(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 359 205**

(51) Int. Cl.:
H04L 9/32 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Número de solicitud europea: **98965522 .0**
(96) Fecha de presentación : **22.12.1998**
(97) Número de publicación de la solicitud: **1048143**
(97) Fecha de publicación de la solicitud: **02.11.2000**

(54) Título: **Procedimiento y aparato para el almacenamiento y uso seguros de claves criptográficas.**

(30) Prioridad: **23.12.1997 US 996758**

(45) Fecha de publicación de la mención BOPI:
19.05.2011

(45) Fecha de la publicación del folleto de la patente:
19.05.2011

(73) Titular/es: **ARCOT SYSTEMS, Inc.**
Suite 200, 3200 Patrick Henry Drive
Santa Clara, California 95054, US

(72) Inventor/es: **Kausik, Balas, Natarajan**

(74) Agente: **Ponti Sales, Adelaida**

DESCRIPCIÓN

Procedimiento y aparato para el almacenamiento y uso seguros de claves criptográficas

CAMPO TÉCNICO

[0001] La invención se refiere en general a asegurar criptográficamente un artículo de datos de control de acceso y, más específicamente, a asegurar el almacenamiento y el uso de claves criptográficas.

ANTECEDENTES DE LA INVENCIÓN

[0002] Las técnicas de seguridad de datos criptográficos asegurar datos mediante la encriptación de los datos usando una clave. Los datos desencriptados sólo se pueden recuperar con la clave. La clave se selecciona para ser lo suficientemente larga que un intruso malicioso no pueda adivinar la clave mediante ensayo y error exhaustivo, incluso con el uso de cantidades substancialmente grandes de recursos informáticos. Por lo tanto, la seguridad de los datos se ha transferido a la seguridad de la clave. A menudo, es deseable almacenar la clave para que el acceso a la clave esté controlado por una frase de paso o PIN (Número de Identificación Personal) que es lo suficientemente corta para que un usuario humano la recuerde fácilmente. Esto permitirá convenientemente al usuario humano utilizar su PIN para recuperar la clave, y luego usar la clave para recuperar los datos encriptados. Desafortunadamente, si el PIN es lo suficientemente corto como para que el ser humano lo recuerde, también es lo suficientemente corto como para que un intruso malicioso lo adivine por ensayo y error exhaustivo, lo que socava la seguridad de la clave, y por lo tanto la seguridad de los datos encriptados. Este ha sido un molesto problema en la seguridad de datos.

[0003] En procedimientos criptográficos asimétricos tales como RSA, cada usuario posee un par emparejado de claves, una clave privada y una clave pública. La clave privada y la clave pública forman un par emparejado y único en el que los mensajes (por ejemplo, mensajes, datos, código, y cualquier otra información que puede representarse digitalmente incluyendo otras claves criptográficas o representaciones criptográficas de información) que están encriptados con la clave privada sólo puede ser desencriptados con la clave pública y viceversa. Esta correspondencia uno a uno entre la clave privada y la clave pública se puede utilizar para crear firmas digitales para mensajes electrónicos y transacciones. Para firmar un mensaje electrónico, el usuario puede simplemente encriptar el mensaje con su clave privada. A continuación, adjuntará su clave pública al mensaje encriptado y lo enviará al destinatario. Alternativamente, el usuario no adjuntará su clave pública al mensaje, pero el destinatario podrá buscar la clave pública del usuario de un directorio de claves públicas. En cualquier caso, para verificar la firma, el destinatario podría desencriptar el mensaje utilizando la clave pública adjunta, y si el desencriptado se realiza correctamente, el destinatario confía en el origen del mensaje.

[0004] Tal como se describió anteriormente, el remitente tendría que encriptar todo el mensaje con su clave privada para firmarlo, lo cual es computacionalmente costoso. Para solucionar esto, basta con computar un hash corto de longitud fija, por ejemplo 128 bits de longitud, del mensaje y luego encriptar el valor hash. Si la función hash es buena, tal como MD5, las posibilidades de que dos mensajes distintos tengan el mismo valor hash es extremadamente pequeño. Por lo tanto, los procedimientos de firma digital típicamente computan los hashes de los mensajes, y encriptan sólo el valor hash. El valor hash encriptado y la clave pública del remitente se adjuntan al mensaje original antes de la transmisión al receptor. Para verificar la firma, el receptor primero computará el hash del mensaje recibido. Si el valor hash computado es el mismo que la forma desencriptada del hash encriptado, el destinatario está seguro del origen del mensaje.

[0005] En lo anterior, la fuerza del proceso de verificación de la firma depende de la confianza del destinatario de que la clave pública adjunta al mensaje es realmente la clave pública del supuesto propietario. Cualquiera puede generar un par emparejado de las claves y puede hacerse pasar por el usuario, a menos que exista un medio para evitar este tipo de mascarada. Con este fin, las claves públicas están a menudo certificadas por notarios de terceras partes, llamados *autoridades de certificación* o CAs para abreviar. Ejemplos de autoridades de certificación son entidades comerciales tales como Verisign y Entrust. El CA une una clave pública de un certificado con la identidad del certificado, y luego firma el mensaje junto con la clave privada del CA, para formar el certificado de la clave pública del certificado. Así, un poseedor de certificado adjuntará su certificado de clave pública al mensaje encriptado antes de enviar el mensaje al destinatario. Para comprobar la identidad del remitente y la autenticidad de su clave pública, el receptor verifica la firma de la CA en el certificado de la clave pública del remitente, utilizando la clave pública de la CA. Como habrá sólo un pequeño número de entidades emisoras de certificados de mucha confianza, la llave pública de la CA sería fiable y de fácil acceso para el destinatario. Así, las firmas de clave pública se pueden utilizar para autenticación entre extraños en el caso incluso si el destinatario y el remitente no tienen ninguna relación previa, el destinatario puede verificar la firma del remitente siempre y cuando el destinatario y el remitente confíen en una CA común.

[0006] La singularidad y la incapacidad de falsificar la firma de un usuario dependen en gran medida de la

capacidad del usuario para mantener privada su clave privada. Cualquiera que tenga acceso a la clave privada de un usuario puede pasar por ese usuario en un anonimato total. Por lo tanto, el uso generalizado de la firma digital para el comercio electrónico y otras aplicaciones requerirán una tecnología para el almacenamiento seguro de claves privadas. En la actualidad, se cree que las claves privadas se almacenan mejor físicamente aislándolas en dispositivos de hardware tales como tarjetas inteligentes, tarjetas Fortezza, tarjetas PCMCIA y otros dispositivos de hardware compacto. Las tarjetas inteligentes son tarjetas de tamaño de tarjetas de crédito que contienen un microprocesador y alguna memoria. La clave privada del usuario y el certificado de la clave privada se escriben en la memoria. Para utilizar la tarjeta, el usuario sólo tiene que introducir la tarjeta en un lector de tarjetas adecuado conectado a un ordenador central, y luego introduce su PIN para activar la tarjeta. Si se introduce el PIN correcto, el procesador de la tarjeta liberará la clave privada para su uso en el equipo central. Si se introduce un PIN incorrecto, el procesador no liberará la clave privada del usuario. Algunas tarjetas inteligentes a prueba de manipulación están configuradas de modo que si se introducen PINs incorrectos en varios intentos consecutivos, la tarjeta se bloquea de forma permanente. Algunas tarjetas inteligentes sofisticadas (a menudo llamadas tarjetas criptográficas) pueden realizar operaciones de encriptado, de modo que la clave privada nunca necesita salir de la tarjeta inteligente. Los bytes a procesar entran en la tarjeta inteligente desde el equipo central, se procesan y se transmiten al ordenador central. Desafortunadamente, incluso las tarjetas criptográficas deben confiar en el equipo central para la transmisión de los bytes de ida y vuelta desde el lector de tarjetas y, por lo tanto, no son perfectamente seguras. Un ordenador central malintencionado podría simplemente sustituir un mensaje por otro antes de la transmisión, de manera que el usuario cree que está firmando un mensaje, mientras que en realidad está firmando otro. Por lo tanto, incluso las tarjetas criptográficas no pueden luchar contra ordenadores centrales maliciosos.

[0007] Aunque la tarjeta inteligente resuelve el problema del almacenamiento seguro de claves privadas, adolece de varios inconvenientes importantes:

1) *Coste inicial alto*: Las tarjetas inteligentes requieren infraestructura adicional de hardware cara en forma de lectores de tarjetas inteligentes;

2) *Gastos generales de administración*: Las tarjetas inteligentes requieren gastos generales de administración para la distribución y el mantenimiento; y

3) *Baja comodidad para el usuario*: El usuario no puede duplicar, hacer copias de seguridad o cotejar las tarjetas inteligentes, debido a sus características a prueba de manipulaciones.

[0008] Una cartera de claves basada en software seguro que no requiere hardware adicional mitigaría algunos de los inconvenientes de la tarjeta inteligente mencionada anteriormente. Desafortunadamente, no hay actualmente disponibles este tipo de soluciones. La tecnología estándar que se utiliza hoy para el almacenamiento de claves, en productos tales como los de Microsoft y Netscape, ofrece muy poca protección contra la manipulación, y se pueden romper bastante fácilmente. En concreto, estas carteras de claves almacenan la clave privada en forma encriptada, utilizando el PIN del usuario como la clave de encriptado. El PIN debe ser lo suficientemente corto como para el usuario lo recuerde, por ejemplo un código de seis dígitos. Si esta cartera de claves de software cae en manos de un hacker, el hacker podría tratar de forma exhaustiva todo el millón de códigos de seis dígitos posibles de manera automática en un ordenador personal en unos pocos minutos, hasta que encuentre el código que abre la cartera de claves. En este punto, el hacker sabe que tiene exactamente el PIN correcto, y tiene acceso a las claves privadas del usuario. Así, el principal problema con el suministro de una única cartera de claves de software son los requisitos de competencia que el PIN ser lo suficientemente corto como para el usuario lo recuerde, pero lo suficiente largo como para hacer que la cartera de claves sea resistente a una manipulación.

