(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 613 881**

(21) Número de solicitud: 201630804

(51) Int. CI.:

H04L 9/14 (2006.01)

(12) PATENTE DE INVENCIÓN B1

(22) Fecha de presentación:

13.06.2016

(43) Fecha de publicación de la solicitud:

26.05.2017

Fecha de la concesión:

22.03.2018

(45) Fecha de publicación de la concesión:

02.04.2018

(73) Titular/es:

DIAZ BAÑO, Alvaro (50.0%)
GRAN VIA DE LES CORTS CATALANES, 996, 4º 2ª
08018 BARCELONA (Barcelona) ES y
DIAZ BAÑO, Pablo (50.0%)

(72) Inventor/es:

DIAZ BAÑO, Alvaro y
DIAZ BAÑO, Pablo

(54) Título: **MÉTODO HÍBRIDO DE CIFRADO Y DESCIFRADO DE DOCUMENTOS ELECTRÓNICOS**

(57) Resumen:
Método híbrido de cifrado y descifrado de documentos electrónicos.
Se genera una clave con un algoritmo aleatorio de generación de números (20); Se cifra el documento empleando la clave y un algoritmo simétrico (31); Se descifra el documento y se obtiene su hash comparándolo con el primer hash; si el producto a cifrar asimétricamente es mayor que la clave pública, se divide en bloques enteros de bytes; la clave simétrica o los bloques son cifrados con un algoritmo asimétrico (32) y una o varias claves públicas; un identificador de aplicación es etiquetado para ser encapsulado; la información para revertir el proceso es etiquetada y encapsulada en el documento cifrado; se comprueba el identificador de aplicación; se extrae la información encapsulada; con el par de clave privada se descifra la clave simétrica o sus bloques; los bloques son unidos; con la clave simétrica se descifra el documento.

ES 2 613 881 B1

D E S C R I P C I Ó N

MÉTODO HÍBRIDO DE CIFRADO Y DESCIFRADO DE DOCUMENTOS ELECTRÓNICOS

SECTOR DE LA TÉCNICA

La presente invención pertenece al sector de la seguridad informática. La presente invención se refiere a un nuevo procedimiento que hace posible el cifrado híbrido sin límite del número o del tamaño de las claves asimétricas empleadas, y desarrolla nuevos procedimientos que son mejoras de la técnica actual haciendo posible: asociar los documentos electrónicos procesados, con las aplicaciones informáticas que los procesan; generar automáticamente claves simétricas sin límite de tamaño; realizar un control de calidad del cifrado mediante la obtención y comparación de huellas digitales hash; incluir y encapsular en el documento electrónico cifrado toda la información criptográfica necesaria para revertir el proceso de cifrado; codificar de forma segura el documento electrónico cifrado para su envío por internet; utilizar claves públicas que no están sometidas a periodos de vigencia limitados, pero sí que están asociadas a claves privadas almacenadas en token de hardware criptográficos de sistemas PKI.

ANTECEDENTES DE LA INVENCIÓN

Existe extensa documentación que describe sistemas de cifrado híbridos, no obstante todos ellos padecen importantes limitaciones que esta invención resuelve definitivamente. Las limitaciones son:

1.- En criptografía asimétrica el documento que se va a cifrar no puede ser mayor que el par de clave pública que se va a utilizar, en el caso de esta invención el documento a cifrar asimétricamente es la clave simétrica. Si una clave simétrica es cifrada por una asimétrica, el producto resultante es de tal tamaño que generalmente ya no puede volver a ser cifrado por otras claves asimétricas, salvo que la clave simétrica sea de tan reducida longitud que no puede ser calificada criptográficamente segura.
La limitación de la técnica actual se evidencia cuando se desea cifrar con varias claves públicas una clave simétrica de un tamaño que pueda ser considerado criptográficamente seguro. Por ejemplo, una clave simétrica de 256 bits al ser cifrada

con una clave pública de 2048 bits, genera una clave simétrica cifrada de 256 bytes (1 byte = a 8 bits), es decir 2048 bits. Si la clave pública fuera de 1024 bits y la simétrica de 128 bits, se obtendría una clave cifrada de 128 bytes, es decir 1024 bits. Por lo tanto, el producto cifrado resultante ya no se puede volver a cifrar con claves de similar longitud.

Existen numerosos entornos en los que el requerimiento de múltiples operadores es necesario, por ejemplo para controlar el acceso a documentos electrónicos altamente confidenciales, en los que se requiere la presencia de dos o más personas autorizadas.

2.- Otra de las limitaciones que presenta el estado actual de la técnica, viene igualmente derivada de la limitación de tamaño. Aunque se documenta en la Patente CN102404120 (A) la obtención automática de claves simétricas mediante algoritmos de generación de números aleatorios, estos algoritmos se restringen específicamente a 32 bits, un tamaño considerado criptográficamente como inseguro. Es de gran interés la posibilidad de obtener automáticamente, mediante el uso algoritmos de generación aleatoria de números, claves simétricas sin limitación de tamaño, puesto que al tratarse de un cifrado híbrido no es necesario que el usuario conozca la clave de cifrado simétrico empleada para cifrar el documento, de esta forma la protección de los documentos se puede realizar sin intervención humana.

3.- Una nueva limitación de los sistemas de cifrado híbridos actuales, es que solo son capaces de encapsular una única clave simétrica cifrada, no existe ningún procedimiento que permita encapsular productos cifrados por múltiples claves asimétricas, o añadir otra información necesaria para revertir este proceso de cifrado. Al encapsular exclusivamente los bytes de cifrado, la información debe de tener siempre la misma longitud, lo cual condiciona el propio proceso de cifrado.

4.- Otra de las limitaciones que presenta la técnica actual, es que solo ofrece dos modalidades de uso para acceder a una clave pública, cada una de ellas con importantes carencias funcionales:

a) Pretty Good Privacy (PGP). Este programa permite la generación de un par de claves, pública y privada, sin que la clave pública tenga que estar asociada a un certificado electrónico. El problema radica en que el sistema no es compatible con los actuales token hardware criptográficos. Esto presupone una importante carencia funcional dado que solo estos token garantizan que la clave privada es inaccesible e iduplicable.

b) Infraestructuras de Clave Pública (PKI). Este sistema aunque es compatible con la práctica totalidad de token hardware criptográficos, requiere la emisión del certificado electrónico x509v3, lo cual plantea un grave problema funcional. La legislación internacional en materia de certificados electrónicos y las normas técnicas relativas a los mismos, desde la RFC 2459 a la RFC 5280 entre otras, determinan la obligación de establecer un periodo de vigencia, de hecho el titular del certificado suele firmar un documento comprometiéndose a cesar en el uso de la clave privada a partir del momento en que caduca el certificado. En un entorno en el que la clave privada es necesaria para poder acceder a la información a lo largo del tiempo, el problema tiene un alcance extremadamente grave.

En ambos supuestos, cada clave pública está almacenada en un único contenedor, lo cual presupone una limitación en procesos automatizados al tener que manejar tantos contenedores como claves públicas intervienen.

