, para convertir $y = E_k(x)$ a $y' = E_{k'}(x)$, primero genera una clave de transferencia $t = k^{\oplus} k'$ (etapa 310). Luego codifica el mensaje con la clave de transferencia t de acuerdo con $y' = y^{\oplus} t$ (etapa 312).

Para el Esquema Multiplicativo M sobre Z_n, para convertir $y = E_a(x)$ a $y' = E_{a'}(x)$, primero genera una clave de transferencia $t = a'a^{-1} \pmod n$ (etapa 310). A continuación codifica el mensaje con la clave de transferencia t de acuerdo con $y' = y.t \pmod n$ (etapa 312).

Para el Esquema de Desplazamiento S sobre Z_n, para convertir $y = E_b(x)$ a $y' = E_{b'}(x)$, primero genera una clave de transferencia $t = b' - b \pmod n$ (etapa 310). A continuación codifica el mensaje con la clave de transferencia t de acuerdo a $y' = y + t \pmod n$ (etapa 312).

Para el Esquema de Sustitución P sobre Π_n, para convertir $y = E_p(x)$ a $y' = E'_{p'}(x)$, primero genera una clave de transferencia $t = p^{-1}{}^{o} p'$ (etapa 310). A continuación codifica el mensaje con la clave t de acuerdo a $y' = t(y)$ (etapa 312).

Como se ha descrito anteriormente, es también posible transferir el derecho a decodificar en esquemas producto de no sólo codificaciones determinadas por la clave sino también esquemas conmutativos. Para definir los esquemas conmutativos, es necesario caracterizar los esquemas de codificación que son esencialmente equivalentes. Supongamos que S = (X, K, E, D) y S' = (X, K', E', D') son dos esquemas de codificación con el mismo espacio de mensajes X. Se dice que S es equivalente de S', y se indica $S \equiv S'$, si hay una aplicación biyectiva (uno a uno y sobre) $h: K \rightarrow K'$ tal que para cada mensaje $x \in X$ y cada clave $k \in K$,

$$E_k(x) = E'_{h(k)}(x)$$

y

$$D_k(x) = D'_{h(k)}(x).$$

Claramente, la relación equivalencia de esquemas ≡ es una relación de equivalencia; esto es, satisface que, para cualesquiera esquemas de codificación S, S', S", se mantiene lo siguiente: (i) S ≡ S; (ii) S ≡ S' implica que S' ≡ S; y (iii) S ≡S' y S' ≡ S" implica que S ≡ S". De este modo, los esquemas de codificación equivalentes forman una clase de equivalencia en la cual cada esquema en la clase no proporciona ninguna funcionalidad más ni ninguna menos que cualquier otra de la clase.

La relación de equivalencia de esquemas permite caracterizar los esquemas de codificación de varios modos. Se dice que un esquema de codificación S es idempotente si $S^2 \equiv S$. Muchos de los esquemas de codificación son idempotentes, incluyendo los esquemas XOR, multiplicativo, desplazamiento, sustitución, y afín. Si un esquema S es idempotente, entonces no hay motivo para usar el producto esquema S^2, ya que requiere una clave extra y no proporciona más funcionalidad.

Otra caracterización sobre los esquemas de codificación que usan la relación de equivalencia de esquemas ≡ es la de esquemas conmutativos. Se dice que dos esquemas de codificación S y S' conmutan si S x S' ≡ S' x S. Trivialmente, cualquier esquema conmuta consigo mismo. Un ejemplo no tan trivial es que el esquema multiplicativo M y el esquema de desplazamiento S conmutan. Para ver que conmutan, es decir M x S ≡ S x M, podemos comparar las ecuaciones

$$E_b(E_a(x) = ax + b(\text{mod } n)$$

y

$$E_a(E_b(x)) = ax + ab(\text{mod } n);$$

y encontrar la aplicación

$$h : K_S x K_M \rightarrow K_M x K_S$$

definido por

$$h(b, a) = (a, a^{-1}b(\text{mod } n))$$

que hace el producto S x M isomorfo con el producto M x S.

Los esquemas de producto de esquemas de codificación determinados por clave y conmutativos disfrutan de un modo sistemático de transferir el derecho de decodificar mensajes. Sea el producto S_1 x S_2 el esquema de producto de dos esquemas determinados por clave y conmutativos. Supongamos que $h = (h_1,h_2) : K_2 \text{ x } K_1 \rightarrow K_1 \text{ x } K_2$ es la aplicación que hace a S_2 x S_1 isomorfo con S_1 x S_2, donde $h_1 : K_2 \text{ x } K_1 \rightarrow K_1$ y $h_2: K_2 \text{ x } K_1 \rightarrow K_2$. En primer lugar, obsérvese que el esquema producto es también determinado por clave; el espacio de claves producto K_1 X K_2 es un grupo con respecto a la operación * definida por

$$(k_1, k_2) * (k_1', k_2') = (k_1.h_1(k_2, k_1'), h_2(k_2, k_1').k_2')$$

Esto es así porque

$$\begin{aligned} E_{(k1,k2)} \circ E_{(k1',k2')} &= E_{k1} \circ E_{k2} \circ E_{k1'} \circ E_{k2'} \\ &= E_{k1} \circ E_{h1(k2.k1')} \circ E_{h2(k2.k1')} \circ E_{k2'} \\ &= E_{k1.h1(k2.k1')} \circ E_{h2(k2.k1')k2'} \\ &= E_{(k1.k2)(k'1.k2')} \end{aligned}$$

Ahora, el derecho a decodificar el mensaje codificado x puede transferirse desde el titular de la clave k al titular de la clave k' en el algoritmo de dos pasos mostrado en la Figura 3. En primer lugar, generar una clave de transferencia $t = (h_1(k_2^{-1},k_1^{-1}.k_1'),h_2(k_2^{-1},k_1^{-1}.k').k_2')$ (etapa 310). A continuación codificar el mensaje con la clave de transferencia t de acuerdo con $y' = E_t(y)$ (etapa 312).

La corrección del algoritmo de transferencia se verifica por la siguiente igualdad:

$$\begin{aligned} E_t(y) &= E_{h1(k2^{-1},k1^{-1}.k1')} \circ E_{h2(k2^{-1},k1^{-1}-k1').k2'}\ (y) \\ &= E_{h1(k2^{-1},k1^{-1}.k1')} \circ E_{h2(k2^{-1},k1^{-1}.k1')} \circ E_{k2'}\ (y) \\ &= E_{k2}{}^{-1} \circ E_{k1}{}^{-1}{}_{.k1'} \circ E_{k2'}\ (y) \\ &= E_{k2}{}^{-1} \circ E_{k1}{}^{-1} \circ E_{k1'} \circ E_{k2'}\ (y) \\ &= D_{k2} \circ D_{k1} \circ E_{k1'} \circ E_{k2'}\ (y) \\ &= E_{k1'} \circ E_{k2'}\ (x) \\ &= E_{(k1'.k2')}\ (x) \end{aligned}$$

donde la última entidad puede decodificarse fácilmente usando la clave $k' = (k_1', k_2')$.

El método se ilustra mejor con el siguiente ejemplo, aplicado al cifrado afín A sobre Z_n. Ya que A = M x S, y M y S son esquemas determinados por clave, conmutativos, el método descrito anteriormente se aplica al esquema afín. Como se ha visto anteriormente, es la aplicación h(b,a) = (a,ab) la que hace S x M isomorfa con la M x S. De este modo, $h_1(b,a) = a$ y $h_2(a,b) = ab \pmod n$. La clave de transferencia t desde (a,b) a (a',b') puede derivarse como

$$\begin{aligned} t &= (h_1(b^{-1},a^{-1}.a'),h_2(b^{-1},a^{-1}.a').b') \\ &= (a'.a^{-1},h_2(b^{-1},a^{-1}.a') + b') \\ &= (a'.a^{-1},(a'.a^{-1})b^{-1} + b') \\ &= (a'a^{-1},-a'a^{-1}b + b') \end{aligned}$$

Entonces, para decodificar y usando una segunda clave (a',b'), en primer lugar se genera una clave $t = (a'a^{-1} \pmod n, -a'a^{-1}b + b' \pmod n) = (t_1,t_2)$ (etapa 310). Luego se codifica el mensaje usando la clave t de acuerdo con $y' = t_1\, y + t_2 \pmod n$ (etapa 312).

Los métodos presentados en este documento para transferir el derecho a decodificar mensajes son *transitivos*. Esto significa que dos transferencias secuenciales desde Alice a Bob y a continuación desde Bob a Carol son equivalentes a una transferencia directa desde Alice a Carol. Es importante observar que, en cada uno de los esquemas de ejemplo, una clave de transferencia es también una clave del esquema.

Por consiguiente, dos claves de transferencia usadas en dos transferencias secuenciales pueden combinarse para formar una clave de transferencia para la transferencia directa. Tomamos el esquema afín como ejemplo. Sean k = (a,b), k' = (a',b'), y k'' = (a'',b'') las claves para Alice, Bob, y Carol, respectivamente. Entonces las claves de transferencia son $t = (a'a^{-1}, -a'a^{-1}b + b')$ desde Alice a Bob, $t' = (a'',a'^{-1}, -a''a'^{-1}b' + b'')$ desde Bob a Carol, y $t'' = (a''a'^{-1},-a''a'^{-1}b' + b'')$ desde Alice a Carol. Es directo verificar que la composición de t y t' como claves en el esquema afín produce t'':

$$\begin{aligned} t.t' &= (t_1't_1,t_1't_2 + t_2') \\ &= ((a''a'^{-1})(a'a^{-1}),(a''a'^{-1})(-a'a^{-1}b + b') + (-a''a'^{-1}b') + (a''a'^{-1}b' + b'')) \\ &= (a''a^{-1},-a''a^{-1}b + b'') \\ &= t'' \end{aligned}$$

En otras palabras, la composición de transferencias secuenciales de los derechos de decodificar mensaje no tiene memoria; todas las transferencias intermedias no se reflejarán en la transferencia global.

Debe observarse que, para los esquemas X, M y S, la etapa de generación de la clave de transferencia es equivalente a "decodificar" k' con k. De este modo, la computación necesaria en la transferencia es la misma que se usa en el método de decodificar y re-codificar para estos esquemas. Se puede pensar que el nuevo método no muestra mejora en esta consideración por la eficacia, pero se ha encontrado que la clave del mensaje es independiente del mensaje y por tanto no necesita computarse más que una vez. Cuando el número de mensajes m involucrado en la transferencia aumenta, esta característica recortará la computación requerida por el método de la re-codificación a la mitad. Además, la clave de transferencia t no filtra ninguna información útil sobre las claves k y k', y una transferencia realizada de acuerdo con los métodos mostrados en este documento no revelará el mensaje x. Estas propiedades hacen que el método propuesto interese cuando la seguridad del mensaje x y de las claves de decodificación k y k' durante una transferencia es el tema.

Una configuración de sistema típica capaz de cumplir los métodos descritos con referencia a la Figura 3 (y descrita con mayor detalle a continuación) se muestra en la Figura 4. Hay tres partes relevantes en la mayor parte de las aplicaciones de cifrado de delegación. Un Cifrador 410, un Cesionista A 412, y un Cesionario B 414. Como se reconocerá, el cifrado, el descifrado, y otras operaciones de proceso realizadas en la invención se facilitan por un procesador (416, 418, 420) bajo cada parte de control. Cada procesador se equipa con memoria (422, 424, 426) para almacenamiento de datos y una interfaz de comunicación (428, 430, 432), capaz de enviar y recibir mensajes.

Esquemas de Cifrado de delegación

El resto de la divulgación, dirigida a los más formales esquemas de cifrado de delegación más que a los esquemas de codificación, se organiza como sigue. En primer lugar, se describe un esquema de cifrado de delegación genérico y se caracteriza de acuerdo a varios criterios. Varios párrafos siguientes muestran lo que se usará mediante la divulgación y el recuerdo del esquema de cifrado de clave pública de ElGamal. Con el propósito de comparación, esta divulgación lista a continuación los dos esquemas de cifrado de delegación existentes y examina sus propiedades en comparación con la presente invención. A continuación se introducen detalles sobre los dos nuevos esquemas de cifrado de delegación, junto con su análisis de seguridad y funcionamiento. Después se dan aplicaciones a estos nuevos esquemas a la distribución masiva de documentos y la protección de ficheros.

Como se ha indicado en la introducción, el objetivo del cifrado de delegación es delegar el derecho de descifrado desde una persona a otra de modo eficiente y seguro. Para la discusión que sigue, es conveniente definir los papeles de las partes que pueden estar involucrados en el cifrado de delegación. Los dos papeles más importantes son los de cesionista y cesionario. El *cesionista* es el titular original de la clave de los mensajes cifrados que quiere delegar el derecho de descifrar a algún otro. Un *cesionario* es el titular de clave designado para realizar el descifrado en representación del cesionista y actuar de ese modo como delegado de descifrado del cesionista. En el ejemplo que motivaba la introducción, Alice es la cesionista y Bob es el cesionario. Otros papeles pueden incluir un *cifrador* que es quien cifra originalmente los mensajes para el cesionista, y el *facilitador* que es el que ayuda a realizar algunas tareas de procesamiento del mensaje, tales como la transformación de los mensajes cifrados para el cesionista en mensajes cifrados para el cesionario. Ciertamente, no es necesario que estos papeles se realicen por partes separadas. Por ejemplo, una parte puede hacer los papeles de cesionista y facilitador, como en los esquemas de Mambo y Okamoto que se discuten a continuación.