[0009] La patente US 5.206.905 describe un dispositivo protegido con contraseña en el que cuando se introduce una contraseña correcta, se genera el contenido de una memoria segura, pero cuando se introduce una contraseña incorrecta se utiliza como valor de inicialización para un generador de números pseudo-aleatorios para producir una salida aleatoria para cada contraseña de entrada.

[0010] El libro “Handbook of Applied Cryptography” de Menezes et al., CRC Press LLC, 1997, describe técnicas criptográficas estándar, y, en particular, describe el uso de PINs en las páginas 394-399 y 551-554.

DESCRIPCIÓN DE LA INVENCIÓN

[0011] La invención se define en las reivindicaciones independientes. Realizaciones particulares se indican en las reivindicaciones dependientes.

BREVE DESCRIPCIÓN DE LAS FIGURAS

[0012]

La figura 1 es una visión esquemática de una cartera de claves criptográficas y subsistemas de generación

de claves, de certificación de claves, y de verificación.

La figura 2 muestra una cartera de llaves convencional.

La figura 3 muestra una realización de ejemplo de una cartera de claves de acuerdo con la presente invención.

La figura 4 muestra una distribución de PINs pseudo-válidos entre el espacio de todos los PINs posibles.

La figura 5 ilustra un barrio alrededor de un PIN correcto.

La figura 6 ilustra un ataque conocido de texto que está dirigido mediante un aspecto de firma de la presente invención.

La figura 7 ilustra una realización de ejemplo de un aspecto de firma de la presente invención.

La figura 8 ilustra una realización de ejemplo para la generación de claves privadas bien formadas en un aspecto de generación de claves de la presente invención.

La figura 9 ilustra un sistema de ejemplo para firmar digitalmente un mensaje, que incluye una cartera de claves y lógica de generación de claves asociada.

La figura 10 ilustra un ataque conocido de clave pública que está dirigido mediante un aspecto de certificado pseudo-público de la presente invención.

La figura 11 ilustra un certificado convencional y pseudo-público de ejemplo.

La figura 12 ilustra un aspecto de servidor de certificados de ejemplo de la presente invención.

DESCRIPCIÓN DETALLADA DE LA INVENCIÓN

[0013] Algunas de las realizaciones descritas a continuación no son realizaciones de la invención según las reivindicaciones.

[0014] La figura 1 proporciona una visión general esquemática de los elementos funcionales del sistema en su conjunto, cada uno de los cuales juega un papel importante en el almacenamiento seguro de la clave privada y su uso posterior. El primer componente es el componente de generación de claves **100**, que crea las claves pública y privada para el usuario. El segundo componente es la cartera de claves **110**, que almacena la clave privada y se utiliza para crear firmas. El tercer componente es el componente de verificación **120**, que se utiliza para verificar las firmas creadas por la cartera de claves. El cuarto componente es el componente de certificación **130**, que se utiliza para certificar la clave pública creada por el componente de generación de claves. El componente de la cartera de claves proporciona la integración del camuflaje criptográfico, mientras que los otros elementos garantizan que la integración es lo suficientemente variada para ser de conveniencia para el usuario legítimo, y sin embargo, parece suficientemente homogénea para frustrar a los intrusos maliciosos. En un ejemplo de realización de la invención, lo anterior se implementa como software que se ejecuta en un ordenador de propósito general.

1. Detalles de la cartera de claves

[0015] A los efectos de análisis, consideramos que la cartera de claves es una caja de seguridad basada en software que contiene la clave privada del usuario. Suponemos también que la caja de seguridad sólo puede ser desbloqueada mediante un PIN secreto que sólo conoce el usuario legítimo. Suponemos que la cartera de claves cae en manos de un hacker malicioso. Se enumeran los tipos de ataques que el hacker puede montar en la caja negra, y proporciona unos medios para resistir cada ataque. En aras de la claridad, la descripción se establecerá respecto al sistema de firma de clave pública RSA. Sin embargo, los expertos en la materia apreciarán que los elementos básicos de la descripción son aplicables a otros sistemas, también incluyendo, sin limitación, los sistemas de firma El-Gamal y DSS, y criptosistemas de curva elíptica.

a. Ataque Hash PIN

[0016] Una cartera de claves convencional se representa esquemáticamente en la figura 2. Un PIN **200** (de manera más general, un código de acceso) introducido para desbloquear la cartera pasa a través de una función hash uno a uno **210.** La función hash también puede incluir un valor de sal u otra característica que mejora la seguridad, tal como se apreciará por parte de personas expertas en la materia. El valor hash **215** del PIN introducido se compara con un valor hash almacenado **220**, que es el valor hash del PIN correcto. Si los dos valores hash

coinciden, el PIN pasa al módulo de desencriptado **240**. La clave privada que ha sido encriptada (con el PIN correcto como clave de encriptado) y almacenada en el campo **230**, es desencriptada mediante el módulo de desencriptación **240**, que es típicamente DES o alguna otra función de encriptado, tal como, por ejemplo, el triple-DES, IDEA o BLOWFISH. Así, la clave privada desencriptada **250** es liberada para su uso.

[0017] Las operaciones criptográficas para calcular el(los) hash(es) y desencriptar el hash almacenado pueden implementarse usando uno o más módulos lógicos de encriptado (por ejemplo, software), y el valor hash correcto y la clave privada pueden almacenarse en los campos de datos protegidos u otras formas de memoria (por ejemplo, lectura de ROM, de lectura de medios informáticos, etc.). Una cartera de claves típica incluiría también la entrada y la salida lógica para la recepción de PINs candidatos y la salida para desencriptar claves privadas, así como la lógica para la gestión, visualización, copia, y manejo de claves y otros datos.

[0018] La naturaleza uno a uno de la función hash asegura que el PIN correcto y sólo el PIN correcto abrirá la cartera de claves. Por desgracia, también proporciona la información completa a hackers maliciosos para automatizar el proceso de adivinar el PIN correcto. En una implementación típica, el PIN es un código de seis dígitos o menos. El hacker simplemente tiene que encontrar el código de seis dígitos que hashea al valor hash almacenado. Si obtiene una copia de la cartera de claves, puede llevar a cabo este ataque a su equipo, totalmente desapercibido y de forma automatizada, en cuestión de unos minutos. Por ejemplo, podría escribir un programa que simplemente compruebe todos los códigos PIN de seis dígitos en la cartera de claves.

[0019] Para resistir al ataque hash de PINs, la presente invención sustituye el hash uno a uno con un hash de muchos a uno, es decir, un hash en el que muchas entradas producen (por ejemplo, regeneran) la salida del mismo hash. Esto se representa en el diagrama de flujo de la figura 3. En una implementación típica, la función hash de muchos a uno **310** podría hashear códigos de seis dígitos para valores hash de dos dígitos. Al igual que en la cartera de claves convencional, el valor hash **315** del PIN introducido 300 se compara en 325 con el valor hash almacenado **320**, que es el valor hash del PIN correcto. Si los dos valores hash coinciden, la cartera de claves se abre. La clave privada se almacena encriptada de nuevo en el campo **330** de la cartera de claves, con el PIN correcto como clave de encriptado. Cuando se introduce el PIN correcto, la clave encriptada es desencriptada y la clave privada correcta almacenada **350** se libera a 335 para su uso. Sin embargo, como la función hash es de muchos a uno, habrá muchos PINs entrados diferentes que se abrirán la cartera de claves. (los PINs que generan aleatoriamente el mismo valor hash como el PIN correcto, incluido el PIN correcto, se llaman PINs pseudo-válidos). Por ejemplo, si la función hash genera aleatoriamente códigos de seis dígitos para los valores hash de dos dígitos, habrá 10.000 PINs pseudo-válidos de seis dígitos que abrirán la cartera de claves, de un total de 1.000.000 de códigos de seis dígitos posibles. Los PINs pseudo-válidos pasarán todos al módulo de desencriptado **340** para desencriptar la clave encriptada almacenada para producir una clave privada candidato. Sin embargo, todas menos una de estas claves privadas candidatas serán desencriptaciones incorrectas de la clave privada almacenada (correcta). Sólo cuando el PIN introducido es el PIN correcto se recuperará la clave privada correcta.

[0020] Preferiblemente, la función hash de muchos anterior se debe elegir para ser un buen hash. Por ejemplo, y sin limitación, MD5 y SHA son buenas funciones hash conocidas. Las buenas funciones hash son un medio para distribuir de manera substancialmente uniforme el pseudo-PIN válido en el espacio de todos los PINs posibles. Por ejemplo, consideremos una función hash a partir de códigos de seis dígitos a valores hash de dos dígitos. De los 1.000.000 de posibles valores de entrada, 10.000 serán PINs pseudo-válidos. Si la función hash es un buen hash , estos valores estarán distribuidos de manera substancialmente uniforme. En particular, uno de cada cien PIN será pseudo-válido, y estos estarán distribuidos aleatoriamente de manera efectiva. Específicamente, las probabilidades son de 1/100 que si el usuario comete un error tipográfico en introducir el PIN correcto, entonces el PIN resultante será un pseudo-PIN válido. Gráficamente, esto se ve en la figura 4, donde el espacio de todos los posibles PINs se muestra como una pared **400**. Los orificios **410** en la pared corresponden a los PINs pseudo-válidos. Sólo uno de estos orificios **420** corresponde al PIN correcto, tal como se muestra en la figura. Observe que hay una zona de PINs alrededor de cada PIN pseudo-válido que no hashea con el valor hash almacenado. Compare esto con la figura 5, que muestra el espacio de los PINs para un hash uno a uno que se utiliza en la cartera de claves convencional. Tenga en cuenta que la figura 5 muestra un solo orificio **510**, correspondiente al PIN correcto. Observe también que la zona local del PIN correcto en la figura 4 se ve como la zona del PIN correcto de la figura 5. En este sentido, la experiencia del usuario legítimo con la cartera de claves de la presente invención es muy similar a su experiencia con la cartera de claves convencional.