5.- Una carencia que presentan los procesos de cifrado es el control de la calidad del cifrado, que pese a su importancia es prácticamente inexistente. Es esencial garantizar la recuperación del documento sin que el mismo pierda su esencia guardando su total integridad respecto al original. Son numerosos los factores que pueden ocasionar la corrupción de un documento al ser cifrado, desde un error en la codificación, hasta un fallo fortuito de hardware. Muchos de estos fallos son indetectables en el momento del cifrado, se afloran cuando se desea recuperar el documento electrónico.

6.- Otra carencia es la transferencia de documentos electrónicos cifrados a través de Internet, generalmente se emplea la técnica de codificarlos previamente a Base 64, pero esta codificación aunque altamente fiable presenta también importantes carencias, tal y como reconoce la recomendación internacional RFC 1521 de la organización internacional Internet Engineering Task Force (IETF), que en su apartado 7.1.1., indica que se debe de prestar máxima atención a los caracteres multibyte (MBCS), p.ej. caracteres chinos, o el kanji chino, que entre otros muchos idiomas no pueden ser representados por 8 bites, invariablemente requieren 16 bits. Y se deja para futuras revisiones resolver este problema.

7.- Otra carencia que presentan los procesos de cifrado es que la aplicación empleada y los documentos por ella procesados no están íntimamente asociados. En contextos de alta seguridad es esencial tener un control de las aplicaciones informáticas que pueden ser empleadas para acceder a la información.

Se ha realizado un amplio estudio sobre las patentes existentes. Ninguna de ellas describe los métodos reivindicados en esta invención. Y alguna de las analizadas no hacen sino confirmar las limitaciones que padece el estado de la técnica actual. Concretamente:

La publicación “Moderne Verfahren der Kryptographie” (“Procedimientos modernos de criptografía”) Beutelspacher, Schwenk, Wolfenstetter, 3. Edición, 1999, Vieweg Verlag, contiene una descripción detallada de los procedimientos criptográficos de clave pública.

El sistema criptográfico RSA que se describe en la Patente de EE.UU. Nº 4.405.829 concedida a Rivest y otros describe un ejemplo de metodología de un sistema criptográfico de clave pública.

La Patente CN102404120 (A) - Encryption method and encryption system for electronic documents
Que incluye un modulo de generación aleatoria de números, que son utilizados para cifrar simétricamente un documento electrónico; la clave simétrica es cifrada con una clave pública que obtiene de un certificado electrónico, único procedimiento posible según la descripción que se realiza en la patente; edita el documento electrónico e incluye en él la clave simétrica cifrada, sin ningún orden ni estructura especifico.
La Reivindicación 1, limita expresamente el algoritmo de generación de números aleatorio al tamaño de 32 bits, es decir una clave simétrica de 4 bytes, algo que critográficamente se considera en la actualidad un cifrado de baja seguridad. Es evidente que los autores en el momento de la redacción, eran conscientes del problema de la longitud de claves que tiene la criptiografía simétrica, y no lograron resolver esta limitación.

EXPLICACIÓN DE LA INVENCIÓN

Con el fin de alcanzar los objetivos y evitar los inconvenientes mencionados en los apartados anteriores, esta invención desarrolla los siguientes procedimientos:

1.- La limitación de tamaño de la clave simétrica respecto a la clave pública, y la limitación de emplear múltiples claves públicas en un cifrado por capas, se resuelve

mediante el procedimiento de calcular antes de cada proceso de cifrado el tamaño del producto a cifrar asimétricamente y, si es de mayor tamaño que la clave asimétrica, el producto a cifrar se divide en bloques enteros de bytes o, incluso, según necesidad, esos bloques se pueden subdividir con el fin de que el producto a cifrar sea siempre de menor tamaño que la clave pública que se va a emplear, no existe limitación de longitud o numero de claves a emplear. Para revertir el proceso, primero se descifra cada bloque siguiendo el orden inverso al del cifrado, y empleando el par de clave privada asociada a la clave pública que se empleó para cifrar; una vez descifrado cada grupo de bloques, se van uniendo siguiendo el orden inverso en que fueron subdivididos o divididos hasta obtener la clave simétrica. Todos los lenguajes de programación tienen la capacidad de editar, dividir y subdividir en bloques enteros de bytes una clave simétrica, para el cifrado asimétrico se emplea un algoritmo de cifrado asimétrico y el mismo algoritmo para descifrar, existen numerosos algoritmos de libre uso, p.ej. RSA, ElGamal, DSA, etc., extensamente documentados en Internet;

2.- Una vez resuelta la limitación de tamaño de la clave simétrica, esta invención presenta la novedad de generar automáticamente, empleando un algoritmo de generación de números aleatorios, una clave simétrica sin limitación de tamaño; existen múltiples algoritmos que permiten obtener claves seguras, por ejemplo de tamaño igual o superior a los 128 bits, extensamente documentados en Internet p. ej. Blum Blum Shub, Fortuna, o Mersenne twister, además la práctica totalidad de token de hardware criptográficos integran una función de generación aleatoria de números que también puede ser utilizada;

3.- Para poder revertir el sofisticado proceso de cifrado de esta invención, se utilizan etiquetas identificativas que posibilitan estructurar la información necesaria para descifrar el documento electrónico, y que actúan como separadores de datos; el conjunto de información y etiquetas conforma uno o varios contenedores de datos que son encapsulados en el cuerpo del documento electrónico cifrado. Los contenidos informativos son asociados a etiquetas identificativas, e incluso se pueden incluir etiquetas que no tienen asociado contenido informativo, p.ej. inicio y final del encapsulado. Este procedimiento permite desarrollar una lógica de proceso que no solo trata cada información de forma independiente, sino que además convierte las etiquetas en separadores lógicos, que permiten encapsular la información en cualquier parte del documento cifrado y posteriormente extraer los bytes exactos de la información encapsulada. Este procedimiento es extremadamente versátil, no tiene limitación alguna en cuanto al formato, longitud, número, o código de identificación de

las etiquetas, la estructura que se considere necesaria según el tipo de cifrado a realizar es implementada en una lógica informática, siendo de esta forma capaz de automatizar la generación de contendedores de datos y su encapsulado; la información al estar estructurada puede ser consultada y los datos necesarios empleados para revertir fácilmente el proceso criptográfico realizado; todos los lenguajes de programación tienen la capacidad de editar documentos, implementar la lógica de etiquetado de información, crear contenedores de datos y encapsularlos.