Con estos papeles situados, un esquema de cifrado de delegación es sólo una descripción de cómo un cesionista, posiblemente con ayuda de un facilitador, delega a un cesionario el derecho de descifrar mensajes originalmente generados por el cifrador para el cesionista. Un esquema de cifrado de delegación puede consistir de cuatro pasos genéricos: cifrado del mensaje, generación de la clave de delegación, transformación de delegación, y descifrado del mensaje. Estas etapas se describirán con mayor detalle a continuación, con referencia a la Figura 5.

1. Cifrado del mensaje E: El cifrador genera un mensaje cifrado usando la clave de cifrado del cesionista y lo suministra al cesionista (etapa 510).

2. Generación delegada π: Para delegar el derecho de descifrado al cesionario, el cesionista genera una clave de delegación π como una señal de compromiso que permite al cesionario descifrar el mensaje cifrado por el cesionista.

3. Transformación delegada Π: Cuando es necesario, el facilitador realiza una transformación delegada Π, posiblemente usando la clave de delegación π, para convertir el mensaje cifrado para el cesionista en un mensaje cifrado para el cesionario (etapa 514).

4. Descifrado del mensaje D: Una vez que se recibe el mensaje transformado y posiblemente la clave de delegación π, el cesionario descifra el mensaje (etapa 516),

Por consiguiente, debe observarse que el esquema genérico anterior cubre las dos soluciones directas para el cifrado de delegación mencionadas en la introducción. El esquema de re-encriptación es un caso especial en el cual el cesionista (Alice) es también el facilitador quien actualmente descifra el mensaje y a continuación lo cifra para el cesionario (Bob), y la delegación π puede considerarse como una colección de las claves de descifrado del cesionista y la clave de cifrado del cesionario, que se usa sólo por el cesionista y no por el cesionario. El esquema de paso de la clave de descifrado del cesionista al cesionario es otro caso especial del esquema genérico, en el cual la clave delegada es la clave de descifrado y la transformación delegada es la transformación identidad.

No obstante, no todos los esquemas que pueden derivarse del caso genérico anterior se califican de esquemas de cifrado de delegación. Intuitivamente, un esquema de cifrado de delegación tiene que satisfacer algunos requisitos básicos, a saber, seguridad, transitividad y ejecución, como se describe a continuación.

Delegación. Para asegurar que, al final de la etapa de descifrado del mensaje, el cesionario es capaz de recuperar el mensaje original correctamente, cualquier mensaje m debe sujetarse a la siguiente ecuación:

$$D(\Pi(E(m, e_A), \pi), d_B, \pi) = m,$$

donde E(m,e) es la función de cifrado del mensaje m bajo la clave de cifrado e, D(c,d,π) es la correspondiente función de descifrado del mensaje cifrado c bajo la clave de descifrado d y posiblemente la clave de delegación π, Π(c,π) es la función de delegación que convierte el mensaje cifrado c de acuerdo con la clave de delegación π, y e_A, e_B, d_A y d_B son la claves de cifrado y descifrado del cesionista A y el cesionario B, respectivamente.

Además de corrección anterior, la funcionalidad de la delegación debe garantizarse. De algún modo, esto significa que, después de que se utiliza la clave de delegación y se ha completado la transformación, la etapa de descifrado del mensaje no debe requerir información privada desde el cesionista, y debe realizarse solamente por el cesionario. De otra forma, esto es equivalente a la que la delegación desde el cesionista es innegable; esto es, una vez que se crea la clave de delegación y se realiza la transformación de delegación, el cesionista no debe ser capaz de denegar la delegación, sin buscar otros medios tales como impedir que el cesionario obtenga la clave de delegación y reciba el mensaje transformado. Como consecuencia de esta funcionalidad, la clave de descifrado del cesionista puede destruirse con la clave de descifrado del cesionario y posiblemente la clave de delegación mantiene la capacidad de descifrar el mensaje. (Esto es útil en la aplicación de protección de ficheros posterior en la Sección 6).

Seguridad. En esencia, un esquema de cifrado de delegación es también un esquema de cifrado al menos desde el punto de vista del cesionario. La introducción de las claves de delegación y las transformaciones no debe comprometer la seguridad y la privacidad de la encriptación en modo alguno. De este modo, debe ser difícil al menos desde el punto de vista computacional para una tercera parte no autorizada recuperar el mensaje original y las claves de descifrado del cesionista y cesionario a partir de la información disponible públicamente.

Además, debe ser difícil la falsificación de claves de delegación válidas por cualquier parte intrusa. Debe estar claro, sin embargo, que la generación de la clave de delegación π requiere el conocimiento de al menos la clave de descifrado del cesionista; de lo contrario el sistema de encriptación subyacente no es seguro.

Transitividad. Naturalmente, la relación de delegación debe ser transitiva. Después de que el cesionista delega el derecho de descifrado, el cesionario debe ser capaz de actuar como un nuevo cesionista para delegar el derecho por su parte a otro cesionario, sólo siguiendo el mismo esquema. Además, debe ser posible para alguien, decir al primer cesionista, que delegue el derecho directamente al nuevo cesionario por combinación de todas las claves de delegación intermedias en una clave de delegación y componer todas las transformaciones de delegación en una transformación única.

Funcionamieto. Como el esquema de re-cifrado es una solución intuitiva, directa para el cifrado de delegación y satisface la delegación anterior, requisitos de seguridad y transitividad, cualquier esquema de cifrado de delegación útil en la práctica no debe tener degradación en el funcionamiento computacional cuando se compara con el esquema de re-encriptación.

Los esquemas de cifrado de delegación pueden variar de acuerdo a sus requisitos de aplicación. Pueden ser establecidos en categorías de acuerdo a muchos aspectos. Los obvios incluyen si están basados en clave pública o en clave privada, y si sus medidas de seguridad son perfectas en el sentido teórico de la información o se basan en la intratabilidad de algunos problemas computacionales. Los siguientes aspectos están relacionados con las claves de delegación y transformación.

Confidencialidad. Mientras el anonimato de los mensajes y las claves de descifrado tiene que cumplirse, el secreto de la claves de delegación y las transformaciones de delegación no es un requisito obligatorio. Un esquema se llama público si las claves de delegación que se generan pueden publicarse sin comprometer su seguridad y las transformaciones de delegación aplicadas en ámbitos de deslealtad; por el contrario, el esquema es privado. En un esquema privado, cuando la clave de delegación se transfiere desde el cesionista al facilitador y el cesionario, debe tenerse cuidado de proteger la clave de delegación de la divulgación. Como resultado, la transformación de delegación que usa una clave de delegación debe realizarse también en privado.

Conmutatividad. En termino de mensajes, el cesionario debe estar incondicionalmente confiado en el cesionista, ya que el cifrado de delegación permite por definición al formador descifrar en representación del último. No obstante, el modelo de confianza puede ser diferente para su información privada. Un esquema de cifrado por delegación es conmutativo si el cesionista y el cesionario tienen que confiar entre sí con respeto a sus claves privadas; en caso contrario no es conmutativa. Un ejemplo conmutativo es que la clave de delegación se crea de tal manera que cada uno de los cesionista y cesionario pueden obtener la clave de descifrado del otro. Siempre que este sea el caso, el mecanismo de cifrado de delegación puede simplificarse por un protocolo de intercambio de claves que permite al cesionario usar la clave de descifrado del cesionista para descifrar los mensajes cifrados directamente.

Generalidad. En muchos casos, el cesionista quiere restringir el ámbito de delegación de derechos de descifrado. Frecuentemente las restricciones pretendidas incluyen que la clave de delegación sólo puede usarse por un cesionario designado, que la clave de delegación puede sólo aplicarse a un mensaje específico, o que la transformación de delegación sólo puede aplicarse por un facilitador específico. Por ejemplo, cuando se usa un esquema de cifrado de delegación en algunas aplicaciones como la custodia de claves, sería ideal que las claves de delegación fuesen independientes de los mensajes a los que se aplican. Pero para delegaciones ocasionales tal como especificar la herencia en el testamento de alguien de forma segura, puede ser muy deseable que una clave de delegación pueda estar restringida a una parte designada (por ejemplo, un nieto), aplicable a un mensaje específico (por ejemplo, alguna parte del testamento) y posiblemente usada en la transformación delegada por una parte en particular (un notario).

Degeneración. Cuando se usa en una situación extrema en la que el cesionista y el cesionario son la misma persona con una misma clave de delegación, el esquema de cifrado debe reducirse a un esquema de cifrado regular, sin introducir ninguna complicación (tal como claves de delegación no triviales, y los requisitos de un facilitador extra).

Como veremos posteriormente, los esquemas de Mambo y Okamoto son privados y no conmutativos. Las claves de delegación en sus esquemas pueden ser independientes del mensaje o dependientes pero no están restringidos a cesionarios designados. El esquema de Blaze y Strauss es justamente opuesto: es público pero conmutativo, y sus claves de delegación son independientes del mensaje pero asociados únicamente con cesionarios designados. En comparación, los esquemas de acuerdo con la invención mostrados en este documento son públicos y no conmutativos, y sus claves de delegación son dependientes del mensaje y restringidos a cesionarios designados.

Cifrado de delegación que usa el Sistema de Cifrado ElGamal

Como los esquemas de cifrado de delegación abordados más adelante en esta divulgación estarán todos basados sobre logaritmos discretos en grupos multiplicativos, se adopta por esto un escenario formal que es común en estas clases de esquemas de cifrado. La notación usada en este documento recuerda el esquema de cifrado de ElGamal. Los esquemas de cifrado basados sobres logaritmos discretos son particularmente ventajosos por sus ventajas técnicas sobre los esquemas tipo RSA y sus generalizaciones naturales a muchos grupos finitos tales como los grupos de curva elíptica sobre campos finitos.

Como se ha mostrado anteriormente, para cualquier número natural n, designemos $Z_n = \{0, 1, ..., n-1\}$ el conjunto números enteros de módulo n, y designemos $Z_n' = \{m \in Zn \mid gcd(m,n) = 1\}$ el grupo multiplicativo de Zn. Obsérvese que, cuando n es primo, $Z_n' = \{1, ..., n-1\}$. Para un módulo n y un número a que es primo relativo con n, designamos a^{-1} la multiplicación inversa de a en módulo n; esto es, a^{-1} es el elemento que satisface $aa^{-1} \equiv 1 \pmod n$.

Un elemento de Z_n' se dice que es de orden m si el número de sus potencias módulo n es m. Un generador g de Z_n', si existe, es un elemento de orden $|Z_n'|$ (el tamaño de Z_n'); en este caso, Z_n' es un grupo cíclico. Cuando n es un número primo, cada elemento de Z'_n excepto 1 es un generador de Z_n'.

Sea Z_n' un grupo cíclico con un generador g. El logaritmo discreto de un elemento x en base g, indicado como $\log_g x$, es el único entero a, $0 \leq a \leq n-1$, tal que $x = g^a \pmod n$. *El problema del logaritmo discreto* es que, dado un número primo p, un generador g de Z_p', y un elemento $x \in Z_p'$, encontrar el entero a, $0 \leq a \leq p-2$, tal que $g^a \equiv x \pmod p$.

Un problema muy estrechamente relacionado es el problema de Diffie-Hellman: dado un número primo p, un generador g de Z_p', y los elementos $g^a \pmod p$ y $g^b \pmod p$ encontrar $g^{ab} \pmod p$. El problema del logaritmo discreto es al menos tan difícil como el problema de Diffie-Hellman porque cualquier solución del problema anterior puede usarse para resolver el problema posterior.

El esquema de cifrado de ElGamal mostrado en la Figura 6 es una parte del basado en el logaritmo discreto, sistema de criptografía de clave pública propuesto por ElGamal para ambos cifrados y firma digital. Ver T. ElGamal, "A public key crytosystem and a signature scheme based on discrete logarithm", *IEEE Trans. on Information Theory*, Vol. 31, pag. 465 - 472 (1985).

Refiriéndonos de nuevo a la Figura 6 en detalle, el esquema de ElGamal se forma (etapa 610) por establecimiento de dos parámetros públicos p y g, donde p es un número primo (típicamente de 512 bits de longitud), tal que p-1 tiene un extenso factor primo de q (típicamente 160 bits) (por ejemplo, p = 2q +1) y g es un generador en Z_p. Una clave privativa para un usuario se fija (etapa 612) eligiendo uniformemente un número aleatorio $a \in Z_{p-1}$. Su clave pública relacionada se calcula (etapa 614) como $\alpha = g^a \pmod p$. El usuario publica α y guarda a en secreto.

Para cifrar un mensaje m a enviar a un usuario A con una clave pública α, se elige uniformemente un número aleatorio $k \in Zp-1$ (etapa 616), y un par de números (r,s), que representan juntos el mensaje cifrado a enviar a A, y se calculan (etapa 618) como sigue:

$$r = g^k \pmod p \qquad y \qquad s = m\alpha^k \pmod p.$$

Para descifrar el mensaje (r,s), el receptor A recupera el mensaje m (etapa 620) calculando

$$m = s(r^\alpha)^{-1} \pmod p$$

Obsérvese que la selección de parámetros públicos se intenta para establecer la ecuación $g^{p-1} \pmod p \equiv 1$ (Teorema pequeño de Fermat). Estos parámetros deben conocerse auténticamente por todos los usuarios. Deben elegirse, digamos, por alguna autoridad de confianza. También, el modo en el que se elige la clave privada asegura que la inversa a^{-1} de a en módulo p-1 existe y es única.