[0021] Otro posible escenario consiste en utilizar un hash débil, es decir, uno que de lugar a la agrupación de los PINs pseudo-válidos, por el que un intruso que adivina un PIN pseudo-válido será más fácil de encontrar a otras. Un usuario legítimo haciendo una serie de errores tipográficos de un dígito también podría obtener una secuencia de PINs pseudo-válidos y, si el sistema acepta la clave privada o mensajes encriptados tiene así una característica de alarma o desactivación bajo fallos repetidos, esto bloqueará inadvertidamente al usuario legítimo. Así, un hash débil está típicamente desfavorecido por el hash bueno. Sin embargo, puede haber algunas aplicaciones donde un hash débil proporciona ciertas características tales como la eficiencia computacional y la facilidad de aplicación que son ventajosas para aplicaciones especializadas.

b. Ataque de firma conocida

[0022] Otro ataque común es el ataque de firma conocida. En este ataque, a veces llamado ataque de texto plano conocido, el pirata informático malintencionado tiene acceso a dos tipos de información: (a) la cartera de claves del usuario, y (b) un mensaje (tanto en texto sin formato y en forma firmada) que se ha firmado previamente por parte del usuario. Este ataque se muestra gráficamente en la figura 6. El hacker intentará todos los PINs posibles **600** en la cartera de claves, y para cada PIN pseudo-válido, utiliza la clave privada de desencriptado**610** para firmar el texto sin formato conocido **620**, creando una firma **630**. Si la firma **630** coincide con la firma conocida del usuario **640** del mismo mensaje de texto sin formato, el hacker sabe que ha descubierto el PIN correcto y que ha recuperado la clave privada del usuario desencriptada correctamente. En el proceso de firma convencional, el mensaje de texto sin formato que se debe firmar con hash utilizando un algoritmo hash (tal como MD5), y el texto sin formato hash se encripta con la clave privada del usuario para formar la firma. A menudo, una plataforma pseudo-aleatoria se agrega al texto sin formato antes de hashear para resistir un ataque de texto sin formato elegido. Estos bits pseudo-aleatorios se generan habitualmente a partir de una semilla que se guarda en la cartera de claves, o alguna otra fuente que se puede seguir y replicar, tal como la hora del día, etc. Una desventaja de estos bits pseudo-aleatorios es que un atacante que determina el mecanismo de generación de aleatoriedad puede obtener información útil para el ataque de firma conocida. Así, un aspecto de la presente invención se resiste a este ataque a través de una variación en el proceso de firma.

[0023] Tal como se muestra en la figura 7, el componente de firma de la presente invención rellena el texto sin formato hash **720** con bits *muy aleatorios* **710**, antes del encriptado con la clave privada **730**, para crear una firma no reproducible **740**. Estos bits muy aleatorios pueden haber sido obtenidos mediante un procedimiento que se basa en una fuente de aleatoriedad fuera de la cartera de claves. Ejemplos de estos son fuentes físicas de aleatoriedad, tales como la variación en el tiempo de búsqueda de la unidad de disco en un ordenador central, los intervalos de tiempo aleatorio entre las pulsaciones de teclas en un teclado, o la entrada de caracteres aleatorios por parte de un usuario. Estos y otros procedimientos para generar una aleatoriedad fuerte son bien conocidos por los expertos en la materia (por ejemplo, véase D. Davis, R. Ihaka , y P. Fenstermacher, “Cryptographic Randomness from Air Turbulence in Disk Drives”, Advances in Cryptology: Proc. Crypto 84, Springer-Verlag, 1985, páginas 183-215, o, más generalmente, Bruce Schneier, Applied Cryptografy , 2ª Ed. Wiley, 1996). El propósito de estos rellenos muy aleatorios es asegurar que las firmas no se pueden replicar por parte de un hacker malicioso, ya que no conoce el relleno aleatorio, y no puede volver a crear el relleno aleatorio a partir de cualquier información almacenada en la cartera de claves, tal como puede ser el caso con un relleno pseudo-aleatorio. Sin embargo, otras aplicaciones de aleatoriedad fuerte para disuadir ataques son bien conocidas por los expertos en la materia, y pueden implementarse en realizaciones alternativas de la presente invención.

c. Ataque de claves formadas III

[0024] Otro ataque es aquel en el que el hacker malicioso intenta todos los PINs posibles y, para cada PIN pseudo-válido, examina la clave desencriptada. Si la clave no está bien formada, el hacker sabe que el PIN pseudo-válido no puede ser el PIN correcto. Por lo tanto, es necesario garantizar que las claves privadas candidatas, derivadas mediante el desencriptado de la clave almacenada con PINs pseudo-válidos, también están bien formadas.

[0025] En el sistema RSA, una clave privada tiene un exponente (*d*) y un módulo (*n*), y se dice que está bien formada si el módulo no tiene ningún factor pequeño y el exponente *d* es menor que *(p-1)(q-1)* y no divisible por *p* o *q,* donde *p* y *q* son los factores primos del módulo *n*. Por lo tanto, el módulo y exponente de las claves privadas candidatas deben cumplir estas condiciones. Una realización de la presente invención que asegura ambas condiciones se muestra en la figura 8. Suponiendo que la clave privada correcta se formó correctamente, el módulo **810** se almacena sin encriptar y no se modifica mediante el proceso de encriptado y desencriptado. Por lo tanto, el módulo de la clave privada candidata está bien formado por definición. El problema, entonces, es asegurar que el exponente de la clave privada candidata (a partir de ahora, el “exponente candidato”) está bien formado. La probabilidad de que el exponente candidato comparta factores primos con el módulo es muy pequeña, y comparable con la probabilidad de factor del módulo por casualidad. Por lo tanto, la principal limitación es el tamaño del exponente candidato en relación con el módulo. Una forma de asegurar esto es la siguiente. Como el exponente de la clave privada correcta (a partir de ahora, el “exponente correcto”) fue bien formado, los exponentes candidatos que son similares en tamaño con el exponente correcto es probable que también estén bien formados.

[0026] Un procedimiento para asegurar esto es dividir el exponente correcto en su porción más significativa **820** y en su porción menos significativa **830**. Por ejemplo, 65537 tiene “65” como sus 2 dígitos más significativos y “537” como 3 dígitos menos significativos. Los bits más significativos se almacenan sin encriptar, mientras que sólo los bits menos significativos del exponente correcto son encriptados usando el PIN y se almacenan. Cuando se desencriptan menos bits significativos almacenados con una PIN pseudo-válido, cambiarán completamente (por ejemplo, 537 podría llegar a ser 142 en el ejemplo anterior). La porción almacenada más significativa y la forma

desencriptada la porción menos significativa se juntan para recuperar el exponente candidato **840**. Sin embargo, la magnitud del exponente candidato vuelto a montar no habrá cambiado significativamente. Al elegir adecuadamente el número de bits menos significativos, se puede controlar el orden de magnitud del exponente candidato vuelto a calcular, para asegurarse de que sigue siendo menor que el módulo. Lo anterior ilustra el concepto de encriptado de bits menos significativos usando aritmética de base 10. La implementación basada en ordenador correspondiente sería similar, excepto usando bits en lugar de dígitos. Por ejemplo, si el módulo tiene 512 o más bits, una implementación de ejemplo podría encriptar sólo los 128 bits menos significativos del exponente usando el PIN como la clave.

[0027] Los expertos en la materia se darán cuenta de que hay muchas formas alternativas de garantizar que las claves privadas candidatas están bien formadas. En un procedimiento alternativo, el módulo de generación de claves selecciona dos números aleatorios *k* y *m,* donde *m* es un número entre *d* y el módulo *n.* En una implementación de ejemplo, *k* podría tener una longitud de 64 bits. La suma *d* + *km* se calcula, *k* se descarta, y *m* se almacena para su uso posterior. Más que almacenar el exponente correcto *d,* la suma *d* + *km* se encripta entonces utilizando el PIN, y se almacena como una suma encriptada. Cuando se introduce un PIN pseudo-válido, la suma encriptada se desencripta para obtener la suma desencriptada, que después se evalúa el módulo *m*. Es decir, un exponente candidato se recupera como el resto después de dividir la suma desencriptada *d* + *km* por *m.* Este exponente candidato es, por construcción, más pequeño que *m*. Como *m* fue seleccionado para ser más pequeño que el módulo *n,* el exponente candidato está, por lo tanto, también garantizado para que sea menor que *n.*

[0028] Lo anterior ilustra dos realizaciones de ejemplo para garantizar la buena formación de claves privadas candidatas compatibles con RSA. Como los expertos en la materia apreciarán, el concepto de garantizar la buena formación también se extiende a otras claves privadas y, más generalmente, a otros tipos datos almacenados con acceso controlado. Por ejemplo y sin limitación, si el elemento de datos almacenados fuera una combinación de un seguro físico, los datos del elemento candidato tendría que ser en el formato adecuado para el marcado de combinación. Cualquier elemento de datos de control de acceso que tiene un formato esperado se puede almacenar usando este aspecto de la presente invención, en la cual se garantiza una buena formación durante el desencriptado de los códigos de acceso candidatos.

d. Ataques combinados

[0029] Para resistir simultáneamente al ataque hash PIN, al ataque de firma conocida y a los ataques de clave de forma poco meditada, los diversos aspectos de la presente invención, tal como se muestran en la figura 3, la figura 7 y la figura 8 se pueden combinar tal como se muestra en el conjunto de la figura 9. Los expertos en la materia reconocerán que cualquier combinación, subconjunto o superconjunto de los ataques pueden resistirse mediante la combinación (o modificación) de los aspectos apropiados de la presente invención, para su uso en entornos en los que la combinación, subconjunto o superconjunto particular de los ataques es una preocupación.

2. Detalles del componente de certificación

[0030] El componente de certificación de la presente invención crea certificados de clave pública que son algo diferentes de los certificados de clave pública convencionales. Esencialmente, las claves públicas tal como se usan aquí no son verdaderamente públicas como con los procedimientos convencionales, sino que están pensadas para una distribución limitada (por ejemplo, dentro de las organizaciones, a través de intranets o de otra manera, dentro de empresas cerradas o pseudo-públicas). Esta desviación de los procedimientos convencionales se utiliza para resistir el ataque siguiente a la clave privada.

a. Ataque de clave pública conocida

[0031] En este ataque, el hacker malicioso tiene acceso a dos tipos de información: (a) la cartera de claves del usuario, y (b) la clave pública del usuario, tal como podría estar fácilmente disponible en un directorio de certificados de clave pública. El ataque se muestra gráficamente en la figura 10. El hacker intentará todos los PINs **1000** posibles en la cartera de claves, y para cada PIN pseudo-válido, usaría la clave privada desencriptada **1010** para encriptar un mensaje de muestra elegido arbitrariamente **1020**, y luego desencriptaría el mensaje encriptado con la clave pública del usuario. Si el mensaje desencriptado **1040** coincide con el mensaje de texto sin formato de la muestra, el hacker sabe que ha descubierto el PIN correcto y ha recuperado la clave privada correctamente desencriptada del usuario.