4.- Esta invención define un nuevo procedimiento que almacena los bytes de una clave pública en un contenedor de información estructurada, junto con el identificador que comparte con su par de clave privada. Para ello se ha desarrollado un procedimiento que partiendo del habitual sistema de elaboración de un certificado x509v3, se interrumpe el proceso en un paso previo, concretamente en el momento de creación del fichero en formato PKCS#10, también llamado certificado autofirmado o de petición; en ese paso la clave privada ya ha sido generada, y está contenida en un token de hardware criptográfico o en almacén de claves por software en el que la clave privada está asociada con un identificador que comparte con la clave pública; este fichero PKCS#10 es editado y su información es cargada en un contenedor de información estructurada, p.ej. un XML, que permite guardar como mínimo: [KEY] bytes de la clave pública; [ID] identificador que comparte con su par de clave privada ; además puede almacenar otros datos como:[TYPE] tipo de clave; [SIZE] tamaño de la clave en Bits. Una peculiaridad de la información criptográfica del fichero PKCS#10, es que aun no se ha sometido a periodo de vigencia, por lo tanto, se puede utilizar indefinidamente. La práctica totalidad de lenguajes de programación tienen la capacidad de crear contenedores de información estructurada como un XML, y pueden utilizar criptosistemas que incluyen esta funcionalidad, son libre uso y están extensamente documentados en internet como Open SSL o Bouncy Castell;

5.- Este método incorpora un control de calidad del cifrado, basado en la generación y comparación de resúmenes hash. Utilizando un algoritmo de digestión que se aplica sobre el documento electrónico original, se obtiene un primer hash que puede ser etiquetado y encapsulado en el cuerpo del documento cifrado, después de cada descifrado se obtiene un hash del documento descifrado, el cual es comparado con el primer hash para determinar su integridad; existen diversos algoritmos de digestión o resumen hash de libre uso ampliamente documentados en Internet, p. ej. SHA1, SHA256, entre otros,

6.- Se ha creado una tabla de conversión que contiene todos los caracteres

reconocidos de forma estándar por la codificación Base64, esta tabla además contiene caracteres no reconocidos en Base64 habiendo incluido equivalencias que se caracterizan por ser de 16 bits; esta tabla permite realizar una codificación segura que garantiza la transmisión por internet del documento cifrado y encapsulado sin correr el riesgo de que se corrompa; el procedimiento lee todos los caracteres contenidos en el documento electrónico cifrado y encapsulado, si alguno de los caracteres no está incluido en la tabla de conversión, el documento no será codificado evitando así su corrupción, y si todos los caracteres son reconocidos aunque alguno de ellos no tenga correspondencia con el estándar Base64, es posible codificarlo con la equivalencia de 16 bits implementada en la tabla. Todos los lenguajes de programación pueden implementar este procedimiento sin que ello presuponga reto alguno relevante.

7.- Se ha desarrollado un procedimiento por el cual la aplicación informática que se emplea para procesar el documento dispone de un identificador o clave de uso, cada documento procesado por esta aplicación incorpora el identificador cifrado o su hash en una etiqueta del contenedor de datos que es encapsulado en el documento electrónico cifrado. Este identificador o clave de uso asocia íntimamente el documento procesado con una determinada aplicación licenciada, la cual a su vez puede estar instalada en n equipos informáticos que conforman una infraestructura asociada por ese elemento común, este identificador puede ser determinado a voluntad del administrador que gestiona la infraestructura, o por el fabricante que la comercializa, o ser generada automáticamente a partir de otro valor que compartan todos los equipos informáticos, p.ej. nombre de la organización, razón de IP, etc. no existe otra limitación que el de seleccionar un identificador que sea univoco, y tampoco existe restricción para almacenar el identificador, el cual puede ser grabado en una posición de memoria del equipo en el que se instala la aplicación, en un fichero electrónico, en el propio código de la aplicación, etc. no existe otra limitación que el seleccionar una ubicación específica. El procedimiento contempla la posibilidad de que el identificador cifrado o el hash ya estén procesados y almacenados, de esta forma la aplicación no tiene que realizar en cada ocasión la transformación, con el consiguiente ahorro de proceso. En cualquier supuesto, el hash se obtiene aplicando al identificador un algoritmo de digestión, y para el cifrado del identificador se puede emplear un algoritmo de cifrado simétrico o asimétrico y clave que como única limitación es que deben de ser siempre la misma. Previo al proceso de descifrado, la aplicación informática determina si el identificador de la aplicación está asociado al identificador incluido en el documento cifrado, para ello edita el contenedor de datos y obtiene la etiqueta e información

etiqueta e información asociada al identificador; si el identificador está cifrado debe de descifrarlo empleando la misma clave caso de ser simétrica, o clave privada asociada al par de pública empleada para cifrar, caso de ser cifrado asimétrico, utilizando siempre el mismo algoritmo que los cifró, se procede a comparar el identificador descifrado con el identificador de aplicación; si se trata del hash del identificador aplicará un algoritmo de digestión sobre el identificador de la aplicación, y comparará el hash obtenido con el hash etiquetado en el contenedor de datos del documento cifrado. La realización de estas comparaciones no presupone reto alguno, y sigue los mismos procesos criptográficos que los ya descritos anteriormente.

8.- Para cifrar el documento emplea una clave simétrica sin limitación de tamaño y un algoritmo simétrico, existen numerosos algoritmos de cifrado simétrico de libre uso, extensamente documentados en Internet, su uso no presupone reto alguno, p.ej. CAST, IDEA, TripleDES, AES, etc; y criptosistemas que facilitan su uso como Open SSL o Bouncy Castell; para descifrar se emplea la misma clave simétrica y el algoritmo simétrico empleado para cifrar.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

Para complementar la descripción que se está realizando y con objeto de ayudar a una mejor comprensión de las características de la invención, se acompaña como parte integrante de dicha descripción un juego de dibujos, en los que se ha representado lo siguiente:

Figura 1 ilustra un esquema de cifrado simétrico.

Figura 2 ilustra un esquema de cifrado asimétrico.

Figura 3 ilustra un esquema de generador de números aleatorios, según patente de invención Número de Solicitud: P200702299.

Figura 4 ilustra un esquema de encapsulado de datos con separadores.

REALIZACIÓN PREFERENTE DE LA INVENCIÓN

Una primera realización preferente del sistema aquí descrito, comprende esencialmente, los siguientes elementos:

a) Una Terminal informática, que dispone de una memoria de almacenamiento de

datos no volátil (1). Se ha conectado a este terminal un token criptográfico PKI, también conocido como dispositivo de firma electrónica HSM (modulo de seguridad hardware (5), con conector USB que ensambla un procesador criptográfico del fabricante ST Microelectrónics. En el ejemplo de aplicación preferente se utiliza un ordenador marca HP modelo Pavilion con procesador Intel® Atom™ 2 Z8300, y con sistema operativo Windows 8, con disco duro SATA de 500 Gb.

La unidad de almacenamiento (1) contiene un fichero PKCS#10 (2) con la clave pública (7) etiquetada con un identificador (19) que es compartido con la clave privada (6); una API PKCS#11 (16) del fabricante ST Microelectronics ; un documento electrónico (8); un XML que contiene el identificador (29) del programa de usuario (4) introducido por el propio usuario; un programa de usuario (4); un criptosistema (3), Bouncy Castle, el cual ofrece una amplia colección de API's que contienen los principales algoritmos para procesos criptográficos simétricos (31) y asimétricos (32), e incluso implementa la capacidad para hacer llamadas según el estándar PKCS#11 (16), leer el contenido de certificados electrónicos X509, leer ficheros PKCS#10 (2), y store software según estándar PKCS#15 que contienen clave privada, además dispone de algoritmos de generación aleatoria de números (20) de tamaño igual o superior a 128 bits. También dispone de algoritmos de digestión (25) para la obtención de hash. Dispone de versión para Java.

b) Un programa de usuario (4). Se ha desarrollado una aplicación en Java. Este software contiene un interface de usuario y toda la lógica requerida:

- Interface que permite al usuario navegar por la unidad de almacenamiento (1), y seleccionar un documento electrónico independientemente de su estado, original (8) o cifrado (12), o encapsulado 15).
- Navegar por la unidad de almacenamiento (1) y seleccionar un fichero PKCS#10 (2).
- Hacer llamadas al criptosistema (3) para todos los procesos criptográficos contemplados en esta invención, incluso llamadas a la librería PKCS#11 (16) para acceder al token criptográfico (5)
- Capacidad para crear, alimentar y consultar una tabla de conversión (34) que tiene al menos dos campos: campo carácter y campo equivalencia hasta 16 bits. Esta tabla (34) es alimentada con todos los caracteres que son reconocidos de forma estándar por la codificación Base 64, si han añadido

caracteres multibyte (MBCS), que no son reconocidos por esta codificación Base 64

- Capacidad para editar, leer y contar los bytes que componen una clave simétrica y una clave asimétrica. Incluyendo la lógica necesaria para valorar, según el proceso de cifrado requerido en cada ocasión, si los bytes que se desean cifrar asimétricamente tienen un tamaño mayor que la clave pública que se desea emplear, determinando en cada supuesto si se debe o no dividir en bloques el producto a cifrar para obtener un tamaño inferior que la clave pública. Y capacidad de revertir la operación de división de bloques realizada. Este es un proceso elemental, disponible en cualquier lenguaje de programación de uso común.
- Introducir un PIN (10) y una clave simétrica (9)
- Capacidad para comparar los bytes varios hash y determinar si son idénticos o distintos.
- Editar un documento electrónico cifrado (12) e introducir el contenedor de datos (21) que contiene la información de los procesos criptográficos llevados a cabo, y que han sido etiquetados (13) conforme a la lógica definida en esta invención:
  1.- cada contenido informativo tiene está asociado a una etiqueta identificativa (13), y
  2.- pueden haber etiquetas (13) que no tengan contenido informativo.
  Se ha elaborado la siguiente estructura de etiquetas (26) para este ejemplo de aplicación práctica:
  [INICIO]
  [H] Hash del documento original
  [O] cantidad de pares de operadores
  [O1]: identificador (19) del operador 1 del par i (i<=O)
  (O1 bytes): los bytes de la clave pública (7) del operador 1
  [O2]: identificador (19) del operador 2 del par i (i<=O)
  (O2 bytes): los bytes de la clave pública (7) del operador 2
  [P] (entero de 4 bytes): cantidad de partes de 64 bytes del cifrado hecho con el operador 1
  [T] (entero de 4 bytes): el tamano del cifrado hecho con el operador 1
  (P * T bytes) : cada parte de 64 bytes se cifra con el operador 2 y se escribe una parte cifrada de tamaño T
  [A] AES

[A2] RSA

[FIN]

Las etiquetas (13) actúan como separadores de la información, a la vez que identifican cada contenido, el conjunto de etiquetas y la información que contienen conforman el contenedor de datos (21).

- Editar el documento con la lógica necesaria para extraer el contenedor de datos (21) del documento electrónico cifrado encapsulado (15), dejando el documento electrónico cifrado (12) libre de bytes del contenedor de datos (21).
- Crear un archivo capaz de contener información estructurada, se ha elegido el formato XML (22). Hacer la llamada correspondiente al criptosistema (3) para acceder a la información de la clave pública (7) contenida en un fichero PKCS#10 (2), y obtener:

  [KEY] bytes de la clave pública, [ID] identificador que comparte con su par de clave privada [TYPE] tipo de clave, [SIZE] tamaño de la clave en Bits. Esta información es almacenada en el contenedor XML (22)
- Capacidad para introducir y leer un string de datos que contiene identificador o clave de uso (29), que está almacenado en un fichero XML.
- Capacidad de comparar los bytes de dos hash, valorando si son idénticos o no.
- Capacidad comparar dos string de datos.

c) Un dispositivo de firma electrónica (5). Se trata de un token HSM (modulo de seguridad harware), con conector USB que ensambla un procesador criptográfico de ST Microelectrónics. Este token dispone de un generador de números aleatorio (20) validado por el NIST, capaz de elaborar números de un tamaño igual o superior a 128 bits, y un contenedor seguro de claves criptográficas, que almacena el par de clave privada (6) asociada a su par de clave pública(7) con un identificador (19) que comparten ambas claves. Además integra algoritmos criptográficos simétricos (31) y asimétricos (32), la lógica necesaria y la capacidad para realizar procesos de cifrado y descifrado sin que la clave privada (6) abandone el contenedor de seguridad. Este token por hardware (5) es interoperable mediante la API PKCS#11 (16) que suministra su fabricante STMicroelectronics.

d) Dos ficheros PKCS#10 (2) que en conformidad con ese estándar internacional, cada fichero contiene su respectiva clave pública (7) asociada a su par de clave privada (6) por el identificador que comparten (19)

e) Un documento electrónico (8) en formato pdf, que contiene como único texto de prueba “Hola mundo”.

Se realiza el siguiente procedimiento:

Cifrado del documento electrónico:

1. Los usuarios (17), conectan en el terminal informático sus respectivos dispositivos de firma electrónica HSM (5), cada uno de ellos contiene una clave privada (6) etiquetada con un identificador (19) que comparte con su par de clave pública (7);

2.- Se disponen de dos ficheros PKCS#10 (2), cada uno de ellos contiene una clave pública (7) etiquetada con un identificador (19) que comparte con su par de clave privada (6), también es posible utilizar un certificado electrónico según estándar X509;

3. Mediante el programa de usuario (4), se selecciona el documento electrónico (8) que se desea cifrar, se selecciona el algoritmo de cifrado simétrico (31) AES, se introduce una clave simétrica (9) de tamaño 256 bits, se especifican los identificadores (19) “prueba1” y “prueba2” de las claves públicas (7) que serán empleadas para cifrar asimétricamente la clave simétrica, se especifica que se requiere cifrado multicapa, y da la orden de iniciar el proceso. El programa de usuario (4) realiza el siguiente proceso:

a) Lee una clave simétrica (9) de tamaño de 256 bits que ha sido introducida por el usuario.

b) Utiliza el criptosistema (3), para que empleando la clave simétrica (9) y habiendo seleccionado el algoritmo simétrico (31) AES, cifre el documento electrónico (8). El criptosistema (3) genera el documento electrónico cifrado (12).

c) Se procede a la lectura de las claves públicas (7) seleccionando aquellas cuyo identificador (19) corresponda a los dos seleccionados por el usuario. Se consulta el tamaño de cada clave pública (2048 bits) (7) y el tamaño de la clave simétrica (256 bits) (9), y teniendo en cuenta que se ha solicitado un cifrado multicapa, evalúa la necesidad de tener que dividir la clave simétrica (9) en bloques enteros de bytes. Confirmando que la clave simétrica, una vez cifrada será dividida en cuatro bloques de 64 bits.

d) La clave simétrica (9) es cifrada con la primera clave pública (7) indicada por el usuario, identificador (19) “prueba1”.

e) La clave simétrica cifrada (11) es editada y dividida en cuatro bloques de 64

bits.

f) Los cuatro bloques (14) obtenidos de la división realizada de la clave simétrica cifrada (11) son cifrados con la segunda clave pública (7) indicada por el usuario, identificador (19) “prueba2”.

g) Se edita la estructura de etiquetas (26) y se carga cada etiqueta con la información relativa al proceso criptográfico realizado:

[INICIO]