A diferencia del esquema de cifrado por clave pública RSA, el esquema de ElGamal es no-determinista, ya que el mensaje cifrado también depende del número aleatorio k. Ciertamente, es similar en naturaleza al protocolo de intercambio de claves de Diffie-Hellman; la clave establecida entre el emisor y el receptor para cifrar y descifrar el mensaje m es $g^{\alpha k}$ (mod p) a partir de $r = g^k$ (mod p) (parte del mensaje cifrado) y $\alpha = g^a$ (mod p) (la clave pública de A). A pesar de todo, la seguridad del esquema de cifrado de ElGamal se basa en la inflexibilidad del problema de logaritmo discreto y el problema de Diffie-Hellman. Hasta la fecha, la práctica en la búsqueda de algoritmos óptimos para el problema del logaritmo discreto no ha encontrado ninguna solución eficiente (compás polinómico). Es similar a la situación para el problema de factorización entera sobre el cual se basa la seguridad del esquema RSA. Además, se ha mostrado también que, para algunos números primos p, resolver el problema del logaritmo discreto es al menos tan difícil como resolver el problema de factorización de un mismo tamaño. Esto implica que para aquellos p, el esquema de ElGamal es al menos tan seguro como el esquema RSA.

Muy recientemente, se han propuesto varios esquemas de cifrado de delegación. Todos estos esquemas siguen el esquema de cifrado de delegación genérico en la delegación del derecho de descifrado: el cifrador envía un mensaje cifrado el cesionista A, quien a continuación delega el derecho de descifrado al cesionario por creación de la clave de delegación, y después de que se completa la transformación de delegación el cesionario finalmente descifra el mensaje. A continuación de presentan dos esquemas de cifrado de delegación representativos y conocidos: uno de Mambo y Okamoto y otro de Blaze y Strauss, ambos de los cuales son variaciones del esquema de ElGamal. Ya que tienen el mismo esquema de organización que el esquema de ElGamal, se omite la organización (ver etapas 610 - 614 de la Figura 6 anterior) de su presentación.

Mambo y Okamoto han propuesto tres esquemas de cifrado de delegación: dos están basados en el esquema de ElGamal y el otro está basado en el esquema RSA. El mostrado en la Figura 6 y descrito a continuación es el basado en ElGamal y comparte sus características básicas con los otros dos esquemas.

Refiriéndonos ahora a la Figura 7, dado un mensaje m que necesita enviarse al cesionista A con la clave pública α, el mensaje m se cifra eligiendo de forma uniforme un número aleatorio $k \in Zp\text{-}1$ (etapa 710) y calculando un par de números (r,s) que representan el mensaje cifrado (etapa 712) como sigue:

$$r = g^k(\text{mod } p) \qquad \text{y} \qquad s = m\alpha^k(\text{mod } p).$$

Para delegar el derecho de descifrado a un cesionario B, el cesionista crea un clave de delegación π eligiendo uniformemente un número aleatorio $a' \in Z_{p-1}$ (etapa 714) y calculando $\pi = aa'$ (mod(p-1)) (etapa 716). A continuación, A suministra la clave de delegación π a B (etapa 718) de una forma segura (por ejemplo, mediante su cifrado con la clave pública de B) y mantiene el valor de a' privado.

Para permitir a B descifrar el mensaje, A calcula $r' = r'^{a-1}$ (mod p) donde $a'\{\}^{-1}$ es el inverso multiplicativo de a' módulo p-1 (etapa 720). El par (r',s) es el mensaje cifrado transformado a enviar a B.

Una vez que se ha recibido el mensaje transformado (r',s) y la clave de delegación π, B descifra el mensaje m (etapa 722) calculando $m = s(r'^{\pi})^{-1}$ (mod p).

Este esquema de cifrado de delegación usa los componentes del cifrado y descifrado del esquema de ElGamal, excepto que la clave privada de B se reemplaza por la clave de delegación π. Esto es correcto porque, cuando se usa π para descifrar el mensaje transformado (r',s), se mantiene lo siguiente:

$$s((r')^{\pi})^{-1}(\text{mod } p) = s(r^{aa'a'-1})^{-1}(\text{mod } p) = mg^{ka}(g^{ka})^{-1}(\text{mod } p) = m$$

La seguridad de este esquema se evalúa en dos aspectos. La complejidad para cualquiera, incluyendo el cesionario B, para descubrir la clave a privada del cesionista A basada sobre toda la información disponible es la misma que la de resolver el problema del logaritmo discreto. La dificultad para cualquiera, incluso con la clave de delegación, de hacerse pasar por A para transformar el mensaje cifrado (es decir para generar (r',s)) es la misma que la de resolver el problema de Diffie-Hellman.

Este esquema tiene varias características muy atractivas. En primer lugar, su seguridad implica que es difícil para B recuperar la clave privada de A. En este sentido, no hay necesidad para A de confiar en B, y por tanto el esquema es no-conmutativo. En segundo lugar, la clave de delegación π generada es independiente del mensaje. B puede usarla para descifrar todos los mensajes transformados por A. En tercer lugar, este esquema satisface el requisito de transitividad. Una vez recibidos la clave de delegación y el mensaje transformado (r',s), el usuario delegado B puede delegar a su vez la delegación a otro usuario C, tratando π como la clave privada a y (r',s) como (r,s) y repitiendo la generación de delegación y el esquema de re-cifrado.

No obstante, la implementación del cifrado de delegación del modo de este esquema tiene varios defectos. En primer lugar, la clave de delegación no contiene información sobre el cesionario delegado B; se deriva solamente de la clave privada del cesionista A. Además, el descifrado del mensaje realizado por B tampoco necesita la clave privada de descifrado de B. Por consiguiente, el mensaje puede recuperarse por cualquiera que obtenga la clave de delegación

y el mensaje cifrado, no necesariamente B. De este modo, B puede pedir a cualquiera que descifre el mensaje pasando directamente la información de delegación. En muchos casos, esto no es deseable; A debe ser capaz de especificar el titular de clave que actúa en representación de A.

En segundo lugar, la clave de delegación π debe ser un secreto entre A y B y necesita transmitirse desde A hasta B de forma segura. Como resultado de que π no contiene información de B y (r',s) que posiblemente se comunican en público, revelar π es esencialmente igual que revelar el mensaje.

En tercer lugar, la información de delegación se ha dirigido por A. El valor de a' usado en la transformación es un secreto para A y es vital para prevenir que B conozca la clave de descifrado de A.

Abreviando, el esquema es no-conmutativo e independiente del mensaje, pero privado e incapaz de especificar el cesionario designado.

Blaze y Strauss han descrito otro esquema de cifrado de delegación de clave pública. Como puede verse en la Figura 8, el esquema es similar en estructura al cifrado de ElGamal, pero con los parámetros usados de diferente modo y la inversa del secreto usado para recuperar el mensaje.

Volviendo de nuevo a la Figura 8 con más detalle, dado un mensaje m que necesita enviarse al cesionista A con clave pública α, el mensaje m se cifra por elección uniforme de un número aleatorio $k \in Z_{p-1}$ (etapa 810) y calculando un par de números (r,s) que representan el mensaje cifrado (etapa 812) como sigue:

$$r = mg^k (\text{mod } p) \qquad \text{y} \qquad s = \alpha^k (\text{mod } p).$$

Para delegar el derecho de descifrado al cesionario B, el cesionista A crea un clave de delegación π por obtención de la clave b de descifrado privada de B (etapa 814) y computando $\pi = a^{-1}b(\text{mod}(p-1))$ (etapa 816), donde a^{-1} es la inversa de la clave privada a de A en módulo p-1. La clave de delegación π puede hacerse pública.

Para usar la clave de delegación π para convertir un mensaje (r,s) cifrado para A en un mensaje cifrado para B, el facilitador (no necesariamente A, ya que la clave de delegación π es pública) computa $s' = s^{\pi}$ (mod p) (etapa 818). El par (r,s') representa el mensaje cifrado transformado, que puede luego transmitirse a B.

Para descifrar el mensaje transformado, B computa $m = r(s'^{b-1})^{-1}$ (mod p) (etapa 820) donde b es la clave privada de B y b^{-1} es la inversa de b en módulo p -1.

El esquema es correcto, ya que en el descifrado del mensaje

$$s'^{b^{-1}} = g^k (\text{mod } p) \qquad \text{y} \qquad m = r(g^k)^{-1} (\text{mod } p)$$

El esquema es seguro en que el mensaje m y las claves secretas a y b no pueden recuperarse a partir de los mensaje cifrados y las claves públicas. Además, la publicación de la clave de delegación no compromete ni el mensaje m ni el secreto de las claves a y b. Más precisamente, el problema de recuperar m a partir de la información pública (α, β, r, s, π, s') es tan difícil como el problema de Diffie-Hellman.

En contraste con el esquema previo, la última característica de seguridad hace innecesario mantener la clave de delegación π en privado. De este modo, el cesionista A puede enviar públicamente π a quien sea (facilitador) para realizar la transformación delegada, o puede simplemente publicarla. Además el esquema no requiere ningún secreto desde A para realizar la transformación de delegación, y por consiguiente permite a cualquiera, de confianza o no realizar la transformación y por tanto elimina la necesidad de la presencia de A así como de B en la transformación.

También a diferencia del esquema previo, no hay diferencia para el usuario B entre descifrar un mensaje cifrado regular y descifrar en mensaje transformado delegado. Esta elegante característica permite al usuario B tratar todos los mensajes cifrados entrantes uniformemente. De hecho, es posible para un facilitador o servidor no de confianza realizar la transformación de delegación y luego enviar el mensaje al usuario B.

En contra de estas características deseables, este esquema es conmutativo; los titulare de claves A y B deben confiar mutuamente entre sí. B puede enterarse de la clave secreta a de A (multiplicando la clave de delegación por b^{-1}). Además, la clave de delegación es también independiente del mensaje, como lo era en el esquema previo, que delegaba a B el derecho de descifrar todos los mensajes cifrados por la clave privada a de A. Por consiguiente, este esquema es público e independiente del mensaje pero conmutativo.

Se han presentado en este documento dos esquemas de cifrado de delegación de acuerdo con la invención, y luego se han analizado en consideración a su seguridad, conmutatividad y funcionamiento. Como el esquema delegado privado, no son conmutativos, y al mismo tiempo, soportan claves de delegación públicas y transformaciones en la manera que lo hacen los esquemas conmutativos. No obstante, difieren de los esquemas privados y conmutativos en que son dependientes del mensaje. Además su funcionamiento global es mejor que el esquema de re-cifrado basado en ElGamal.

De nuevo, estos esquemas comparten el mismo esquema de organización que el esquema de ElGamal, y asumen que el cesionista A delega el derecho de descifrar al cesionario B.

Para entender como adaptar el esquema a un esquema de cifrado delegado, es útil examinar algunos detalles del esquema de ElGamal. Debe observarse que el componente r del mensaje cifrado m es independiente de la clave privada a y la clave pública α. Como $s = m\alpha^k \pmod p = mg^{ka} \pmod p$, α se usa sólo en el componente s, y a esta incorporada implícitamente en el exponente de s. De este modo, es suficiente para la transformación de delegación convertir el mensaje cifrado para A en el mensaje cifrado para B quitando la clave privada a de A a partir de s y reemplazarla por la clave privada b de B. Para impedir que B obtenga la clave privada a de A, la función que genera la clave delegada debe ser de algún modo "de sentido único". Ciertamente, esto puede conseguirse con ayuda del número aleatorio k como sigue:

$$\pi = g^{k(b-a)} \pmod p.$$

Por consiguiente, la transformación de delegación que completa la conversión del mensaje debe ser tal como la siguiente:

$$s' = s\pi \pmod p = mg^{ka}g^{k(b-a)} \pmod p = mg^{kb} \pmod p.$$

El argumento anterior conduce el esquema de la Figura 9. Este reúne que la clave de delegación y la transformación satisfacen los requisitos de seguridad y proporciona las características de ser público y no conmutativo.

Refiriéndonos de nuevo a la Figura 9, dado un mensaje m que necesita enviarse a un cesionista A con clave pública α, el mensaje m se cifra por elección uniforme de un número aleatorio $k \in Z'_{p-1}$ (etapa 910) y calculando un par de números (r,s) que representan el mensaje cifrado (etapa 912) como sique:

$$r = g^k \pmod p \qquad \text{y} \qquad s = m\alpha^k \pmod p.$$

Para delegar el derecho de descifrado a un cesionario B, el cesionista A crea un clave de delegación π por obtención de la clave de descifrado auténtica b de B (etapa 914) y calcula $\pi = r^{b-a} \pmod p$ (etapa 916).

El mensaje se transforma desde (r,s) a (r,s') por cálculo de $s' = s\pi \pmod p$ (etapa 918). El mensaje m se descifra entonces por B a partir de (r,s') computando $m = s'(r^b)^{-1} \pmod p$ (etapa 920).