[0032] Para resistir a este ataque, una realización de la invención no permite que las claves públicas sean verdaderamente públicas. Como una cuestión de conveniencia, llamaremos como estas claves públicas de distribución limitada “claves pseudo-públicas” y llamaremos a los certificados que contienen, por ejemplo, las claves pseudo-públicas “certificados pseudo-públicos”. Específicamente, los certificados pseudo-públicos contienen la clave pseudo-pública del usuario en forma encriptada. Sólo las partes autorizadas pueden acceder a una clave pseudo-pública para verificar la firma del usuario. Esto está en fuerte contraste con el uso convencional de los certificados de clave pública, donde cualquier persona puede verificar una firma de clave pública. Por supuesto, la cartera de claves

y otros aspectos o formas de realización de la presente invención pueden utilizarse con certificados convencionales solamente, pero se proporciona una seguridad incluso mayor si se utilizan también claves y certificaciones pseudo-públicas, tal como se describe aquí. Los expertos en la materia apreciarán fácilmente que los actuales dispositivos y procedimientos de emisión de certificaciones pueden adaptarse fácilmente para dar cabida a la realización anterior de la presente invención. Por lo tanto, las implementaciones de hardware y/o de software específicas de esta realización de un componente de certificación no necesitan describirse en detalle. Por el contrario, sólo las diferencias de los certificados convencionales se describen a continuación. Los lectores expertos en la materia reconocerán que los certificados convencionales vienen en varios formatos, el más notable de los cuales es el formato X.509 y sus revisiones; sin embargo, los elementos esenciales de todos los formatos convencionales son similares, si se consideran en relación con la presente invención.

[0033] Un certificado de clave pública convencional y una posible realización de un certificado pseudo-público se muestran uno al lado del otro en la figura 11. El certificado pseudo-público de ejemplo puede tener el mismo formato que el certificado convencional. Sin embargo, el cuerpo del certificado **1100** que contiene la clave pseudo-pública está codificado en una forma que sea legible sólo por un verificador autorizado. Por ejemplo, en una implementación, el encriptado puede ser la clave pública del verificador autorizado. Sólo los servidores de autentificación que tienen acceso a la clave privada correspondiente pueden desenvolver el certificado del usuario para acceder a la clave pública del usuario. Si hay varios verificadores autorizados, el cuerpo del certificado podría llevar varias copias encriptadas de la clave pseudo-pública, encriptándose cada copia con la clave pública de uno de los verificadores. Cada empresa o entidad que utiliza este aspecto de la presente invención tendría un servidor de certificados que tiene los componentes de certificación descritos anteriormente para apoyo de sus certificados pseudo-públicos. Las personas expertas en la materia apreciarán que la característica importante del certificado pseudo-público es que la clave pública se encripta y que se puede desencriptar sólo por parte de los verificadores autorizados, y esta característica se puede lograr de muchas maneras diferentes utilizando una variedad de algoritmos criptográficos. Por ejemplo, en una realización alternativa del certificado de clave pseudo-pública, la clave pública se encripta mediante una clave DES, y la clave DES se encriptará con la clave pública del verificador autorizado.

[0034] El certificado resultante sería firmado entonces por la autoridad de certificación similar al de un certificado convencional. Es la naturaleza pseudo-pública de las claves públicas en la presente invención la que proporciona dos importantes ventajas en la gestión de claves. En primer lugar, como la autoridad de certificación es explícitamente consciente de que está autorizada para utilizar los certificados de clave pseudo-pública, la responsabilidad legal de la entidad emisora puede, como cuestión práctica, ser limitada. Esto está en contraste con el certificado convencional, donde la autoridad de certificación no tiene ningún conocimiento previo de quién utilizará el certificado. En segundo lugar, la revocación de un certificado de clave pública es fácil, ya que la autoridad de certificación sólo tiene que notificar a aquellos verificadores autorizados que utilicen los certificados de clave pública.

[0035] Los certificados de la forma propuesta serán emitidos por el componente de certificación, que actúa como un servidor de certificados, tal como se muestra en la figura 12. Como los expertos en la materia apreciarán, el servidor comprenderá una serie de módulos lógicos que se pueden implementar en software, hardware, o una combinación de ambos. El usuario que desee obtener la certificación presentará una solicitud firmada digitalmente para esta entrada **1210** al servidor de certificados **1200**. Dicha solicitud típicamente contiene la clave pública del usuario al que va a ser certificado, junto con su nombre u otros atributos de identificación. El servidor de certificados verificará la firma digital del usuario utilizando la clave pública presentada. Si la firma se verifica correctamente, el servidor comprobará la información de la identidad del usuario en la base de datos **1220,** y luego emitirá un certificado de clave pública **1230** de la forma propuesta como salida. Los expertos en la materia reconocerán que la base de datos de identidad del usuario podría ser suplantada por otras fuentes de información para verificar la identidad del usuario que solicita el certificado.

[0036] Una realización alternativa del servidor certificados pseudo-públicos podría incluir una unidad de modificaciones a conectarse a un servidor de certificados convencional. Esta unidad añadida podría funcionar en la entrada o en la salida del servidor de certificados convencional. En caso de que la unidad de modificaciones funcione en la entrada, volvería a empaquetar la solicitud de certificado mediante el encriptado de la clave pública del usuario, e incluiría la clave pública encriptada entre los atributos de identificación. La unidad de modificaciones entonces adjuntaría una clave pública ficticia a la solicitud, firmaría la solicitud con la clave privada asociada y pasaría la solicitud al servidor de certificados convencional. La salida del servidor de certificados convencional sería un certificado que contiene la clave pública encriptada del usuario como uno de los atributos de identificación. En caso de modificación, la unidad opera en la salida de un servidor de certificados convencional, la unidad volvería a empaquetar el certificado convencional producido por el servidor de certificados convencional mediante la encriptación del exponente de clave pública en el certificado in situ, y luego sobrescribiría la firma del servidor de certificados con una nueva firma del certificado modificado. Las personas expertas en la materia apreciarán que otras realizaciones alternativas son posibles.

3. Detalles del componente de generación de claves

[0037] Este componente genera las claves privada y pública de un usuario en el tiempo de preparación, cuando el usuario crea sus credenciales. La creación de la clave pública (ya sea en el sentido convencional o en el sentido pseudo-público) en este aspecto de la presente invención es generalmente similar a las técnicas convencionales de generación de claves, pero con una ligera modificación para resistir el ataque siguiente.

a. Ataque exponente de clave pública conocida

[0038] Este es un ataque que es particular para el criptosistema RSA, y es una variante del ataque clave pública conocido que se describe anteriormente. En el sistema RSA, es común el uso de claves públicas con exponentes simples fijos (por ejemplo, 3 ó 65537) para acelerar las operaciones criptográficas. Desafortunadamente, esto hace posible que el hacker malicioso monte un ataque de clave pública conocida. El hacker probará todos los PINs disponibles en la cartera de claves, y para cada PIN pseudo-válido, extraerá la clave privada desencriptada y la separará en el exponente privado y el módulo. Como que una clave pública RSA consiste en el exponente conocido y el mismo módulo, el hacker puede combinar los dos para montar una clave pública candidata. A continuación, podría montar el ataque de clave pública conocida que se ha descrito anteriormente. Para resistir este ataque, el aspecto de generación de claves de la presente invención puede utilizar claves públicas con exponentes largos, digamos 64 a 128 bits, que se generan aleatoriamente en el momento de generación de claves.

4. Detalles del componente de verificación

[0039] El componente de verificación de la presente invención se diferencia de dos maneras del componente de verificación en sistemas convencionales. El componente de verificación debe respetar la naturaleza pseudo-pública del certificado de clave pública, y adoptar las medidas adecuadas para extraer la clave pública del usuario de un certificado antes de la verificación de la firma del usuario. En una realización de ejemplo de este aspecto de la invención, estos incluyen la recepción de un certificado que contiene una clave pseudo-pública encriptada del titular del certificado, y el uso de la clave privada de un verificador autorizado para desencriptar la clave pseudo-pública. El componente de verificación a continuación utiliza la clave pseudo-pública para verificar una firma digital en un mensaje enviado por el titular del certificado. En una realización alternativa, si una clave DES se ha utilizado para encriptar la clave pseudo-pública, la clave DES primero se desencriptará con la clave privada del verificador, y a su vez la clave DES se utilizará para desencriptar la clave pseudo-pública. No importa cuál sea el mecanismo de desencriptado, el componente de verificación también debe incluir lógica para detectar intentos de intrusión de hackers fraudulentos, por ejemplo, los que firman mensajes con claves candidatas privadas incorrectas correspondientes a los códigos de acceso pseudo-válidos de los aspectos de la cartera de claves de la presente invención. En tal caso, un hacker fraudulento podría robar u obtener de otra manera el certificado pseudo-público del usuario legítimo y enviar el certificado junto con un mensaje fraudulento firmado con la clave privada candidata incorrecta. La inconsistencia entre la clave pseudo-pública legítima del usuario correcto en el certificado y la clave privada del candidato incorrecta permite la detección del usuario fraudulento. En particular, en una realización, si la firma de un usuario particular no se verifica en tres intentos sucesivos, el componente de verificación concluye que un robo podría estar en marcha, y congela los privilegios de acceso del usuario a la espera de una investigación adicional. Además de (o en lugar de) la congelación del acceso, el componente de verificación puede hacer sonar una alarma alertando a un operador del intento de robo. Hay otros procedimientos de detección de intentos de robo en el componente de verificación, y otros posibles cursos de acción sobre la detección de un robo. Tal como los expertos en la materia apreciarán, el componente de verificación comprometerá una serie de módulos lógicos que se pueden implementar en software, hardware, o una combinación de ambos.