[H] Hash del documento original (23)

[H1] Hash (28) del identificador (29) del programa de usuario (4)

[O]: 1 (par de operadores)

[O1]: Prueba1 (19)

(O1 bytes): los bytes de la clave pública (7) del operador 1

[O2]: Prueba2 (19)

(O2 bytes): los bytes de la clave pública (7) del operador 2

[P] 4 bloques de 64 bytes del cifrado hecho con el operador 1

[T] 256 bytes el tamaño del cifrado hecho con el operador 1

(P * T bytes) : cada parte de 64 bytes se cifra con el operador 2 y se escribe una parte cifrada de tamaño T (14)

[A] AES

[A2] RSA

[FIN]

Este conjunto de información conforma el contenedor de datos (21)

h) Se edita el documento electrónico cifrado (12), y se encapsula en su interior el contenedor de datos (21)

Descifrado del documento electrónico cifrado:

1. Los usuarios (17) conectan en el terminal informático sus respectivos dispositivos de firma electrónica HSM (5),

2. Los usuarios (17) mediante el programa de usuario (4), introducen su PIN (10) de activación de clave privada (6) y ordenan al criptosistema (3) que realice el descifrado del documento electrónico cifrado (12) y control de calidad del cifrado. El programa de usuario realiza el siguiente proceso:

a) Edita el documento electrónico cifrado encapsulado (15), y localiza las etiquetas [INICIO] y [FIN].

b) Habiendo localizado las etiquetas [INICIO] y [FIN], procede a extraer el

contenedor de datos (21), obteniendo así el documento electrónico cifrado (12).

c) Edita el contenedor de datos (21) y obtiene los identificadores de clave pública (19), los bloques cifrados (14) y los algoritmos de cifrado empleados.

3. Se ordena al criptosistema que realice el descifrado de los bloques cifrados (14) utilizando el dispositivo de firma (5), para realizar esta operación además de transferir los bloques cifrados (14) se facilita: PIN (10) de activación de la segunda clave privada (6), utilizada al cifrar los bloques, y su identificador de clave “Prueba2” (19).

4. Los bloques descifrados son editados y unidos, obteniendo así la clave simétrica cifrada (11). Se ordena al criptosistema que realice el descifrado de la clave simétrica cifrada (11) utilizando el dispositivo de firma (5), para realizar esta operación además de transferir la clave simétrica cifrada (11) se facilita: PIN (10) de activación de la primera clave privada (6), utilizada al cifrar los bloques, y su identificador de clave “Prueba1” (19).

Se obtiene la clave simétrica (9)

5. Se ordena al criptosistema (3) que proceda al descifrado el documento electrónico cifrado (12), para realizar esta operación se facilita además del documento electrónico cifrado (12), la clave simétrica (9) y el algoritmo de cifrado simétrico (31) empleado. Obteniendo el documento electrónico descifrado (33)

Según una segunda realización del sistema, además de todos los elementos indicados en la primera realización, para mejorar el acceso a las claves públicas se procede a:

1. Complementar el paso 2 del proceso de cifrado, con el siguiente procedimiento:
   a) Mediante el programa de usuario (4), se crea un archivo XML (22) con los TAG: [KEY], [ID],[TYPE], [SIZE], y ordena al Criptosistema (3) la lectura de los datos contenidos en dos ficheros PKCS#10 (2). La información contenida en esos dos ficheros PKCS#10 (2) es cargada en los TAG del archivo XML (22): [KEY] bytes de la clave pública (7), [ID] “prueba1” “prueba2” identificador que comparte con su par de clave privada (19) [TYPE] “RSA” tipo de clave [SIZE] “2048” tamaño de la clave en Bits.

Según una tercera realización del sistema, además de todos los elementos indicados en la primera realización, para garantizar la calidad del cifrado, se procede a:

1. Introducir un paso previo (a) al Paso 3 del proceso de cifrado, con el siguiente procedimiento:
   a) Solicita al criptosistema (3) que obtenga un hash (23) del documento electrónico original (8) aplicando el algoritmo de digestión (25) SHA256.

2. Introducir unos paso previos (a) (b) y (c) al Paso 3 c) del proceso de cifrado, con el siguiente procedimiento:
   a) Solicita al criptosistema (3) que utilizando la clave simétrica (9) y el algoritmo simétrico (31) AES empleado para cifrar documento electrónico original (8) descifre el documento (12).
   b) Solicita al criptosistema (3) que obtenga un hash (24) del documento electrónico descifrado, utilizando el mismo algoritmo de digestión (25) SHA256 que se empleo para obtener el primer hash (23).
   c) Se compara el primera hash (23) obtenido del original, con el hash (24) obtenido del documento descifrado (33). Si el resultado es que son idénticos, se considera que el proceso de cifrado y descifrado ha sido correcto. Si por el contrario los hash son distintos, el documento descifrado (33) no es integro.
3. Si el proceso de cifrado y descifrando ha sido correcto, en el paso 3 g) carga el hash (23) en la etiqueta [H]
4. Introduce un último paso 6 al proceso de descifrado, con el siguiente procedimiento:
   a) Se solicita al criptosistema (3) que aplique, sobre el documento electrónico descifrado, el mismo algoritmo de digestión (25) SHA256 que se empleo para obtener el primer hash (23), obteniendo el hash (24).
   b) Se compara el primera hash (23) obtenido del original, con el hash (24) obtenido del documento descifrado (33). Si el resultado es que son idénticos, se considera que el proceso de cifrado y descifrado ha sido correcto. Si por el contrario los hash son distintos, el documento descifrado (33) no es integro.

Según una cuarta realización del sistema, además de todos los elementos indicados en la primera realización, para obtener de forma automática una clave simétrica segura, se procede a:

1. Introducir un paso que sustituye el paso 3 a) del proceso de cifrado, con el siguiente procedimiento:
    a) Se genera de forma automática y desasistida una clave simétrica (9) empleando un algoritmo de generación de número aleatorios (20) capaz de generar números de un tamaño igual o superior a los 128 bits. Para realizar este proceso emplea la capacidad criptográfica del token criptográfico hardware (5):
        a. El programa de usuario (4) mediante el criptosistema (3) ordena a la API PKCS#11 (16) que mediante un algoritmo aleatorio de generación de números (20) obtenga un número aleatorio de tamaño 256 bits, el cual es utilizado como clave simétrica (9).

Según una quinta realización del sistema, además de todos los elementos indicados en la primera realización, para asociar el programa de usuario (4) con los documentos que procesa, procede a:

1. Introducir un paso previo al paso 3 g) y que lo complementa con el siguiente procedimiento:
    a) Lee el identificador (29) del programa de usuario (4) que está almacenado en un fichero XML.
    b) Utilizando el criptosistema (3) saca el hash (28) del identificador (29), y lo incluye en una etiqueta [H1] del contenedor de datos (21) que es encapsulado en el documento electrónico cifrado (12)
2. Introducir un paso complementario al Paso 2. c) del descifrado del documento, y que lo complementa con el siguiente procedimiento:
    a) Lee el hash (28) del identificador (29) de la etiqueta [H1],
    b) Lee el identificador (29) del programa de usuario (4) que está almacenado en un fichero XML, y utilizando el criptosistema (3) saca el hash (31) del identificador (29)
    c) Compara el hash (28) contenido en la etiqueta, con el hash (30) obtenido de identificador (29), si son idénticos el documento está asociado a este programa de usuario.