Claramente, este esquema usa las etapas de cifrado y descifrado del mensaje del esquema de ElGamal. Es correcto ya que el mensaje m puede recuperarse desde

$$s'(r^b)^{-1} \pmod p = s\pi(r^b)^{-1} \pmod p = mg^{ak}g^{k(b-a)}(g^{kb})^{-1} \pmod p = m$$

Una bonita característica de este esquema es que, no sólo hace que los mensajes regulares y los mensajes delegados cifrados no parezcan diferentes para el cesionario B, sino también el esquema coincide con el esquema de ElGamal cuando A y B son el mismo usuario con la misma clave; en este caso, el valor de la clave de delegación π es igual a 1 y la transformación de delegación es la transformación identidad.

Es fácil de ver que el esquema es transitivo. Una vez recibido el mensaje transformado delegado, el cesionario B puede actuar como el cesionista A para delegar nuevamente el derecho de descifrado, supongamos, a otro cesionario C por repetición de la etapa de generación de delegación con las claves b y c en lugar de a y b.

También como el esquema conmutativo, la etapa de generación de delegación requiere ambas claves privadas de A y de B para generar la clave de delegación π. Como una alternativa, esta etapa puede realizarse por alguien que sea de la confianza de ambos A y B. Como se ha apuntado anteriormente, la clave privada de A se necesita definitivamente, porque de lo contrario alguien puede publicar la clave de delegación para recuperar el mensaje y el esquema de cifrado subyacente no es seguro. Para establecer y comunicar la clave privada b de B, pueden usarse muchos protocolos de intercambio de claves tal como el de Diffie-Hellman. Como se muestra con más detalle a continuación, en algunas aplicaciones prácticas el requisito de la clave b o no es un problema o puede relajarse.

Pero a diferencia de los esquemas privados y conmutativos, este esquema no hace fácil para el cesionario B descifrar mensajes cifrados para A salvo los que se intenta. Claramente, la clave de delegación π contiene un pieza de información que es específica del mensaje cifrado m, a saber, el número aleatorio k. En este sentido, el esquema de delegación es dependiente del mensaje. Además, el esquema es no-conmutativo en el sentido de que es difícil para B descubrir la clave privada a de A. Este hecho, junto con el funcionamiento del esquema se establecerá después de presentar el siguiente esquema.

Obsérvese que, en el esquema anterior, la transformación delegada sólo cambia el componente s del mensaje cifrado. Ya que s es la parte que actualmente lleva la información acerca del mensaje m, el esquema puede no ser eficiente cuando m es un mensaje muy largo. Por ejemplo, la clave delegada generada podría ser tan larga como el mensaje y el gasto de esfuerzo en la transformación de delegación sería lineal con respecto a la longitud del mensaje entero.

El esquema presentado en la Figura 10 pretende mejorar esta situación. Éste usa la etapa de cifrado del mensaje del esquema conmutativo en el cual el mensaje m se desplaza desde s hasta r. Su clave delegada y transformación ahora no tienen dependencia directa sobre el mensaje m.

Como se muestra en la Figura 10, dado un mensaje m que necesita enviarse a un cesionista A con la clave pública α, el mensaje m se cifra por elección uniforme de un número aleatorio $k \in Z'_{p-1}$ (etapa 1010) y calculando un par de números (r,s) que representan el mensaje cifrado (etapa 1012) como sigue:

$$r = mg^k \pmod p \qquad \text{y} \qquad s = \alpha^k \pmod p.$$

Para delegar el derecho de descifrado a un cesionario B, el cesionista A crea un clave de delegación π por obtención de la clave de descifrado auténtica b de B (etapa 1014) y calculando $\pi = (s^{a^{-1}})^{b-a} \pmod p$ (etapa 1016), donde a^{-1} es la inversa de a en módulo p-1.

El mensaje se transforma desde (r,s) a (r,s') por cálculo de $s' = s\pi \pmod p$ (etapa 1018). El mensaje m se descifra a continuación por B desde (r,s') por computación de $m = r(s'^{b^{-1}})^{-1} \pmod p$ (etapa 1020), donde b^{-1} es la inversa de b en módulo p-1.

Este esquema es correcto ya que

$$\begin{aligned} r(s'^{b^{-1}})^{-1} \pmod p &= r((s\pi)^{b^{-1}})^{-1} \pmod p \\ &= r((s(s^{a^{-1}})^{b-a})^{b^{-1}})^{-1} \pmod p \\ &= r((g^{ka}g^{k(b-a)})^{b^{-1}})^{-1} \pmod p \\ &= r((g^{kb})^{b^{-1}})^{-1} \pmod p \\ &= mg^k(g^k)^{-1} \pmod p \\ &= m \end{aligned}$$

Otras propiedades de este esquema se pueden verificar de la misma manera que en el esquema previo.

Debido a su naturaleza similar, sólo se analiza el primero de los dos nuevos esquemas en esta sección respecto a su seguridad y no-conmutatividad. Puede realizarse casi la misma discusión para el segundo esquema. Además, aunque el primer esquema (así como también el segundo) es transitivo y su seguridad puede involucrar más de dos titulares de claves, el análisis que se da considera el caso de dos titulares de clave; el caso general es también similar. Para claridad en la presentación, la frase "(mod p)" se omitirá en esta sub-sección; esta ocurrencia debe estar clara por el contexto.

Recordemos que, además de los parámetros del esquema (p,g), la información disponible desde el esquema incluye

$$\alpha = g^a, \quad \beta = g^b, \quad r = g^k, \quad s = mg^{ak}, \quad \pi = g^{k(b-a)}, \quad s' = mg^{bk}.$$

Por las razones mostradas a continuación, el esquema es seguro desde el punto de vista computacional. Es difícil recuperar el mensaje m y las claves secretas a y b a partir de la información pública, supuesto que los problemas de Diffie-Hellman y el logaritmo discreto son difíciles de resolver. Ya que la clave de delegación es parte de la información pública, esto implica que su publicación no compromete el mensaje ni las claves secretas. Una consecuencia de esto es que es también difícil para cualquiera falsificar un clave de delegación válida de manera sistemática. Además de eso, el esquema se muestra no ser conmutativo en el sentido de que incluso con la clave privada de B, es aún difícil recuperar la clave privada de A. Si la clave de delegación se genera ciertamente por una tercera parte de confianza para ambos A y B, este hecho implica que no es necesario tampoco para B confiar en A. Esto es una mejora significativa sobre el esquema conmutativo.

Además, como se ha establecido anteriormente, los esquemas de cifrado de delegación de la invención son más eficientes que re-cifrar un mensaje. A continuación, en la Tabla 2, está el funcionamiento de los dos esquemas de cifrado de delegación usando el algoritmo de ElGamal, en términos de la cantidad de computación que requieren. En la Tabla 2, se listan el número de operaciones de multiplicación, operaciones exponenciales, e inversiones, todas ellas realizadas en módulo p, para estos esquemas.

TABLA 2

Operaciones	Re-Cifrado			1er Esquema (Fig. 9)			2º Esq. (Fig. 10)		
	mult.	exp.	inv.	mult.	exp.	inv.	mult.	exp.	inv.
Cifrado	1(x2)	2(x2)	0(x2)	1	2	0	1	2	0
Gen. Clave Deleg.				0	1	0	0/1	2/1	1/0
Transformación				1	0	0	1	0	0
Descifrado	1(x2)	1(x2)	1(x2)	1	1	1	1	1	2/1
Total	4	6	2	3	4	1	3/4	5/4	3/1

Obsérvese que el número total de operaciones de re-cifrado usando el esquema de ElGamal es el doble que el número de operaciones para un cifrado y descifrado simple de ElGamal, ya que el mensaje debe primero cifrarse, luego descifrarse, luego re-cifrarse, luego re-descifrarse. Además, la computación en el segundo esquema puede optimizarse por (i) pre-computando las inversas a^{-1} y b^{-1} en la etapa de organización del esquema y (ii) multiplicando las dos componentes exponenciales (modulo (p-1)) en la etapa de generación delegada en lugar de usar dos operaciones exponenciales. El segundo conjunto de números en el segundo esquema resultan de la optimización. Sobre todo, los esquemas de cifrado de delegación inventados presentados en este documento tienen mejor funcionamiento que el simple, esquema de re-cifrado basado en ElGamal.

Aplicaciones

Los esquemas de cifrado de delegación públicos y no conmutativos proporcionan un mecanismo de clave para la implementación de un amplio rango de aplicaciones. La distribución masiva de documentos y protección de ficheros son dos motivos clave para este descubrimiento. Estas aplicaciones corresponden a dos situaciones típicas para el cifrado de delegación. La primera se refiere al caso en el que el cesionista es quien cifra el mensaje en primer lugar, mientras que la siguiente es la auto-delegación en la cual el cesionista y el cesionario son el mismo titular pero con diferentes claves.

De nuevo, obsérvese que un documento se refiere a cualquier fichero digital cuyo contenido podría ser texto, gráficos, audio, vídeo, ejecutable o incluso multi-media. Usualmente, un documento es demasiado grande en tamaño, incluso después de una compresión. Ya que los algoritmos de clave pública tienden a ser muy lentos cuando se comparan con los algoritmos convencionales de clave privada tales como DES, IDEA y RC4, y los algoritmos de clave privada requieren el establecimiento de claves secretas con las que empezar, la aproximación más práctica a la distribución masiva y segura de documentos sobre redes es combinar los mecanismos de clave pública y de clave privada. Típicamente, se usa un algoritmo de clave privada eficiente para cifrar el documento por uso de una clave generada aleatoriamente, llamada la clave de sesión, y se usa la clave pública para cada receptor de documento para cifrar esta clave de sesión. Los receptores usan sus claves privadas para recuperar la clave de la sesión secreta y a continuación la usan para descifrar el documento.

Ciertamente, el método de distribución de documentos anterior tiene el sabor del cifrado de delegación; el propietario cifra el documento primero usando un esquema de clave privada y luego concede el derecho de descifrado, bajo petición, a sus receptores vía un esquema de clave pública. Esto reúne que, puede usarse uno cualquiera de los dos sistemas de cifrado de delegación para combinar las mejores características del método dentro de un simple, esquema de cifrado normal.

Tomamos, por ejemplo, el segundo esquema mostrado anteriormente (Figura 10). Dos observaciones están en orden. Primero, el componente r del mensaje cifrado puede generarse usando cualquier esquema de cifrado de clave privada con $K = g^k \pmod p$ como la clave de sesión secreta. En consecuencia, el mensaje m se puede recuperar en la etapa de descifrado del mensaje por su descifrado de clave privada correspondiente usando la clave de sesión secreta $K' = s'^{\,b^{-1}} \pmod p = K$. En efecto, el esquema de cifrado de clave secreta usado en el esquema es $r = E_k(m) = mK \pmod p$ para cifrado y $m = D_{k'}(r) = rK'^{-1} \pmod p$ para descifrado. Otro ejemplo simple es el esquema de cifrado basado en la XOR orientada a bit ($\oplus$). En este caso, la computación de r y m puede reemplazarse por

$$r = E_k(m) = m \oplus K \qquad y \qquad m = Dk(r) = r \oplus K.$$

Ciertamente, los esquemas de cifrado de clave privada más sofisticados tal como DES y triple-DES pueden emplearse si es necesaria una seguridad más fuerte.

La segunda observación es que, si el cesionista A es el que cifra el mensaje m, entonces A puede mantener el número aleatorio k privado y usar la clave pública de B $\beta = g^h \pmod p$, en lugar de la clave privada b de B, para generar la clave delegada:

$$\pi = (\beta\alpha^{-1})^k \pmod p,$$

donde α es la clave pública de A. Esto elimina el requisito para la clave privada b de B (o clave de intercambio entre A y B), e implica que B tampoco necesita confiar en A.

Estas dos observaciones conducen al esquema de distribución de documentos mostrado en la Figura 11, que está basado en el segundo esquema de cifrado de delegación de acuerdo con la invención, mostrado anteriormente (y en conexión con la Figura 10). En este esquema, se usa un esquema de cifrado de clave privada para cifrar el mensaje sólo una vez para todos los receptores, mientras que se usa una porción de clave delegada de menor expediente para cifrar una pequeña cantidad de información - la clave de sesión - particularizada una vez por cada receptor. Una característica beneficiosa de este esquema es que el documento cifrado puede almacenarse en un depósito accesible públicamente, y la transformación de delegación puede realizarse por el propietario del documento A, el receptor B, o el almacén donde está físicamente almacenado el documento, dependiendo de las necesidades de los sistemas de distribución y gestión de documentos reales.

Refiriéndonos de nuevo a la Figura 11, el esquema se organiza de la misma forma que un esquema de ElGamal estándar (ver Figura 6, descrita anteriormente). Además, se selecciona un esquema de cifrado de clave privada, simétrico, (etapa 1110). Su función de cifrado es $m \rightarrow E_k(m)$ y la función de descifrado es $r \rightarrow D_k(r)$, donde K es alguna clave privada.

Para cifrar un documento m, el propietario A en primer lugar elije un número aleatorio uniformemente $k \in Z_{p-1}$ (etapa 1112) y calcula una clave de sesión $K = g^k \pmod p$ (etapa 1114). El documento cifrado (r,s) se calcula a continuación como sigue:

$$r = E_k(m) \qquad \text{y} \qquad s = K^a \pmod p.$$

(etapa 1116), donde a es la clave privada de A. A mantiene privados el par (s,k).