5. Modificaciones, mejoras y realizaciones alternativas

[0040] Lo anterior ha descrito diversos aspectos de la invención. Aunque en una realización preferida, la cartera de claves, el componente de generación de claves, el componente de verificación de claves y el componente de certificación de claves se utilizan conjuntamente para proporcionar una tecnología segura para el almacenamiento de claves criptográficas y su utilización, los expertos en la materia apreciarán que en realizaciones alternativas, varios subconjuntos de todo el sistema también se pueden combinar para aplicaciones específicas que no requieren todos los componentes.

[0041] Además, a pesar de que todo lo anterior se ha descrito respecto a un sistema basado en software, esto no es estrictamente necesario. Por ejemplo, algunos o todos los componentes pueden ser desplegados utilizando microcódigos y PLAs o ROMs, lenguaje de programación de propósito general y microprocesadores de propósito general, o ASICs. Es decir, la invención no se limita al software en sí mismo, sino que podría desplegarse en prácticamente cualquier forma de lógica, incluido el software puro, una combinación de software y hardware, o incluso hardware solamente.

[0042] Además, a pesar de las varias realizaciones o aspectos se han descrito respecto a la criptografía RSA (para las claves públicas y/o pseudo-públicas y certificados públicos y/o pseudo-públicos) o criptografía DES (para el

encriptado de PIN y el almacenamiento de la clave pseudo-pública en el certificado pseudo-público), los expertos en la materia apreciarán que son posibles muchas modificaciones y mejoras en la tecnología de encriptado de tales ejemplos. Más generalmente, cada una de las operaciones mencionadas se pueden implementar de una amplia variedad de técnicas criptográficas, incluyendo muchos tipos de encriptación simétrica o asimétrica, así como los CCR, hashes, resúmenes de mensajes, u otras funciones de una vía. Por ejemplo, una operación de encriptado asimétrico puede ser sustituida por una (opcionalmente con clave) función de una vía donde la integridad es la principal preocupación, o el encriptado de una clave de sesión simétrica seguido por el uso de la clave de sesión para el encriptado de texto plano, y varias otras alternativas que son bien conocidas por los expertos en la materia. Finalmente, aunque la realización de ejemplo se ha descrito respecto a PINs que protegen una clave privada, los expertos en la materia se darán cuenta de que la misma tecnología de camuflaje criptográfico puede utilizarse con otros tipos de códigos de acceso y las representaciones criptográficas para proteger cualquier artículo de datos de control de acceso. Por lo tanto, se pretende que el alcance de la invención esté limitado sólo por las reivindicaciones adjuntas a continuación.

REIVINDICACIONES

1. Aparato para gestionar el acceso a un elemento de datos de control de acceso criptográficamente seguro, que comprende:

a) una lógica de entrada (300) configurada para recibir un código de acceso candidato;

b) una primera memoria (320) configurada para almacenar un elemento de datos de control de acceso;

c) una primera lógica primer criptográfica (310, 325) operativamente conectada con dicha lógica de entrada y con dicha primera memoria y configurada para procesar dicho elemento de datos de control de acceso utilizando dicho código de acceso candidato para proporcionar un elemento de datos de control de acceso procesado para una pluralidad, pero no para todos de los códigos de acceso candidatos, preservando al mismo tiempo un formato estructural característico para dichos elementos de datos de control de acceso, donde sólo uno de los elementos de datos de control de acceso es válido, y en el que dicha primera lógica criptográfica se configura para detectar un intento de utilizar un código de acceso inválido; y

d) una lógica de salida (335) configurada para proporcionar dicho elemento de datos de control de acceso procesado punto a un usuario de dicho aparato.

2. Aparato según la reivindicación 1, en el que:

e) al menos parte de dicho elemento de datos de control de acceso se ha encriptado usando un código de acceso correcto;

f) una segunda memoria está configurada para almacenar una representación criptográfica de dicho código de acceso correcto;

g) dicha primera lógica criptográfica incluye:

i) una segunda lógica criptográfica operativamente conectada a dicha lógica de entrada y está configurada para regenerar dicha representación criptográfica de dicho código de acceso correcto en respuesta a dicho código de acceso candidato perteneciente a una pluralidad de códigos de acceso potencialmente válidos; y

ii) una tercera lógica criptográfica configurada para recibir dicha representación criptográfica regenerada de dicha segunda lógica criptográfica, y conectada operativamente con dicha primera memoria y dicha lógica de entrada para utilizar dicho código de acceso candidato recibido en desencriptar dicho elemento de datos de control de acceso encriptado almacenado para producir un elemento de datos de control de acceso desencriptado.

3. Aparato según la reivindicación 2, en el que dicho elemento de datos de control de acceso es una clave criptográfica.

4. Aparato según la reivindicación 3, en el que la clave criptográfica es una clave privada.

5. Aparato según la reivindicación 4, que también comprende una primera clave pública correspondiente a dicha clave privada, en el que dicha primera clave pública sólo es accesible a personas autorizadas.

6. Aparato según la reivindicación 5, que también comprende un certificado que contiene dicha primera clave pública, donde el certificado contiene dicha primera clave pública en forma encriptada.

7. Aparato según la reivindicación 6, en el que dicha primera llave pública se encripta con una segunda clave pública que tiene una clave privada correspondiente que no se conoce con excepción para los verificadores autorizados.

8. Aparato según la reivindicación 4, en el que dicha clave privada incluye un módulo que no tienen factores pequeños, y un exponente menor que dicho módulo.

9. Aparato según la reivindicación 4, en el que dicha clave privada incluye una representación de texto plano de dicho módulo, y una representación criptográfica de al menos una parte de un exponente correspondiente a dicho módulo.

10. Aparato según la reivindicación 9, que también comprende una tercera memoria configurada para almacenar un número mayor que dicho exponente y menor que dicho módulo; y en el que al menos parte de dicho exponente se almacena en una forma expandida que, cuando se evalúa el módulo de dicho número, es igual a dicha al menos una parte de dicho exponente.

11. Aparato de la reivindicación 9, en el que dicha al menos parte de dicho exponente representa al menos un bit menos significativo de dicho exponente.

12. Aparato según la reivindicación 3, en el que dicha segunda lógica criptográfica configurada para regenerar dicha representación criptográfica de dicho código de acceso correcto incluye un hash en el cual una pluralidad de entradas producen la misma salida hash.

13. Aparato según la reivindicación 12, en el que dicho hash se aplica de manera que dicha pluralidad de códigos de acceso potencialmente válidos se distribuyen uniformemente entre una pluralidad de códigos de acceso inválidos.

14. Aparato según la reivindicación 2, en el que dicha representación criptográfica incluye una función hash; y dicha segunda lógica criptográfica configurada para regenerar dicha representación criptográfica de dicho código de acceso incluye un hash en el cual una pluralidad de entradas producen la misma salida hash.

15. Aparato según la reivindicación 14, en el que dicho hash se aplica de manera que dicha pluralidad de códigos de acceso potencialmente válidos se distribuyen uniformemente entre una pluralidad de códigos de acceso inválidos.

16. Aparato según la reivindicación 14, en el que dicho elemento de datos de control de acceso es una clave privada.

17. Aparato según la reivindicación 16, que también comprende una primera clave pública correspondiente a dicha clave privada, donde dicha primera clave pública sólo es accesible para partes autorizadas.

18. Aparato según la reivindicación 16, que también comprende lógica de firma digital, que incluye:

h) una lógica de entrada configurada para recibir un mensaje para ser firmado;

i) una lógica aleatoria configurada para generar datos aleatorios; y

j) una cuarta lógica criptográfica operativamente conectada a dicha lógica de entrada y dicha lógica aleatoria y configurada para:

i) rellenar dicho mensaje recibido con dichos datos aleatorios generados;

ii) firmar dicho mensaje rellenado con dicho elemento de datos de control de acceso desencriptados.

19. Aparato según la reivindicación 2, en el que dicha tercera lógica criptográfica está configurada para no permitir dicho desencriptado cuando dicho código de acceso candidato recibido es un código de acceso inválido.

20. Aparato según la reivindicación 2, implementado como un dispositivo de hardware.

21. Aparato según la reivindicación 2, que también comprende lógica de firma digital, que incluye:

h) una lógica de entrada configurada para recibir un mensaje para ser firmado;

i) una lógica aleatoria configurada para generar datos aleatorios; y

j) una cuarta lógica criptográfica operativamente conectada a dicha lógica de entrada y dicha lógica aleatoria y configurada para:

i) rellenar dicho mensaje recibido con dichos datos aleatorios generados; y

ii) firmar dicho mensaje rellenado con dicho elemento de datos de control de acceso desencriptado.

22. Aparato según la reivindicación 21, en el que dichos datos aleatorios generados se producen a partir de una fuente externa de dicho aparato.

23. Aparato según la reivindicación 21, en el que dichos datos aleatorios generados se originan en un almacén físico.

24. Aparato según la reivindicación 2, en el que dicha tercera lógica criptográfica para desencriptar incluye una función criptográfica simétrica.

25. Aparato según la reivindicación 24, en el que dicha función criptográfica simétrica es DES.

26. Aparato según la reivindicación 1, en el que dicho elemento de datos de control de acceso almacenado es una clave privada que tiene una clave pública correspondiente que incluye un exponente.

27. Procedimiento para proporcionar un elemento de datos de control de acceso criptográficamente seguro almacenado, que comprende las etapas de:

a) recibir un código de acceso candidato de un usuario;

b) acceder a un elemento de datos de control de acceso almacenado;

c) procesar criptográficamente dicho elemento de datos de control de acceso utilizando dicho código de acceso candidato para proporcionar un elemento de datos de control de acceso procesado a partir de una pluralidad pero no todos los códigos de acceso candidatos, preservando al mismo tiempo un formato estructural característico para dicho elemento de datos de control de acceso, en el que sólo uno de los elementos de datos de control de acceso tratados es válido, y en el que dicho procesamiento incluye la detección de un intento de utilizar un código de acceso inválido; y

d) proporcionar a dicho usuario dicho elemento de datos de control de acceso procesado.