REIVINDICACIONES

1.- Método híbrido de cifrado y descifrado de documentos electrónicos que comprende los siguientes pasos:

Paso 1: Se cifra el documento electrónico (8) utilizando una clave simétrica (9) y un algoritmo simétrico (31), dando como resultado un documento cifrado (12). Se selecciona un algoritmo de cifrado asimétrico (32) con el que realizar un cifrado asimétrico multicapa de la clave simétrica (9), empleando una clave asimétrica (7) para cifrar cada capa, que dará como resultado una clave simétrica cifrada (11).

Paso 2: Se comprueba el tamaño de la clave simétrica (9), la longitud y número de claves públicas (7) que se van a utilizar, así como el tipo de cifrado requerido (32). Cuando el producto (9) u (11) a cifrar asimétricamente tiene un tamaño igual o superior al de la clave pública (7) con la que se desea cifrar, ese producto se divide en bloques enteros de bytes (14).

Paso 3: Si el número de claves públicas (7) a emplear lo requiere, los bloques enteros de bytes (14) cifrados, pueden ser subdivididos en otros bloques enteros de bytes (14) para que el tamaño siempre sea menor que la clave pública (7) a utilizar, este proceso se puede replicar en tantas ocasiones como sea necesario.

Paso 4: Para el cifrado de la clave simétrica (9) y / o los bloques enteros de bytes (14) en que ha sido dividida, se emplea un algoritmo asimétrico (32) y los bytes de una o varias claves públicas (7) que están almacenadas en un fichero PKCS#10 (2), o en un certificado electrónico X509v3.

Paso 5: Se crea un contenedor de datos (21) que almacena toda la información criptográfica necesaria para revertir el proceso de cifrado, este contenedor (21) estructura la información mediante etiquetas (13), además estas etiquetas sirven como separadores de información, por lo que se puede incluir etiquetas con ese único y exclusivo fin. No existe límite ni restricción alguna en cuanto al tipo de etiqueta, formato, extensión o cantidad.

Paso 6: Se edita el documento electrónico cifrado (12), y se encapsula en su interior el contenedor de datos, obteniendo un documento electrónico cifrado encapsulado (15)

Paso 7: Para el descifrado del documento electrónico cifrado encapsulado (15) se extrae el contenedor de datos (21) obteniendo el documento electrónico cifrado (12).

Paso 8: Del contenedor de datos (21) se obtienen los identificadores (19) de las claves públicas empleadas, los bloques (14) o la clave simétrica cifrada (11), así como toda la información necesaria para realizar las operaciones criptográficas de descifrado.

Paso 9: Utilizando el identificador (19) de la clave pública empleada para cifrar se accede a la clave privada (6), la cual puede estar contenida en un token de hardware criptográfico (5), o en un contenedor de información (22) que incluye el identificador(19) de clave, con la clave privada (6) se procede al descifrado de los bloques (14) o de la clave simétrica cifrada (11), utilizando para el descifrado el mismo algoritmo asimétrico (32) que se utilizó para cifrarla.

Paso 10: Si la clave simétrica (9) fue dividida en bloques enteros de bytes (14), los bloques descifrados son unidos para obtener la clave simétrica (9).

Paso 11: Utilizando la clave simétrica descifrada (9), se procede al descifrado del documento cifrado (12) utilizando el mismo algoritmo simétrico (31) que se utilizó para cifrarla. Obteniendo el documento electrónico descifrado (33)

2.- Procedimiento según Reivindicación 1, caracterizado porque en el Paso 4 comprende además:

Editar el fichero PKCS#10 (2) y extraer de él, como mínimo, los bytes de la clave pública (7), y su identificador (19). Almacenar esa información en un archivo de información estructurada (22), y utilizar ese archivo (22) para leer los bytes de la clave pública (7) y su identificador (19) a fin de realizar los procesos de cifrado.

3.- Procedimiento según Reivindicación 1, caracterizado por realizar un paso previo (a) al Paso 1, por realizar un paso previo (b) al Paso 2, y un paso complementario (c) al Paso 11, que comprende:

a) Se obtiene un hash (23) del documento electrónico original (8) utilizando un algoritmo de resumen (25).

b) Se descifra el documento cifrado (12) con la clave simétrica (9) y el algoritmo simétrico (31) empleado para cifrarlo, se aplica sobre el documento descifrado (33) el mismo algoritmo de resumen (25) empleado en el paso a), obteniendo un hash (24) que se compara con el primer hash (23) a), si ambos hash son idénticos el documento electrónico descifrado (33) es integro respecto al documento original (8).

c) Se aplica sobre el documento descifrado (33) el mismo algoritmo de resumen (25) empleado en el paso a), obteniendo un hash (24) que se compara con el primer hash (23) a), si ambos hash son idénticos el documento electrónico descifrado (33) es integro respecto al documento original (8).

4.- Procedimiento según Reivindicación 1, caracterizado porque dispone de un contenedor de datos (34) que almacena equivalencias de todos los caracteres reconocidos de forma estándar en Base 64 y, equivalencias que no son reconocidas de forma estándar por Base 64, y la lógica de equivalencias registradas en ese contenedor (34) se emplea para realizar un paso posterior (a) al Paso 6, y realizar un paso previo (b) al Paso 7, que comprende:

a) Se realiza una comprobación del documento electrónico cifrado y encapsulado (15), a fin de determinar si alguno de sus caracteres coincide con alguno de los contenidos en el contenedor (34), si coincide se procede a la codificación del documento (15) empleando la lógica de las equivalencias almacenadas en el contenedor (34), si algún carácter no coindice, no se realiza la codificación.

b) Si ha sido posible realizar la codificación del documento, se realiza la decodificación empleando la lógica de equivalencias del contenedor (34)

5.- Procedimiento según Reivindicación 1, caracterizado por realizar un paso previo al

Paso 1, que comprende:

La generación automática de una clave simétrica (9) empleando un algoritmo de generación de números aleatorios (20) sin limitación de tamaño.

6.- Procedimiento según Reivindicación 1, caracterizado porque el programa de usuario (4) dispone de un identificador unívoco (29), y con ese identificador (29) se realiza un paso previo (a) al Paso 5, y un paso previo (b) al Paso 8 que comprende:

a) Incluir el identificador (29) en el contenedor de datos (21) que se encapsula en el documento electrónico cifrado (12). También es posible aplicar un algoritmo de resumen (25) al identificador (29) y almacenar el hash (28). Esta información indicará si los datos almacenados son el identificador (29) o el hash (28) de ese identificador (29).

b) Se comprueba el identificador (29) del programa de usuario (4) que ha iniciado el proceso de descifrado, y se compara con el identificador (29) almacenado en el contenedor de datos (21), si no son idénticos el programa de usuario (4) no está asociado al documento cifrado (12). Si el identificador (29) almacenado en el contenedor de datos (21) es un hash (28), se aplica un algoritmo resumen (25) sobre el identificador (29) del programa de usuario (4) obteniendo un hash (30), ambos hash se comparan y si son iguales, el programa de usuario (4) está asociado al documento electrónico cifrado (12).