Bajo la petición desde un receptor B del documento cifrado (r,s), A en primer lugar obtiene la auténtica clave pública β de B (etapa 1118) y recupera k a partir del par (s,k) (etapa 1120). A continuación A computa $\pi_B = \beta^k \, s^{-1} \pmod p$ (etapa 1122), donde s^{-1} es la inversa de s en módulo p, como la clave de delegación para B.

El documento se transforma a continuación por computación $s' = s\pi_B \pmod p$ (etapa 1124); el par (r,s') representa el documento transformado particularizado para B.

Para descifrar el documento particularizado (r,s') y recuperar el documento original m, B primero recupera la clave de la sesión por cálculo $K = s'^{b^{-1}} \pmod p$ (etapa 1126), donde b^{-1} es la inversa de b en módulo p-1. A continuación se descifra el documento calculando $m = D_k(r)$ (etapa 1128).

Como se ha descrito anteriormente, una adaptación de la presente invención es también aplicable a la aplicación de protección de ficheros. Usualmente, la protección de ficheros en sistemas inseguros tales como ordenadores portátiles y hardware conectados en red involucra cifrados de larga duración de ficheros. De este modo, las claves de cifrado usadas para cifrar ficheros tienen una tiempo de vida mucho mayor que sus equivalentes de comunicación. Mientras que la clave secreta primaria de usuario, de larga duración, puede ser la representación fundamental de la identidad de red del usuario, hay un peligro de que pueda terminar comprometida si se usa para muchos ficheros sobre un largo periodo de tiempo. Si la clave primaria se pierde o se roba, no sólo se pierden los contenidos de los ficheros cifrados con su revelación, sino también información personal de usuario basada en la clave tal como cuentas de tarjetas de crédito, números de la seguridad social, y así sucesivamente. Por consiguiente, es preferible a menudo usar un método en-linea en el cual se deduce una nueva clave de descifrado a partir de la clave primaria cada vez que se necesita cifrar un fichero y tenerlo actualizado sobre una base de regularidad.

Con los esquemas de cifrado de delegación mostrados en este documento, se pueden generar nuevas claves de descifrado y actualizar constantemente mediante auto-delegaciones para mantenerlas frescas. Una vez que se crea una nueva clave y se genera la clave de delegación correspondiente, se puede destruir la clave secreta antigua, manteniendo la capacidad de descifrar el fichero con la nueva clave y la clave de delegación.

La figura 12 muestra un esquema de protección de ficheros que usa un tarjeta inteligente para almacenar y actualizar las claves de descifrado. Está basada de nuevo en el segundo esquema de cifrado de delegación presentado en este documento, como se ilustra en la Figura 10.

Como se muestra en la Figura 12, par cifrar un fichero m, un procesador incorporado en la tarjeta inteligente elige un número aleatorio $k \in Z_{p-1}$ (etapa 1210) y computa

$$r = mg^k (\text{mod } p) \qquad y \qquad s = (g^k)^a (\text{mod } p)$$

(etapa 1212), donde a es la clave privada de la tarjeta inteligente. El par (r,s) representa el fichero m en la forma cifrada.

Cuantas veces sea necesario o se desee, por ejemplo cada pocas semanas o después de un número determinado de accesos, la tarjeta inteligente genera uniformemente otro número aleatorio a' $\in Z_{p-1}$ (etapa 1214) y computa s' = $(s^{a^{-1}})^{a'}$ (mod p) (etapa 1216), donde a^{-1} es la inversa multiplicativa de a en módulo p-1. El fichero cifrado (r,s) se reemplaza a continuación con (r,s') (etapa 1218), y la clave de descifrado a se reemplaza con una nueva clave de descifrado a' (etapa 1220). Estos pasos 1214 - 1220 pueden repetirse tantas veces como se desee.

Para recuperar el fichero original m a partir de su versión cifrada (r,s), el procesador de la tarjeta inteligente usa la última clave de descifrado a para computar $m = rs^{a^{-1}}$ (mod p) (etapa 1222).

Obsérvese que la etapa de cifrar el fichero puede comenzar con cualquier clave secreta que el genere, no necesariamente la clave privada de la tarjeta inteligente.

Para guardar los ficheros cifrados frescos por actualización de los datos de cifrado con una pieza de información generada por la tarjeta inteligente ayuda a mantener copias útiles simples de los ficheros protegidos. Esto, en algún sentido, proporciona también protección de copia. Además, la no-conmutatividad de los esquemas hace inútiles las copias previas de los ficheros, y la correspondiente información secreta almacenada se ha cambiado (y preferiblemente destruido).

Cifrado de delegación que Usa el Sistema de Criptografía de Cramer-Soup

Aunque los ejemplos anteriores y todos los algoritmos emplean varias adaptaciones del sistema de Criptografía de ElGamal, debe observarse que pueden adaptarse también otros sistemas de criptografía adaptados por un esquema de acuerdo con la invención.

Por ejemplo, el sistema de criptografía de clave pública de Cramer-Soup es un sistema de criptografía propuesto recientemente que es el primer sistema práctico de clave pública para ser probablemente inmune al ataque de texto cifrado elegido adaptativo. Ver R. Cramer y Shoup, "A practical Pubic Key Cryptosystem Provably Secure against Adaptative Chosen Ciphertext Attack", *Proceedings of CRYPTO 98*, Springer Verlang LNCS, vol. 1462, pag. 13-25 (1998). El ataque de texto cifrado elegido adaptativo asume que atacante puede obtener el descifrado de cualquier texto cifrado elegido distinto del texto cifrado objetivo. Por ejemplo, si el texto cifrado objetivo para el cual se busca el texto inicial es c, entonces el atacador asume tener acceso a un "oráculo de descifrado" que descifrará cualquier texto cifrado excepto c, incluyendo por ejemplo c+1, 4c etc. RSA Y ElGamal caen fácilmente a esta clase de ataque. Una noción diferente, pero no equivalente de ataques activos contra la seguridad se llama no-maleabilidad; no obstante, los sistemas no-maleables conocidos no son prácticos.

En la Figura 13 se muestra más adelante una descripción de una versión libre de información inservible del sistema de criptografía de Cramer-Shoup, la seguridad del cual se basa estrictamente en el problema de decisión de Diffie-Hellman para un grupo arbitrario. Más adelante, se ilustrara cómo delegar el derecho a descifrar es un esquema de Cramer-Shoup en dos situaciones diferentes.

Refiriéndonos inicialmente a la Figura 13, el sistema se organiza eligiendo un grupo G como grupo de primer orden q, donde q es grande (etapa 1310). El sistema asume que los mensajes de texto en claro son (o pueden ser codificados como) elementos del grupo G, y los mensajes de texto cifrados son elementos de G4 = G x G x G x G; esto es, un mensaje de texto cifrado es cuatro veces más largo que su correspondiente mensaje de texto original.

Un buen ejemplo del grupo G es el subgrupo de orden q en el conjunto multiplicativo Z_p para algún número primo grande p = 2q + 1. En este caso, un mensaje m del conjunto {1,. . .,q} puede "codificarse" elevándolo al cuadrado en módulo p, resultando en un elemento de G, y el mensaje m puede recuperarse a partir de su codificado por computación de la raíz cuadrada única de su codificado en módulo p, en el conjunto {1,...,q}.

Se genera una clave como sigue. En primer lugar se eligen elementos aleatorios $g_1, g_2 \in G$, y se eligen elementos aleatorios $x_1, x_2, y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}, z \in Z_q$. (etapa 1314). A continuación, se computan el grupo de elementos $c = g_1^{x1} g_2^{x2}$, $d_1 = g_1^{y11} g_2^{y12}$, $d_2 = g_1^{y21} g_2^{y22}$, $d_3 = g_1^{y31} g_2^{y32}$, y $h = g^z$ (etapa 1316). Se calcula entonces la clave pública para que sea (g_1, g_2, c, d_1, d_2, d_3, h) (etapa 1318) y la clave privada se calcula para que sea ($x_1, x_2, y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}$, z) (etapa 1320).

Dado un mensaje m ∈ G, el método de cifrado comienza eligiendo $r \in Z_q$ aleatoriamente (etapa 1322). A continuación el texto cifrado (u_1, u_2, e, v) se calcula como sigue (etapa 1324):

$$u_1 = g_1^r, \quad u_2 = g_2^r, \quad e = h^r m, \qquad y \qquad v = c^r d_1^{u1r} d_2^{u2r} d_3^{er}.$$

Dado el texto cifrado (u_1, u_2, e, v), en primer lugar el correspondiente algoritmo comprueba si $v = u_1^{x1+u1y11+u2y21+ey11} u_2^{x2+u1y12+u2y22+ey12}$ (etapa 1326). Si no, el esfuerzo de descifrado se rechaza (etapa 1328). De lo contrario, el mensaje m se calcula como $m = e / u_1^{z}$ (etapa 1330).

La corrección de un sistema de criptografía puede verificarse por comprobación de que el descifrado y cifrado de un mensaje produce el mensaje. En este caso, ya que $u_1 = g_1^{r}$ y $u_2 = g_2^{r}$ tenemos $u_1^{x1} u_2^{x2} = g_1^{rx1} g_2^{rx2} = c^r$. Asimismo,

$$U_1^{x1+u1y11+u2y21+ey11} u_2^{x2+u1y12+u2y22+ey12} = c^r d_1^{u1r} d_2^{u2r} d_3^{er} \quad \text{y} \quad u_1^{z} = h^r.$$

Entonces, para el texto cifrado válido, la comprobación realizada en el algoritmo de descifrado pasará.

La seguridad de este sistema de criptografía descansa bajo la dificultad en resolver el problema de decisión de Diffie-Hellman. Un algoritmo que resuelve el problema de decisión de Diffie-Hellman es un test estadístico que puede distinguir de modo efectivo las dos siguientes distribuciones: (a) cuádruplas aleatorias (g_1, g_2, u_1, u_2) $\in G^4$, y (b) cuádruplas aleatorias (g_1, g_2, u_1, u_2) $\in G^4$, donde g_1, g_2 son aleatorios y $u_1 = g_1^{r}$ y $u_2 = g_2^{r}$ para algún número aleatorio $r \in Z_q$.

Relacionado con el problema de decisión de Diffie-Hellman (dados g, g^x, y g^y computar g^{xy}), y el problema del logaritmo discreto (dado g y g^x, computar x). Dentro del tiempo polinómico, el problema de decisión de Diffie-Hellman puede reducirse al problema de Diffie-Hellman que a su vez puede reducirse al problema del logaritmo discreto. Es la relación entre los tres problemas lo que lleva a la posibilidad de delegar el derecho a descifrar para el sistema de Cramer-Shoup.

Asumamos que alguien quiere delegar el derecho a descifrar desde un delegante (Alice, A) a un delegado (Bob, B). Supongamos que Alice tiene la clave pública (g_1, g_2, c, d_1, d_2, d_3, h) y la clave privada (x_1, x_2, y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}, z), y que Bob tiene la clave pública (g_1', g_2', c', d_1', d_2', d_3', h') y la clave privada (x_1', x_2', y_{11}' y_{12}', y_{21}', $y_{22:}$', y_{31}', y_{32}', z').

Recordemos, que para un mensaje de texto original $m \in G$, el mensaje de texto cifrado para el delegante A es $M = (u_1, u_2, e, v)$, donde $u_1 = g_1^{r}$, $u_2 = g_2^{r}$, $e = h^r m$, y $v = c^r d_1^{u1r} d_2^{u2r} d_3^{er}$. De modo similar, si el mensaje m está cifrado directamente por el delegado B, el mensaje de texto cifrado es M' = (u_1', u_2', e', v'), donde $u_1' = g'^{r'}_1$, $u'_2 = g'^{r'}_2$, $e' = h'^{\,r'} m$, y $v' = c'^{\,r'} d'^{u1r'}_1 d'^{u2r'}_2 d'^{er'}_3$, donde r' es también un número aleatorio de Z_q. Obsérvese además que $v = (c d_1^{u1}, d_2^{u2} d_3^{e})^r$ y $v' = (c' d_1'^{u1} d_2'^{u2} d_3'^{e'})^{r'}$.

En base a las ideas que se han mostrado anteriormente, para delegar el derecho de descifrar desde A hasta B involucra una clave de transferencia π, usando esa clave de transferencia para transformar M en M'. En lo siguiente, se asume que las componentes g_1', g_2' de la clave pública de B son idénticas a las componentes g_1, g_2 de la clave pública de A (análogamente que los parámetros de sistema ElGamal descritos anteriormente). También, se asume que el número aleatorio r' es el mismo que r. Bajo estas dos suposiciones, los elementos u_1',u_2' del mensaje de texto cifrado de B son los mismos que los elementos u_1, u_2 del mensaje de texto cifrado de A.

Refiriéndonos de nuevo a la Figura 14, el sistema se organiza eligiendo G como un grupo de primer orden q, donde q es grande (etapa 1410). Luego, como anteriormente, la clave se genera como sigue. En primer lugar, se eligen los elementos aleatorios g_1, $g_2 \in G$ (etapa 1412), y los elementos x_1, x_2, y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}, $z \in Z_q$ (etapa 1414). A continuación, se computan los elementos del grupo $c = g_1^{x1} g_2^{x2}$, $d_1 = g_1^{y11} g_2^{y12}$, $d_2 = g_1^{y21} g_2^{y22}$, $d_3 = g_1^{y31} g_2^{y32}$, y $h = g_1^{z}$ (etapa 1416). La clave pública se calcula a continuación para que sea (g_1, g_2, c, d_1, d_2, d_3, h) (etapa 1418) y la clave privada se calcula para que sea (x_1, x_2, y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}, z) (etapa 1420).