28. Procedimiento según la reivindicación 27, en el que dicha etapa de procesamiento criptográfico de dicho elemento de datos de control de acceso procesado criptográficamente utilizando dicho código de acceso candidato incluye:

e) acceder, a partir de una primera memoria, a un elemento de datos de control de acceso, por lo menos parte del cual se ha encriptado usando un código de acceso correcto;

f) acceder, desde una segunda memoria, a una representación criptográfica de dicho código de acceso;

g) regenerar dicha representación criptográfica de dicho código de acceso correcto en respuesta a dicho código de acceso candidato perteneciente a una pluralidad de códigos de acceso potencialmente válidos; y

h) usar dicho código de acceso candidato recibido, desencriptar dicho elemento de datos de control de acceso encriptado para producir un elemento de datos de control de acceso desencriptado.

29. Procedimiento según la reivindicación 28, en el que dicho elemento de datos de control de acceso es una clave criptográfica.

30. Procedimiento según la reivindicación 29, en el que dicha clave criptográfica es una clave privada.

31. Procedimiento según la reivindicación 30, en el que dicha clave privada es un elemento de un par de claves criptográficas que incluye una primera clave pública correspondiente a dicha clave privada, en el que dicha primera clave pública sólo es accesible a partes autorizadas.

32. Procedimiento según la reivindicación 31, en el que dicha primera memoria o dicha segunda memoria incluye un certificado que contiene dicha primera clave pública, donde el certificado contiene dicha primera clave pública en forma encriptada.

33. Procedimiento según la reivindicación 32, en el que dicha primera llave pública se encripta con una segunda clave pública que tiene una clave privada correspondiente que no se conoce con excepción de los verificadores autorizados.

34. Procedimiento según la reivindicación 31, en el que dicha clave privada incluye un módulo que no tiene ningún factor pequeño, y un exponente menor que dicho módulo.

35. Procedimiento según la reivindicación 31, en el que dicha clave privada incluye una representación de texto plano de dicho módulo; y una representación criptográfica de al menos una parte de un exponente correspondiente a dicho módulo.

36. Procedimiento según la reivindicación 35, en el que:

i) dicha clave privada se almacena en dicha primera memoria como una forma expandida de al menos parte de dicho exponente; y

j) dicha etapa de desencriptar dicho elemento de datos de control de acceso encriptado incluye:

i) recuperar de dicha tercera memoria un número mayor que dicho exponente y menor que dicho módulo;

ii) recuperar dicha forma expandida de al menos parte de dicho exponente de dicha primera memoria; y

iii) evaluar dicha forma expandida de al menos parte de dicho exponente, módulo de dicho número, para recuperar dicha al menos parte de dicho exponente.

37. Procedimiento según la reivindicación 35, en el que dicha al menos parte de dicho exponente representa al menos un bit menos significativo de dicho exponente.

38. Procedimiento según la reivindicación 30, en el que dicha etapa de regeneración de dicha representación criptográfica de dicho código de acceso incluye realizar un hash en el cual una pluralidad de entradas producen la misma salida hash.

39. Procedimiento según la reivindicación 38, en el que dicho hash se aplica de manera que dicha pluralidad de códigos de acceso potencialmente válidos se distribuyen uniformemente entre una pluralidad de códigos de acceso inválidos.

40. Procedimiento según la reivindicación 29, en el que dicha representación criptográfica incluye una función hash, y dicha etapa de regeneración de dicha representación criptográfica de dicho código de acceso incluye realizar un hash en el cual una pluralidad de entradas producen la misma salida hash.

41. Procedimiento según la reivindicación 40, en el que dicho hash se aplica de manera que dicha pluralidad de códigos de acceso potencialmente válidos se distribuyen uniformemente entre una pluralidad de códigos de acceso inválidos.

42. Procedimiento según la reivindicación 40, en el que dicho elemento de datos de control de acceso es una clave privada.

43. Procedimiento según la reivindicación 42, en el que dicha clave privada es un elemento de un par de claves criptográficas que incluyen una clave pública correspondiente a dicha clave privada.

44. Procedimiento según la reivindicación 42, que también comprende las etapas de:

i) recibir un mensaje para ser firmado;

j) generar datos aleatorios;

k) rellenar dicho mensaje recibido con dichos datos aleatorios generados; y

l) firmar dicho mensaje rellenado con dicho elemento de datos de control de acceso desencriptado.

45. Procedimiento según la reivindicación 29, en el que dicha etapa de desencriptar dicho elemento de datos de control de acceso no está permitida cuando dicho código de acceso candidato recibido es un código de acceso inválido.

46. Procedimiento según la reivindicación 29, que también comprende las etapas de:

i) recibir un mensaje para ser firmado;

j) generar datos aleatorios;

k) rellenar dicho mensaje recibido con dichos datos aleatorios generados; y

l) firmar dicho mensaje rellenado con dicho elemento de datos de control de acceso desencriptado.

47. Procedimiento según la reivindicación 46, en el que los datos aleatorios generados se producen a partir de una fuente externa de dicho aparato.

48. Procedimiento según la reivindicación 46, en el que los datos aleatorios generados se originan a partir de una fuente física.

49. Procedimiento según la reivindicación 29, en el que la etapa de desencriptar dicho elemento de datos de control

de acceso encriptado incluye realizar una operación criptográfica simétrica en el mismo.

50. Procedimiento según la reivindicación 49, en el que dicha operación criptográfica simétrica es DES.

51. Procedimiento según la reivindicación 29, en el que dicho elemento de datos de control de acceso almacenado es una clave privada que tiene una clave pública correspondiente que incluye un exponente.

52. Medio legible por ordenador en el que se almacena un programa de ordenador que, cuando se ejecuta en un ordenador, realiza las etapas del procedimiento según cualquiera de las reivindicaciones 27 a 51.

FIG. 1

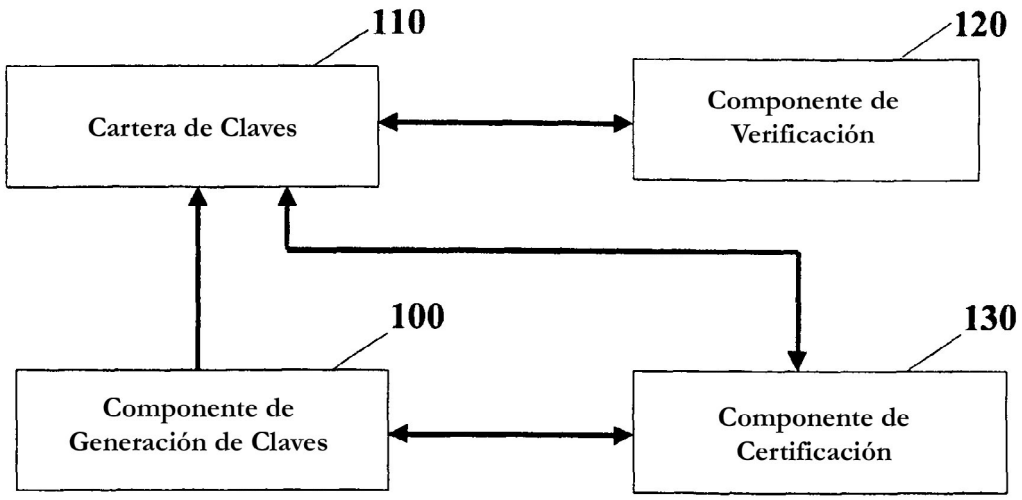

110
120
Cartera de Claves
Componente de Verificación
100
130
Componente de Generación de Claves
Componente de Certificación

FIG. 2

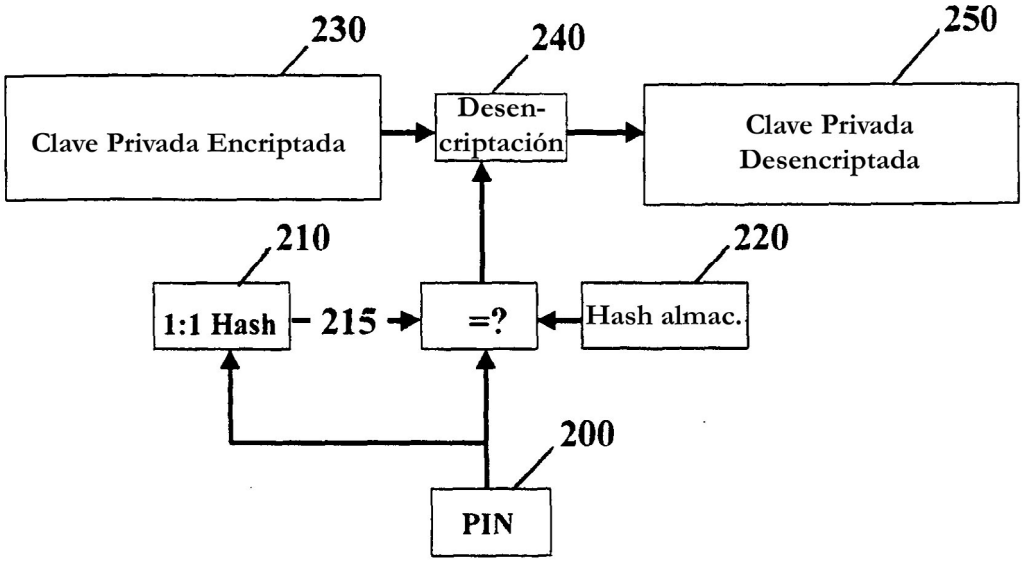

230
240
250
Clave Privada Encriptada
Desen-
criptación
Clave Privada
Desencriptada
210
220
1:1 Hash
215
=?
Hash almac.
200
PIN