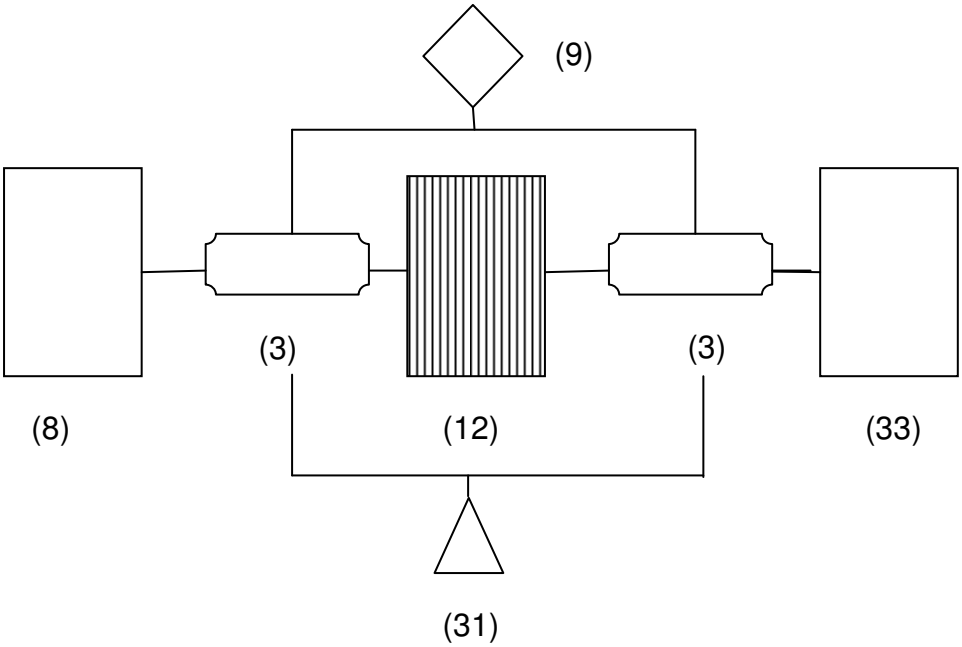

Figura 1

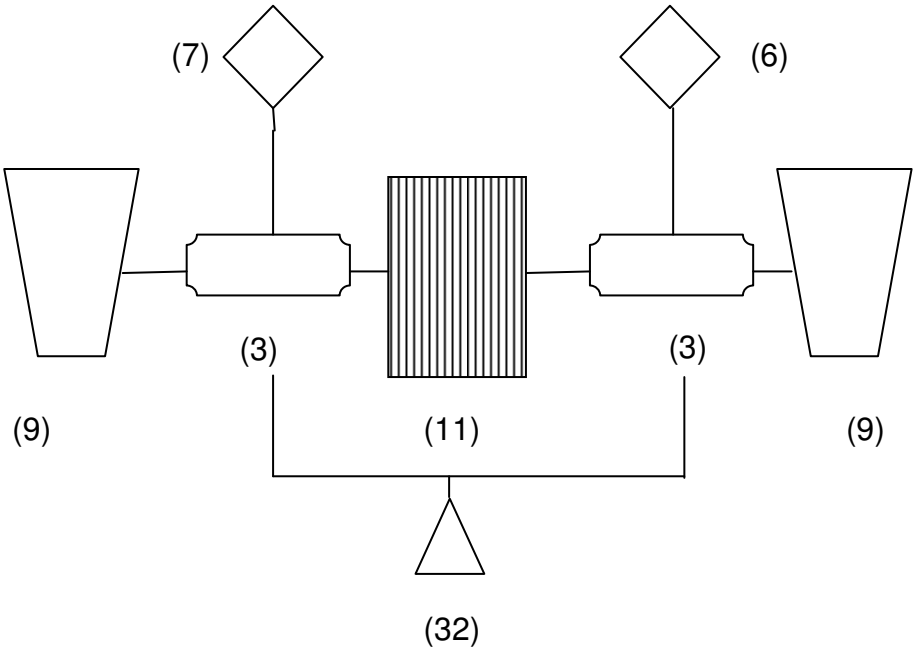

Figura 2

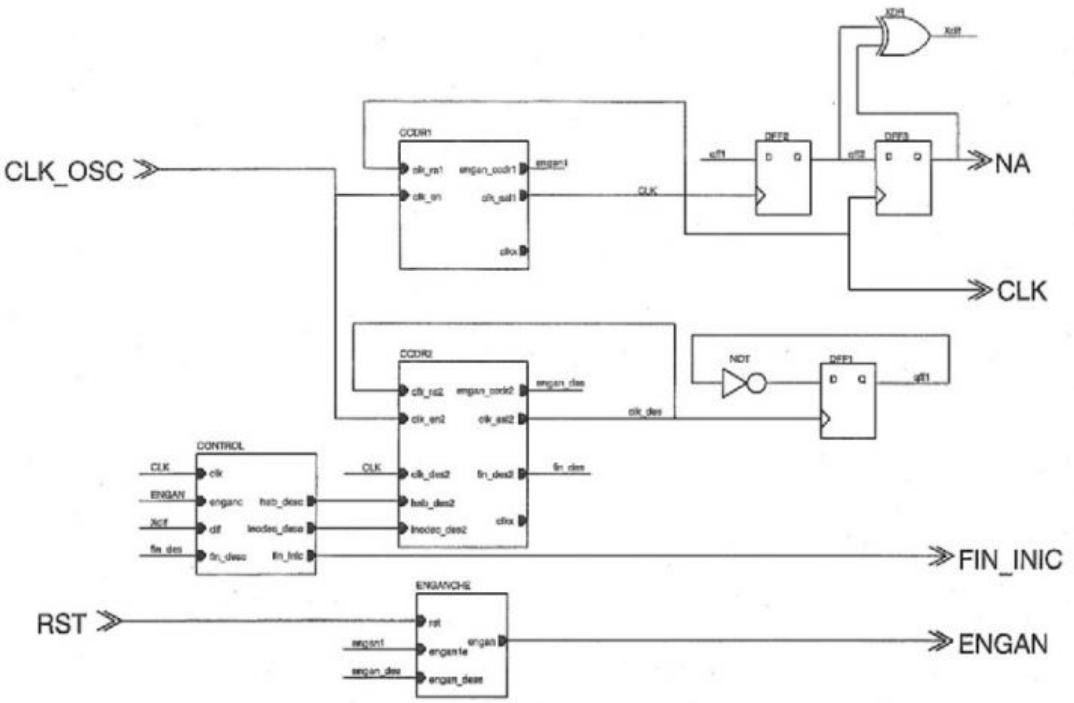

Figura 3

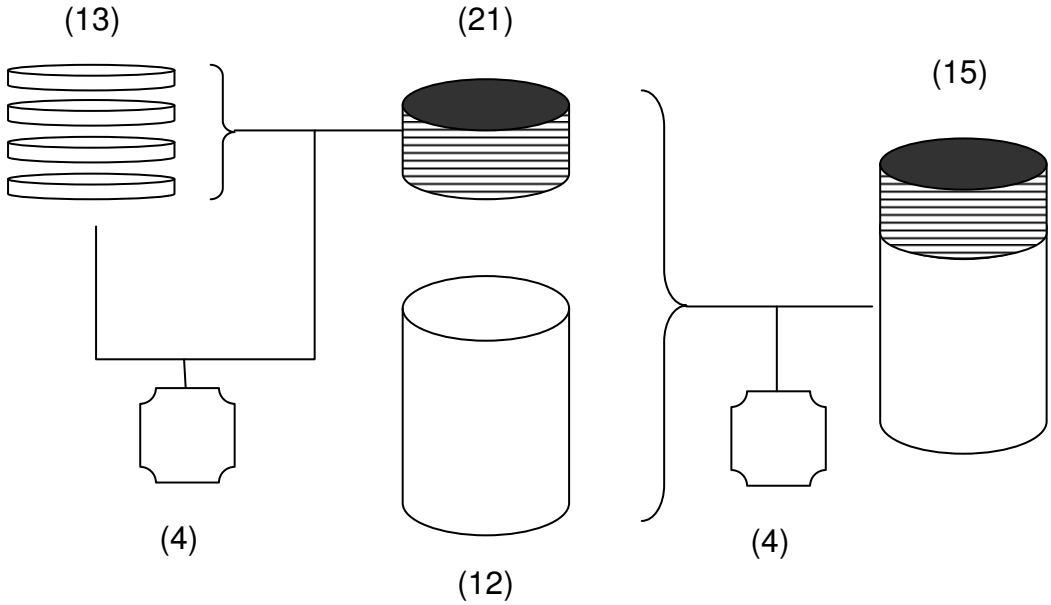

Figura 4

OFICINA ESPAÑOLA
DE PATENTES Y MARCAS

ESPAÑA

(21) N.º solicitud: 201630804

(22) Fecha de presentación de la solicitud: 13.06.2016

(32) Fecha de prioridad:

INFORME SOBRE EL ESTADO DE LA TECNICA

(51) **Int. Cl. :** ***H04L9/14*** (2006.01)

DOCUMENTOS RELEVANTES

Categoría	**(56) Documentos citados**	**Reivindicaciones afectadas**
A	US 2012179909 A1 (SAGI SURYA R et al.) 12/07/2012, Resumen	1-6
A	US 2009225988 A1 (YAMAZAKI MASAHITO) 10/09/2009, Reivindicación 4.	1-6
A	EP 0792041 A2 (IBM) 27/08/1997, Descripción: col. 2, líneas 3-20.	1-6
A	US 2008270807 A1 (FORLENZA RANDOLPH MICHAEL et al.) 30/10/2008, Todo el documento	1-6
A	US 2009097657 A1 (SCHEIDT EDWARD M et al.) 16/04/2009, Todo el documento	1-6
A	WO 2012021839 A2 (ORSINI RICK L et al.) 16/02/2012, Todo el documento.	1-6

Categoría de los documentos citados
X: de particular relevancia
Y: de particular relevancia combinado con otro/s de la misma categoría
A: refleja el estado de la técnica
O: referido a divulgación no escrita
P: publicado entre la fecha de prioridad y la de presentación de la solicitud
E: documento anterior, pero publicado después de la fecha de presentación de la solicitud

El presente informe ha sido realizado

☒ para todas las reivindicaciones ☐ para las reivindicaciones nº:

Fecha de realización del informe	**Examinador**	**Página**
18.05.2017	M. Muñoz Sanchez	1/4

Documentación mínima buscada (sistema de clasificación seguido de los símbolos de clasificación)

H04L

Bases de datos electrónicas consultadas durante la búsqueda (nombre de la base de datos y, si es posible, términos de búsqueda utilizados)

INVENES, EPODOC, WPI, NPL, XPIEE, XPI3E

Fecha de Realización de la Opinión Escrita: 18.05.2017

Declaración

Novedad (Art. 6.1 LP 11/1986)	Reivindicaciones	1-6	**SI**
	Reivindicaciones		**NO**
Actividad inventiva (Art. 8.1 LP11/1986)	Reivindicaciones	1-6	**SI**
	Reivindicaciones		**NO**

Se considera que la solicitud cumple con el requisito de aplicación industrial. Este requisito fue evaluado durante la fase de examen formal y técnico de la solicitud (Artículo 31.2 Ley 11/1986).

Base de la Opinión.-

La presente opinión se ha realizado sobre la base de la solicitud de patente tal y como se publica.

1. Documentos considerados.-

A continuación se relacionan los documentos pertenecientes al estado de la técnica tomados en consideración para la realización de esta opinión.

Documento	Número Publicación o Identificación	Fecha Publicación