Dado un mensaje $m \in G$, el método de cifrado comienza eligiendo el número aleatorio $r \in Zq$ (etapa 1422). A continuación se calcula el texto cifrado (u_1, u_2, e, v) como sigue (etapa 1424): $u_1 = g_1^{r}$, $u_2 = g_2^{r}$, $e = h^r m$, y $v = c^r d_1^{u1r} d_2^{u2r} d_3^{er}$.

Si está disponible la clave privada de B para generar la clave de transferencia π, tal clave se obtiene (etapa 1426) y a continuación puede calcularse π (etapa 1428) como sigue:

$$\pi = (\varepsilon, \theta, \delta_1, \delta_2, \delta_3)$$

donde

$\varepsilon = e'/e = g_1^{(z'-z)}$

$\theta = c'^{r}/c^r = g_1^{(x'1-x1)r} g_2^{(x'2-x2)r} = u_1^{x'1-x1} u_2^{x'2-x2}$

$\delta_1 = d'^{\,r}_1/d_1^{\,r} = u_1^{y'11-y11} u_2^{y'12-y12}$

$\delta_2 = d'^{\,r}_2/d_2^{\,r} = u_1^{y'21-y21} u_2^{y'22-y22}$

$\delta_3 = d'^{\,rr}_3 d_3^{\,r} = u_1^{y'31r-y31} u_2^{y'32r-y32}$

La transformación de texto cifrado es entonces

$$u'_1 = u_1, \qquad u'_2 = u_2, \qquad e' = e\varepsilon, \qquad \text{y} \qquad v' = v\theta\delta_1^{u1}\delta_2^{u2}\delta_3^{e}$$

Esto transforma el texto cifrado (u_1, u_2, e, v) en (u_1, u_2, e', v') (etapa 1430)

El receptor/delegado es también capaz de descifrar el texto cifrado transformado (u1, u2, e', v'). Como anteriormente, el algoritmo de descifrado en primer lugar comprueba si $v' = u'_1{}^{x'1u'1y'21+u'2y'21+e'y'31}u'_2{}^{x'2+u'1y'12+u'2y'22+e'y'32}$ (etapa 1432). Si no, el esfuerzo de descifrado se rechaza (etapa 1434). De lo contrario, el mensaje m se calcula como $m = e'/u'_1{}^{z'}$ (etapa1434).

En el caso de que sólo pueda usarse la clave pública del delegado B por delegación del derecho de descifrar el mensaje de delegante A a B, se necesita almacenar y usar el número aleatorio r usado inicialmente en cifrar el mensaje para A. Esto puede ser un problema cuando la parte que genera la clave de transferencia no es A, y puede no ser un problema si la parte es, en efecto A. En cualquier caso, si está disponible, se puede generar la clave de transferencia π, usando la clave pública de B como sigue:

$$\pi = (\varepsilon, \theta, \delta_1, \delta_2, \delta_3)$$

donde

$\varepsilon = e'/e = (g_1{}^{z'}/g_1{}^{z})^r = (h'/h)^r$

$\theta = c'^r/c^r = (c'/c)^r$

$\delta_1 = d'_1{}^r/d1^r = (d_1'/d_1)^r$

$\delta_2 = d'_2{}^r/d_2{}^r = (d_2{}^r/d_2)^r$

$\delta_3 = d'_3{}^{rr}/d_3{}^r = (d'_3{}^r/d_3)^r$

La transformación delegada es entonces

$$u'_1 = u_1, \qquad u'_2 = u_2, \qquad e' = e\varepsilon, \qquad \text{y} \qquad v' = v\theta\delta_1^{u1}\delta_2^{u2}\delta_3^{e}.$$

Es directo verificar, en cualquier caso, que el delegado B puede usar su propia clave privada para descifrar el texto cifrado (u'_1, u'_2, e', v') transformado por los métodos mostrados anteriormente. Ya que los mecanismos usados en este documento sobre los sistemas de criptografía de Cramer-Shoup son los mismos que los usados anteriormente sobre los sistemas de criptografía de ElGamal, son públicos y no conmutativos, asumiendo el problema de Diffie-Hellman y el problema del logaritmo discreto son difíciles de resolver.

Como se ha descrito anteriormente, mediante la mejora de esquemas de cifrado de clave pública común con la capacidad de cifrado de delegación, se hace posible soportar el descifrado de delegación designado. Esta divulgación ha presentado dos esquemas de cifrado de delegación, públicos y no-conmutativos, que han heredado los méritos de los esquemas existentes y descartado sus inconvenientes. Los nuevos esquemas se han mostrado para que tengan aplicaciones directas en la distribución masiva de elementos y la protección de ficheros. La idea básica de estos nuevos esquemas también se ha aplicado a sistemas de criptografía de otros tipos tales como el sistema de criptografía de Cramer-Shoup, mejorándolos dentro de los esquemas de cifrado de delegación.

Mientras que se han descrito los diversos aspectos de la presente invención con referencia a varios aspectos y sus realizaciones, se ofrecen esas realizaciones a modo de ejemplo, no como limitación. La descripción detallada anterior de la invención se ha presentado a propósito de ilustración y descripción. No se intenta ser exhaustivo o limitar la invención a la forma precisa revelada, y obviamente son posibles muchas modificaciones y variaciones a la luz de lo enseñado anteriormente. Las realizaciones descritas se eligieron para la mejor explicación de los principios de la invención y sus aplicaciones prácticas para permitir con ello a otros expertos en la técnica la mejor utilización de la invención en diversas realizaciones y con diversas modificaciones como adaptaciones al uso particular contemplado.

REIVINDICACIONES

1. Un método para cifrar un documento original para distribución a un receptor selecionado de una pluralidad de posibles receptores, que comprende las etapas de:

cifrar (1116; 1424) el documento original con una clave de sesión para crear un documento cifrado;

elegir (1112; 1422), por el cesionista, un elemento aleatorio, dicho elemento aleatorio es privado para dicho cesionista;

generar (1122; 1428), por dicho cesionista, una clave de delegación basada en una clave pública correspondiente al receptor seleccionado y dicho elemento aleatorio; y

transformar (1124; 1430), por dicho cesionista, al menos un componente del documento cifrado que comprende la información cifrada relativa a dicho documento original con la clave de delegación para crear un documento transformado permitiendo por ello que sólo dicho receptor seleccionado recupere el documento original a partir del documento cifrado por uso de la clave privada correspondiente a dicha clave pública.

2. El método de la reivindicación 1, en el cual dicho paso de cifrado (1116; 1424) se realiza por dicho cesionista.

3. El método de la reivindicación 1, en el cual dicho paso de cifrado (1116; 1424) se realiza por un cifrador, y dicho documento cifrado se proporciona a dicho cesionista antes de dicha etapa de transformación (1124; 1430).

4. El método de cualquiera de las reivindicaciones 1 a 3, que comprende además la etapa de transmitir el documento transformado al receptor seleccionado.

5. El método de cualquiera de las reivindicaciones 1 a 4, que comprende además las etapas de:

recuperar la clave de sesión a partir del documento transformado; y

descifrar el documento transformado con la clave de sesión para recuperar el documento original.

6. El método de la reivindicación 5, en el cual la etapa de recuperación se realiza por aplicación de una clave privada correspondiente al receptor seleccionado.

7. El método de cualquiera de las reivindicaciones 1 a 6, en el cual la etapa de cifrado se realiza con un esquema de cifrado simétrico de clave secreta.

8. El método de cualquiera de las reivindicaciones 1 a 6, en el cual la etapa de cifrado se realiza con un esquema de cifrado que se basa en el sistema de criptografía de ElGamal.

9. El método de la reivindicación 7, en el cual el documento cifrado comprende una primera porción representativa del documento original cifrado vía el esquema de cifrado simétrico de clave secreta usando la clave de sesión, y una segunda porción representativa de la clave de sesión cifrada usando una clave privada del propietario.

10. El método de cualquiera de las reivindicaciones 1 a 9, en el cual el documento original se distribuye al receptor seleccionado mediante al menos un cesionista intermedio adicional por repetición de las etapas de generación y de transformación para cada cesionista intermedio adicional.

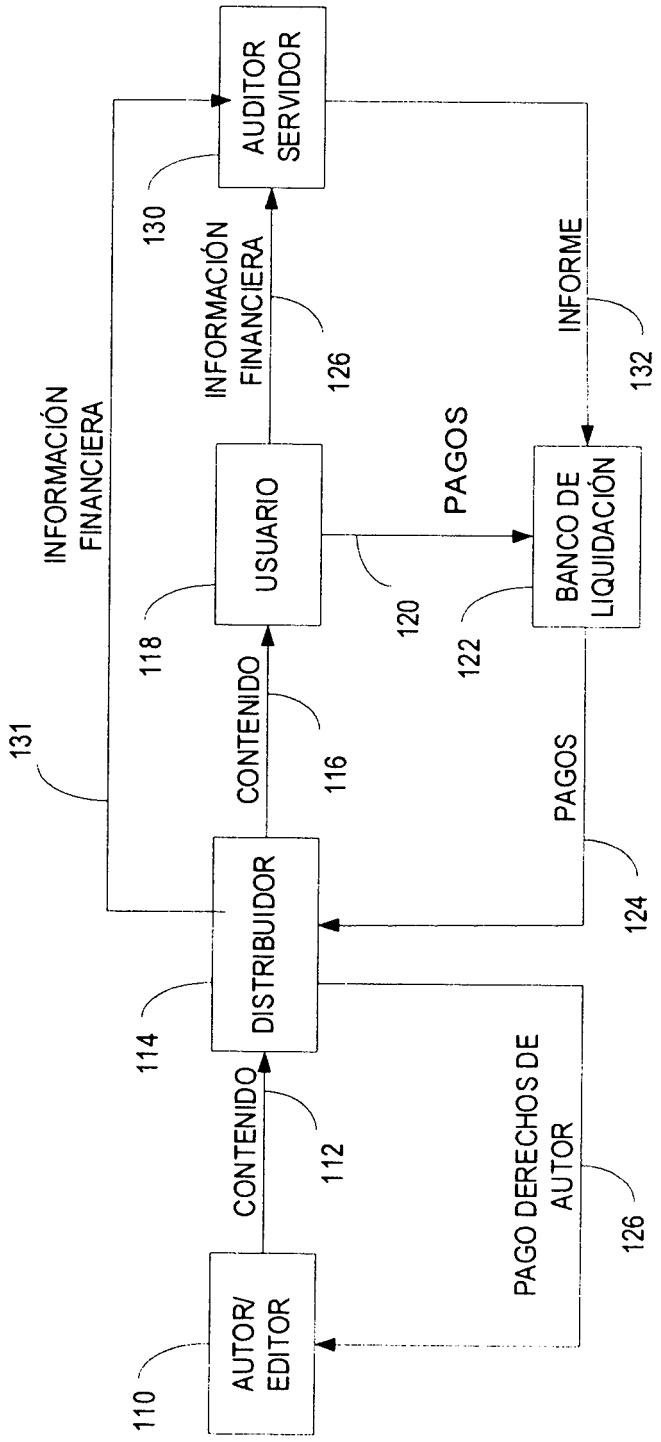

AUDITOR SERVIDOR
130
INFORMACIÓN FINANCIERA
126
INFORME
132
INFORMACIÓN FINANCIERA
131
USUARIO
118
PAGOS
120
BANCO DE LIQUIDACIÓN
122
CONTENIDO
116
PAGOS
124
DISTRIBUIDOR
114
CONTENIDO
112
PAGO DERECHOS DE AUTOR
126
AUTOR/ EDITOR
110