FIG. 3

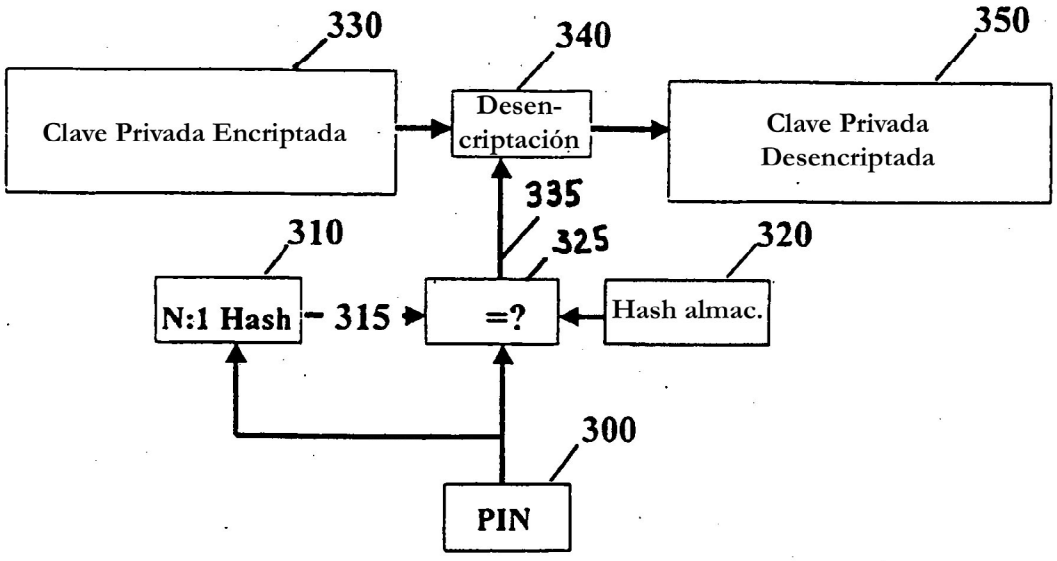

330
340
350
Clave Privada Encriptada
Desen-
criptación
Clave Privada
Desencriptada
335
310
325
320
N:1 Hash
315
=?
Hash almac.
300
PIN

FIG. 4

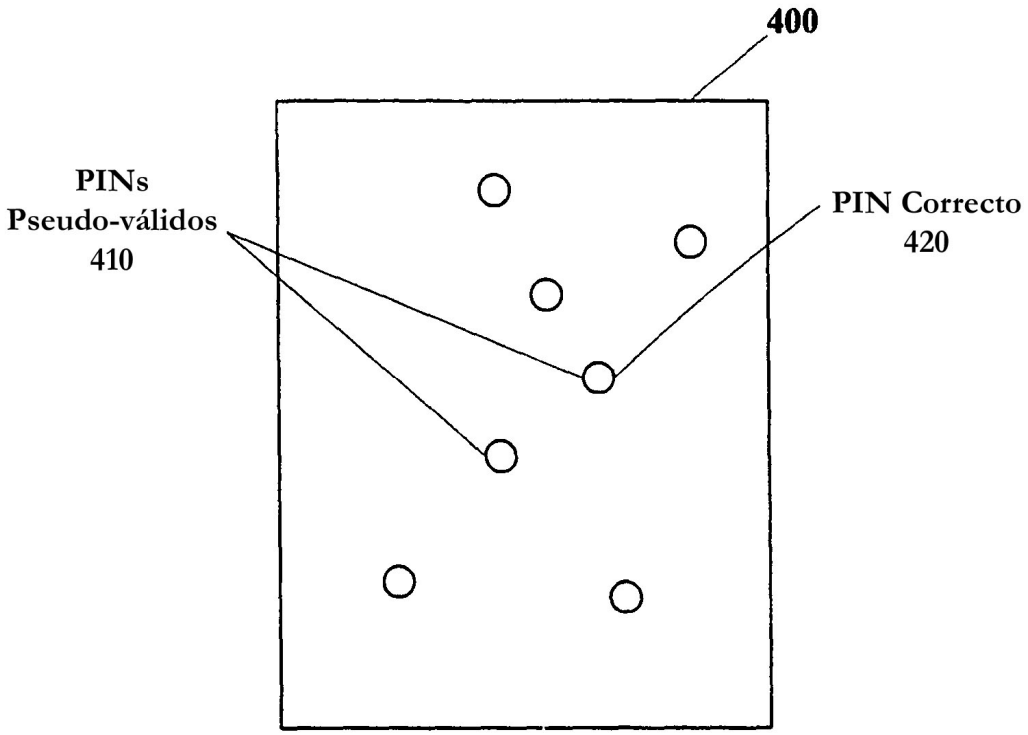


FIG. 5

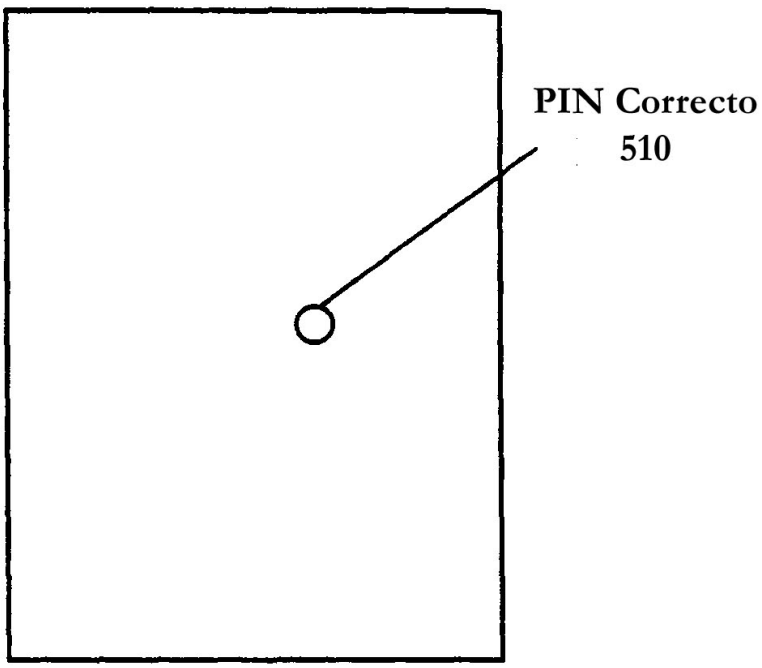


FIG. 6

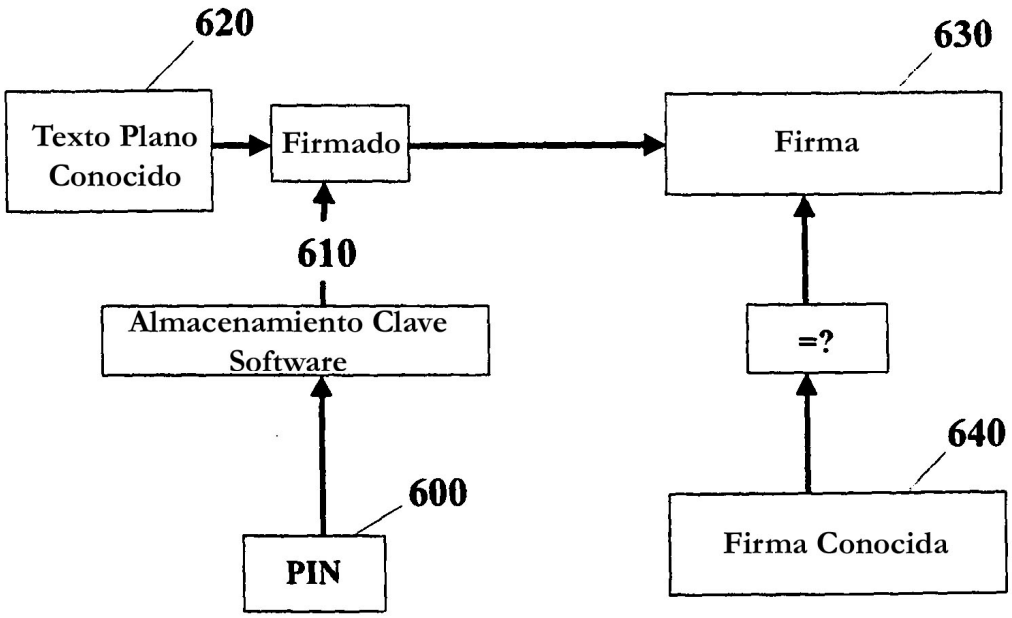

620
630
Texto Plano Conocido
Firmado
Firma
610
Almacenamiento Clave Software
=?
640
600
PIN
Firma Conocida

FIG. 7

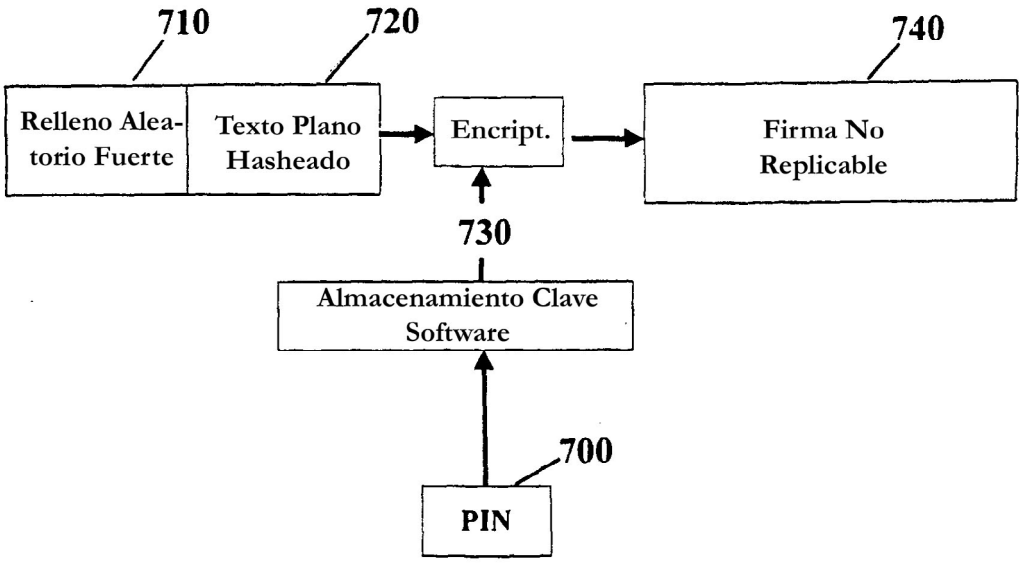

710
720
740
Relleno Aleatorio Fuerte
Texto Plano Hasheado
Encript.
Firma No Replicable
730
Almacenamiento Clave Software
700
PIN