D01	US 2012179909 A1 (SAGI SURYA R et al.)	12.07.2012
D02	US 2009225988 A1 (YAMAZAKI MASAHITO)	10.09.2009
D03	EP 0792041 A2 (IBM)	27.08.1997
D04	US 2008270807 A1 (FORLENZA RANDOLPH MICHAEL et al.)	30.10.2008
D05	US 2009097657 A1 (SCHEIDT EDWARD M et al.)	16.04.2009
D06	WO 2012021839 A2 (ORSINI RICK L et al.)	16.02.2012

2. Declaración motivada según los artículos 29.6 y 29.7 del Reglamento de ejecución de la Ley 11/1986, de 20 de marzo, de Patentes sobre la novedad y la actividad inventiva; citas y explicaciones en apoyo de esta declaración

Se considera D01 el documento más próximo del estado de la técnica al objeto de la solicitud.

Reivindicaciones independientes

Reivindicación 1: El documento D01 describe un sistema para proteger documentos electrónicos en el que dichos documentos se cifran con una clave simétrica y en el que dicha clave simétrica a su vez se cifra con una clave pública de un par de un criptosistema. El documento D01 no hace referencia a cómo se trataría el uso de claves simétricas de gran longitud ni a cómo se integrarían los resultados de cifrado en el documento (resumen Epodoc), ni tampoco al cifrado sucesivo de los bloques de clave simétrica, que se vayan construyendo, con distintas claves públicas, características que darían solución al problema técnico objetivo de cómo compatibilizar el aumento de la seguridad en documentos electrónicos utilizando claves de cifrado simétricas de mayor longitud y la posibilidad de que los productos de cifrado (de la clave simétrica con una clave pública) se puedan recifrar más veces con otras claves públicas.

El documento D02 divulga el encadenamiento (o cifrados sucesivos) con distintas claves públicas de una clave común (reivindicación 4) pero no incluye la división de la misma o la previsión de que se divida.

Por su parte el documento D03 sí reconoce el problema asociado a que la clave simétrica sea demasiado larga para le cifrado con clave pública y sugiere la división de dicha clave simétrica en bloques (columna 2, líneas 3-20). Sin embargo, el documento D03 no menciona cómo hacer esta división, el número de bloques etc.

En consecuencia el experto en la materia, a la luz de los documentos recuperados del estado de la técnica, no se vería orientado a solucionar el problema técnico objetivo expuesto como se indica en esta reivindicación. Por tanto, la reivindicación tiene novedad según el art. 6.1 y actividad inventiva según el art. 8.1 de la Ley 11/86 de Patentes.

Reivindicaciones dependientes

Reivindicaciones 2-6: estas reivindicaciones tienen novedad y actividad inventiva por depender de la reivindicación 1 que también la posee según los arts. 6.1 y 8.1 de la Ley 11/86 de Patentes.