FIG. 1

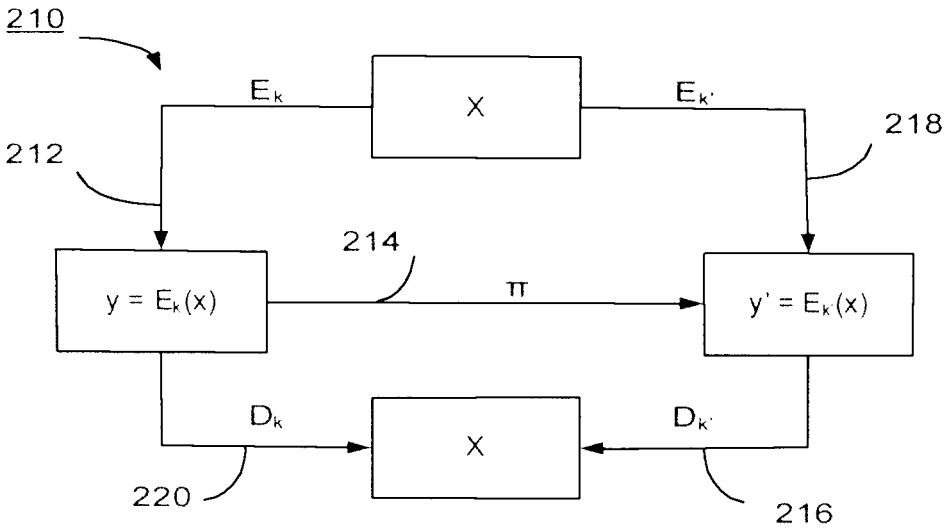

210
E_k
X
$E_{k'}$
218
212
214
$y = E_k(x)$
π
$y' = E_{k'}(x)$
D_k
X
$D_{k'}$
220
216

FIG. 2

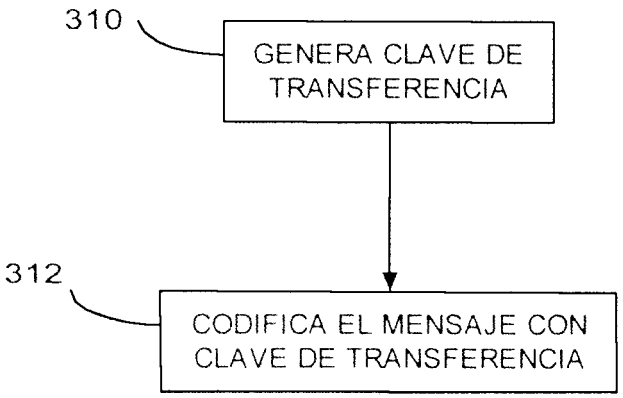
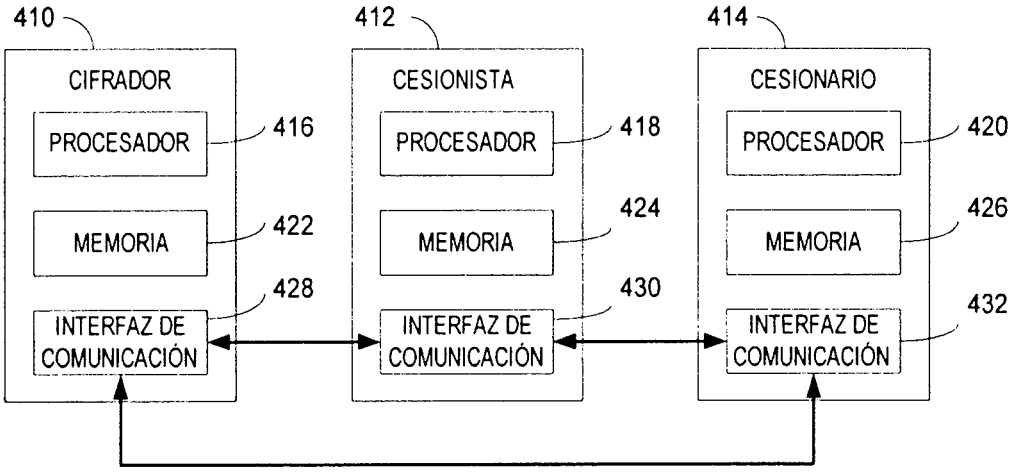

310
GENERA CLAVE DE TRANSFERENCIA
312
CODIFICA EL MENSAJE CON CLAVE DE TRANSFERENCIA

FIG. 3

FIG. 4


410
CIFRADOR
PROCESADOR
416
MEMORIA
422
INTERFAZ DE COMUNICACIÓN
428
412
CESIONISTA
PROCESADOR
418
MEMORIA
424
INTERFAZ DE COMUNICACIÓN
430
414
CESIONARIO
PROCESADOR
420
MEMORIA
426
INTERFAZ DE COMUNICACIÓN
432

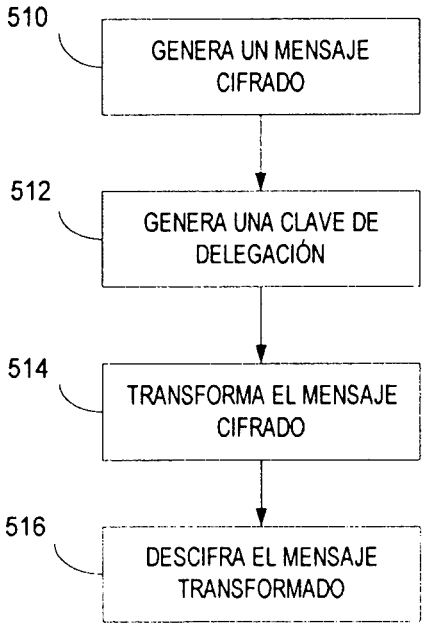

510
GENERA UN MENSAJE CIFRADO
512
GENERA UNA CLAVE DE DELEGACIÓN
514
TRANSFORMA EL MENSAJE CIFRADO
516
DESCIFRA EL MENSAJE TRANSFORMADO

FIG. 5

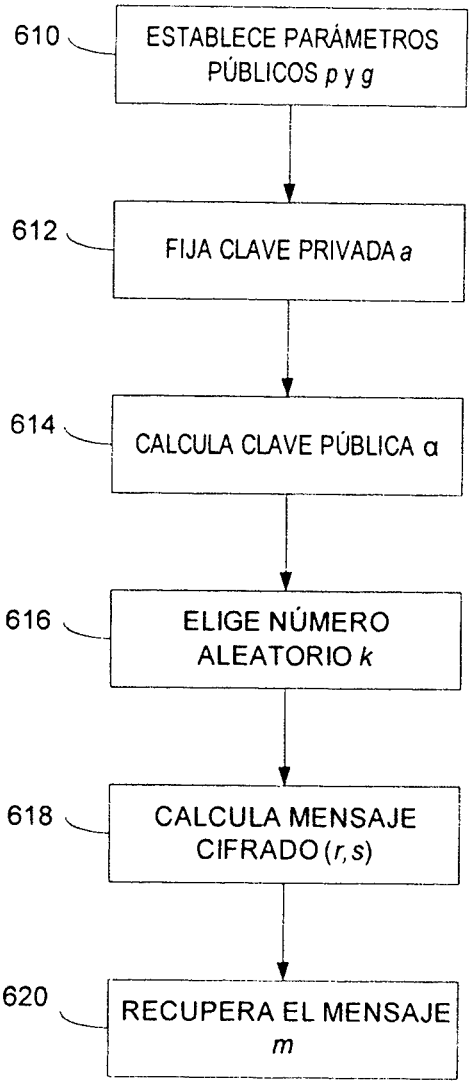

610
ESTABLECE PARÁMETROS PÚBLICOS p y g
612
FIJA CLAVE PRIVADA a
614
CALCULA CLAVE PÚBLICA α
616
ELIGE NÚMERO ALEATORIO k
618
CALCULA MENSAJE CIFRADO (r, s)
620
RECUPERA EL MENSAJE m

FIG. 6

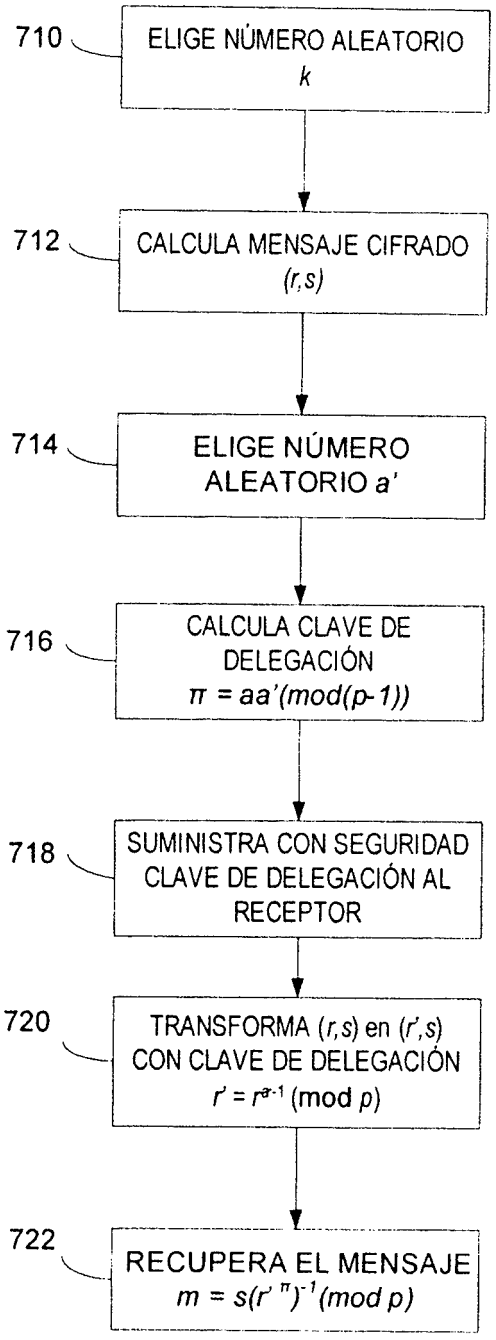

710
ELIGE NÚMERO ALEATORIO
k
712
CALCULA MENSAJE CIFRADO
(r,s)
714
ELIGE NÚMERO ALEATORIO a'
716
CALCULA CLAVE DE DELEGACIÓN
π = aa'(mod(p-1))
718
SUMINISTRA CON SEGURIDAD CLAVE DE DELEGACIÓN AL RECEPTOR
720
TRANSFORMA (r,s) en (r',s) CON CLAVE DE DELEGACIÓN
r' = r^a-1 (mod p)
722
RECUPERA EL MENSAJE
m = s(r'^π)^-1(mod p)

FIG. 7

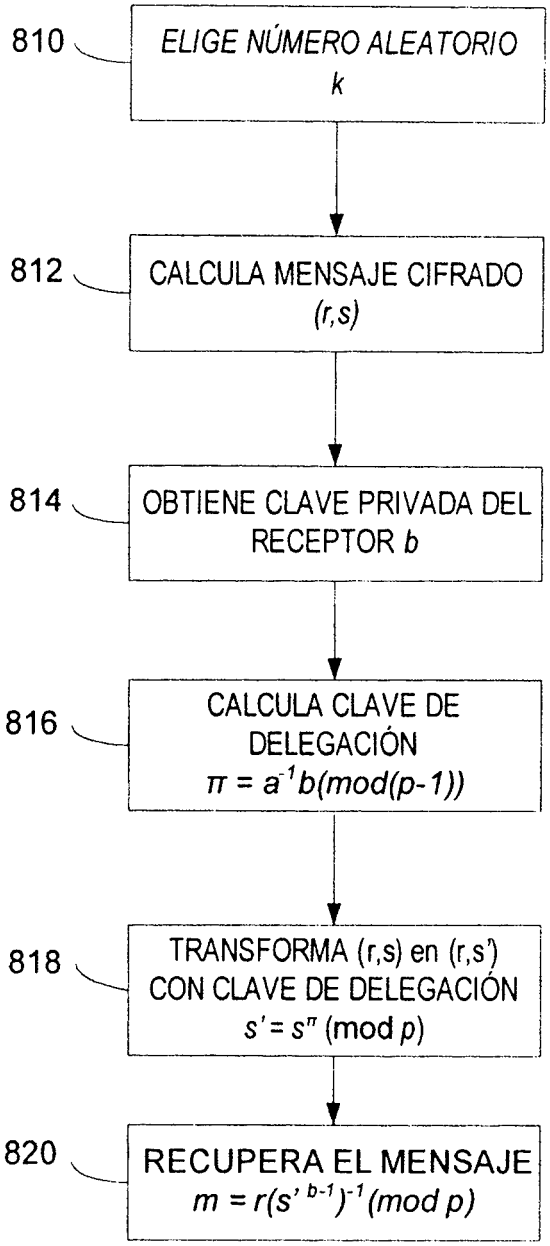

810
ELIGE NÚMERO ALEATORIO
k
812
CALCULA MENSAJE CIFRADO
(r,s)
814
OBTIENE CLAVE PRIVADA DEL
RECEPTOR b
816
CALCULA CLAVE DE
DELEGACIÓN
π = a-1b(mod(p-1))
818
TRANSFORMA (r,s) en (r,s')
CON CLAVE DE DELEGACIÓN
s' = sπ (mod p)
820
RECUPERA EL MENSAJE
m = r(s' b-1)-1(mod p)

FIG. 8

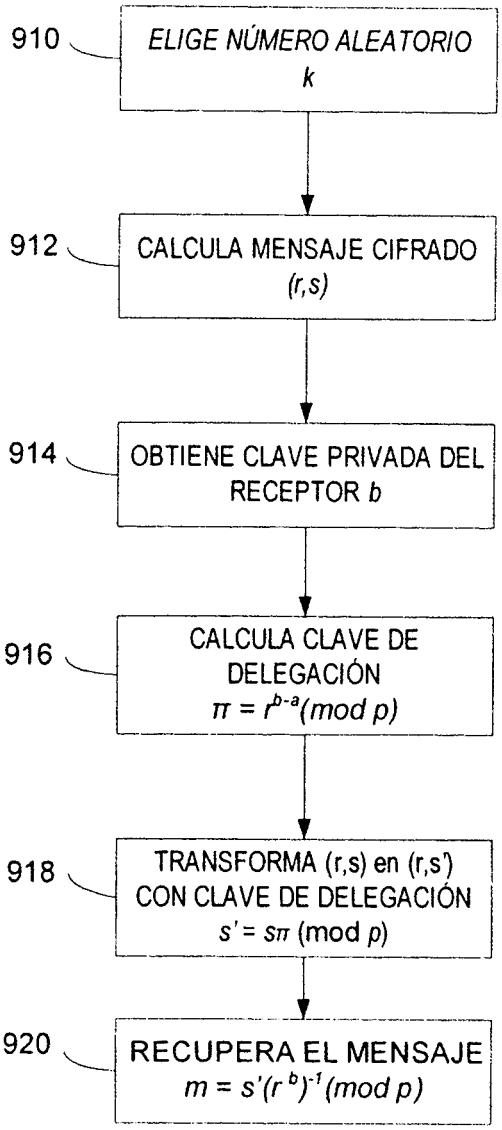

910
ELIGE NÚMERO ALEATORIO
k
912
CALCULA MENSAJE CIFRADO
(r,s)
914
OBTIENE CLAVE PRIVADA DEL
RECEPTOR b
916
CALCULA CLAVE DE
DELEGACIÓN
π = r^(b-a)(mod p)
918
TRANSFORMA (r,s) en (r,s')
CON CLAVE DE DELEGACIÓN
s' = sπ (mod p)
920
RECUPERA EL MENSAJE
m = s'(r^b)^-1(mod p)