FIG. 8

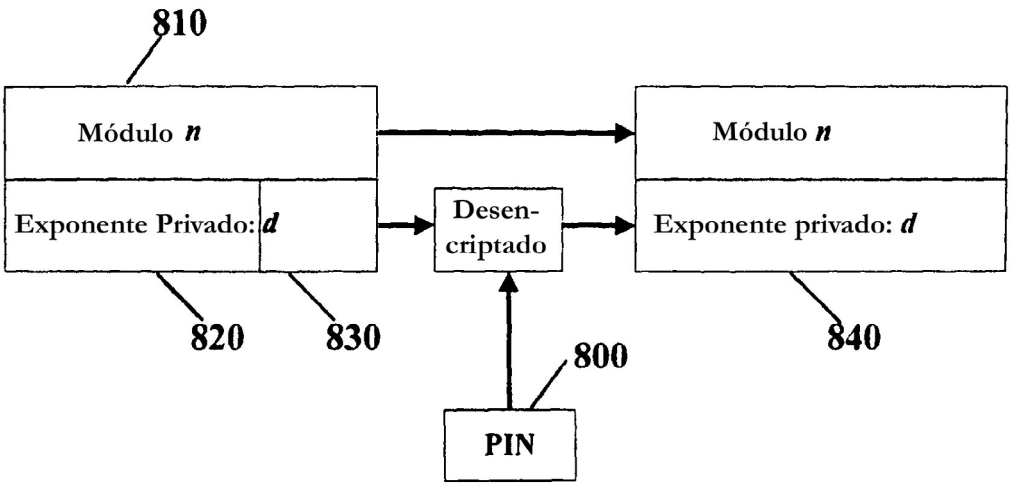

810
Módulo n
Exponente Privado: d
Desen-
criptado
Módulo n
Exponente privado: d
820
830
800
840
PIN

FIG. 9

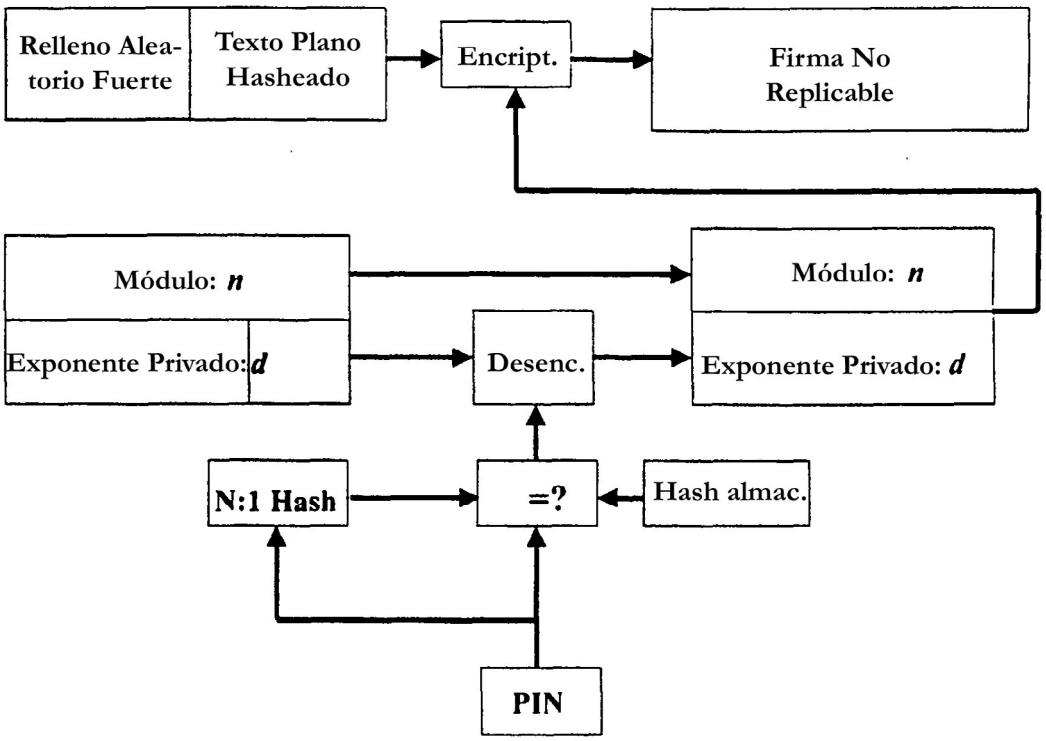

Relleno Aleatorio Fuerte
Texto Plano Hasheado
Encript.
Firma No Replicable
Módulo: n
Exponente Privado: d
Desenc.
Módulo: n
Exponente Privado: d
N:1 Hash
=?
Hash almac.
PIN

FIG. 10

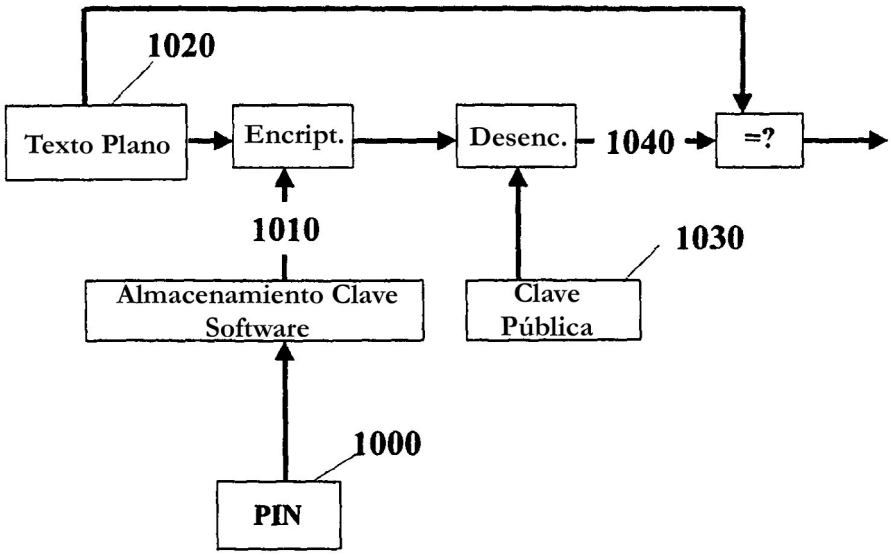

1020
Texto Plano
Encript.
Desenc.
1040
=?
1010
1030
Almacenamiento Clave Software
Clave Pública
1000
PIN

FIG. 11

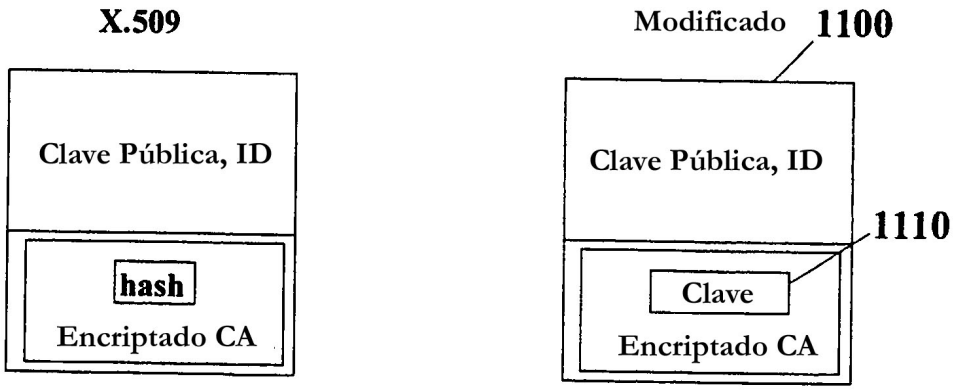
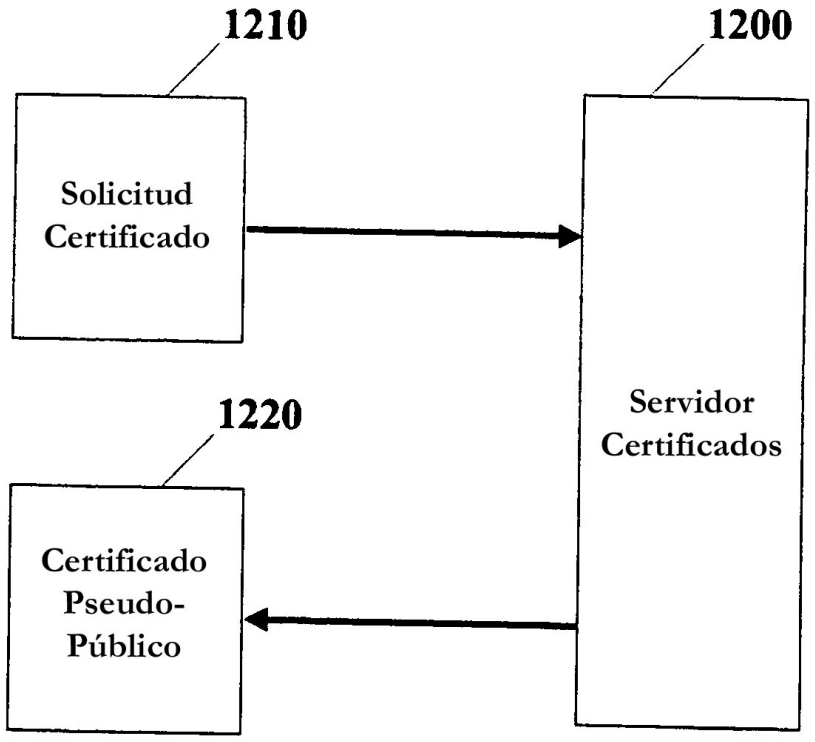

X.509
Clave Pública, ID
hash
Encriptado CA
Modificado
1100
Clave Pública, ID
1110
Clave
Encriptado CA

FIG. 12


1210
1200
Solicitud Certificado
Servidor Certificados
1220
Certificado Pseudo-Público