FIG. 9

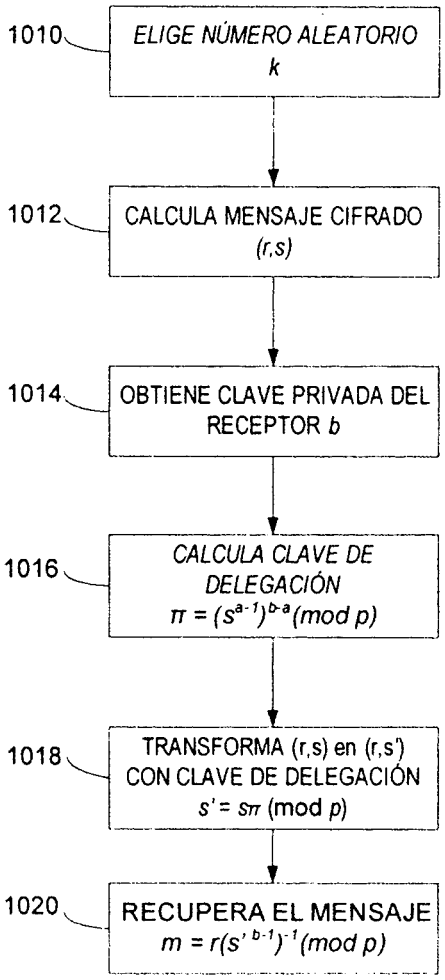

1010
ELIGE NÚMERO ALEATORIO
k
1012
CALCULA MENSAJE CIFRADO
(r,s)
1014
OBTIENE CLAVE PRIVADA DEL RECEPTOR b
1016
CALCULA CLAVE DE DELEGACIÓN
$\pi = (s^{a-1})^{b-a} (mod\ p)$
1018
TRANSFORMA (r,s) en (r,s') CON CLAVE DE DELEGACIÓN
$s' = s\pi\ (mod\ p)$
1020
RECUPERA EL MENSAJE
$m = r(s'^{b-1})^{-1} (mod\ p)$

FIG. 10

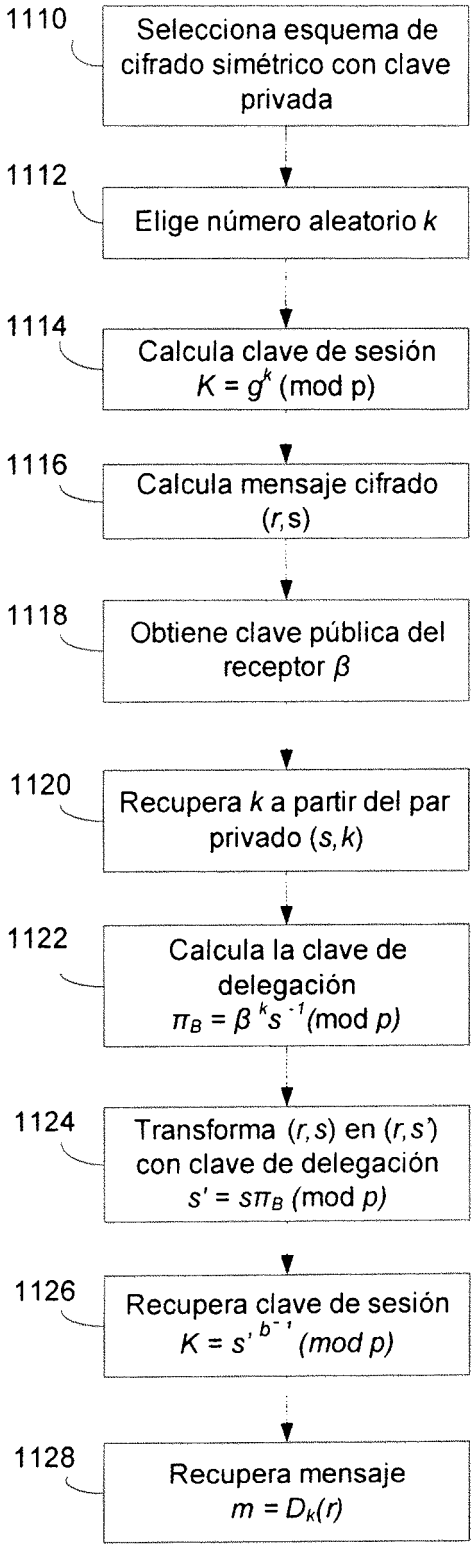

1110
Selecciona esquema de cifrado simétrico con clave privada
1112
Elige número aleatorio k
1114
Calcula clave de sesión $K = g^k \pmod p$
1116
Calcula mensaje cifrado (r,s)
1118
Obtiene clave pública del receptor β
1120
Recupera k a partir del par privado (s,k)
1122
Calcula la clave de delegación $\pi_B = \beta^k s^{-1} \pmod p$
1124
Transforma (r,s) en (r,s') con clave de delegación $s' = s\pi_B \pmod p$
1126
Recupera clave de sesión $K = s'^{b^{-1}} \pmod p$
1128
Recupera mensaje $m = D_k(r)$

FIG. 11

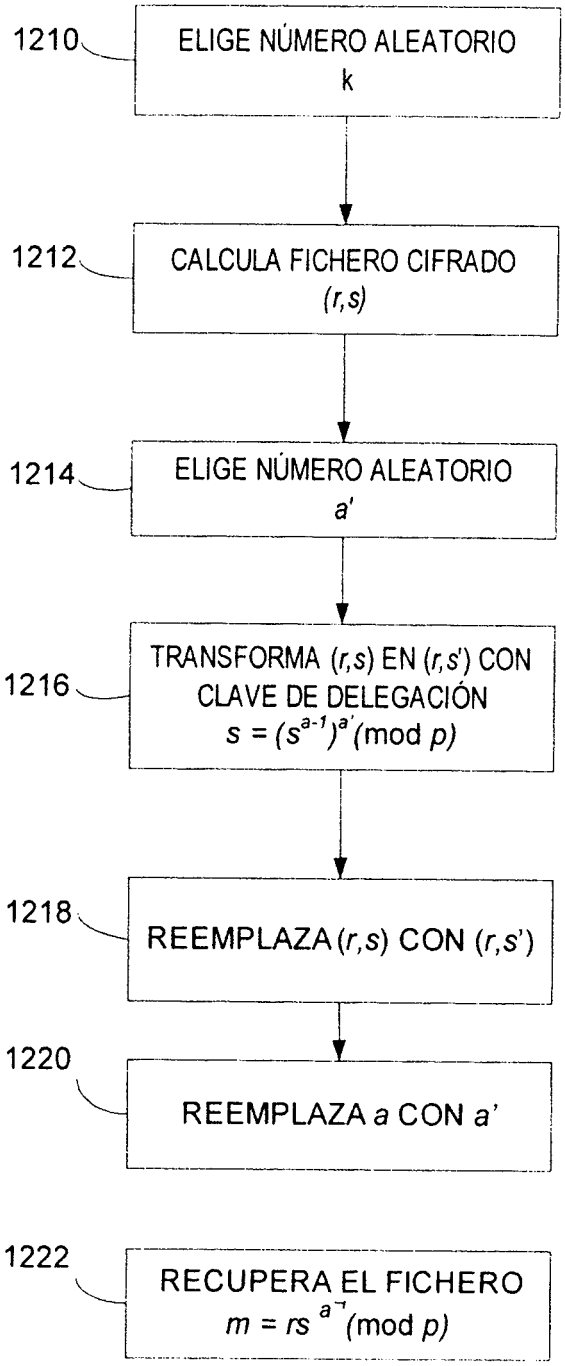

1210
ELIGE NÚMERO ALEATORIO
k
1212
CALCULA FICHERO CIFRADO
(r,s)
1214
ELIGE NÚMERO ALEATORIO
a'
1216
TRANSFORMA (r,s) EN (r,s') CON
CLAVE DE DELEGACIÓN
s = (s^a-1)^a' (mod p)
1218
REEMPLAZA (r,s) CON (r,s')
1220
REEMPLAZA a CON a'
1222
RECUPERA EL FICHERO
m = rs^a^-1 (mod p)

FIG. 12

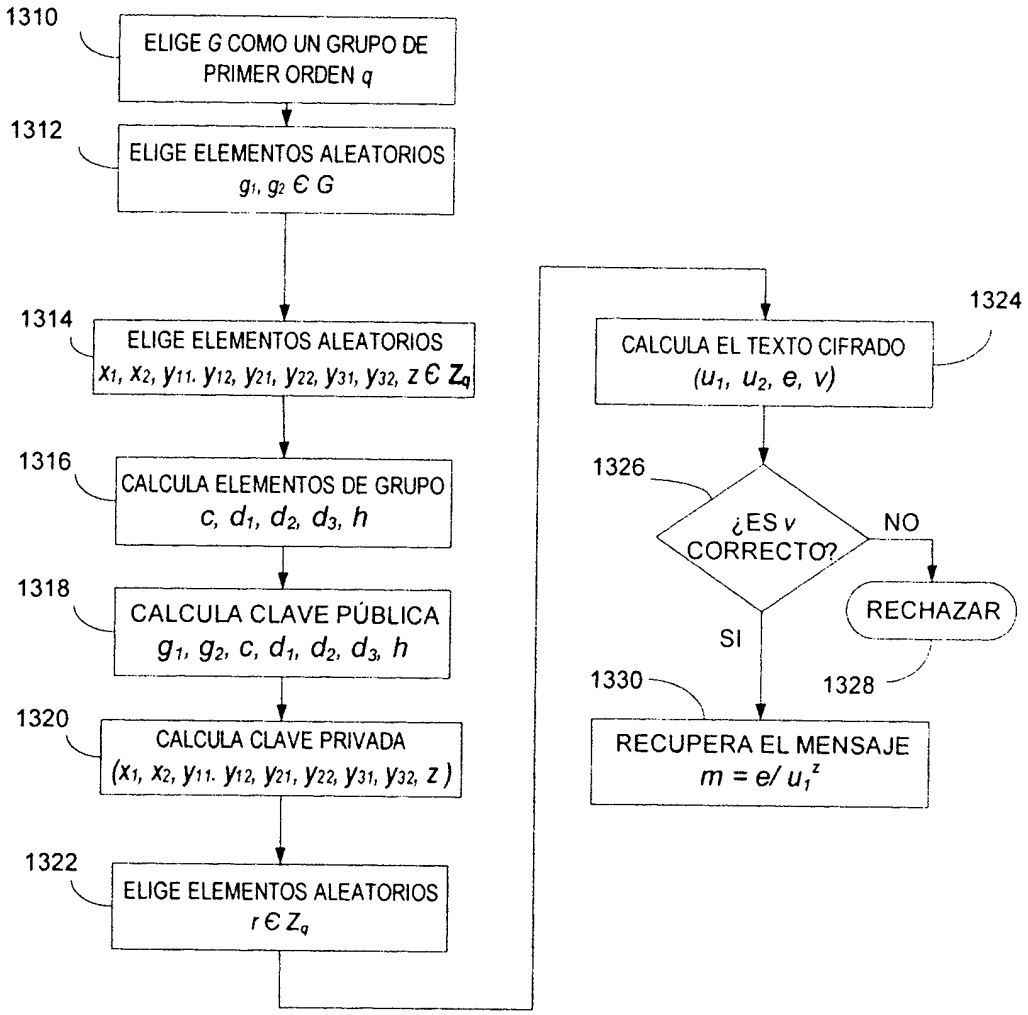

1310
ELIGE G COMO UN GRUPO DE PRIMER ORDEN q
1312
ELIGE ELEMENTOS ALEATORIOS $g_1, g_2 \in G$
1314
ELIGE ELEMENTOS ALEATORIOS $x_1, x_2, y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}, z \in Z_q$
1316
CALCULA ELEMENTOS DE GRUPO c, d_1, d_2, d_3, h
1318
CALCULA CLAVE PÚBLICA $g_1, g_2, c, d_1, d_2, d_3, h$
1320
CALCULA CLAVE PRIVADA $(x_1, x_2, y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}, z)$
1322
ELIGE ELEMENTOS ALEATORIOS $r \in Z_q$
1324
CALCULA EL TEXTO CIFRADO (u_1, u_2, e, v)
1326
¿ES v CORRECTO?
NO
RECHAZAR
1328
SI
1330
RECUPERA EL MENSAJE $m = e / u_1^z$

FIG. 13