(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 717 999**

(51) Int. CI.:

H04L 9/06 (2006.01)
H04L 9/32 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **01.12.2015 PCT/EP2015/078115**

(87) Fecha y número de publicación internacional: **09.06.2016 WO16087395**

(96) Fecha de presentación y número de la solicitud europea: **01.12.2015 E 15801867 (1)**

(97) Fecha y número de publicación de la concesión europea: **20.02.2019 EP 3228044**

(54) Título: **Método criptográfico por bloques para cifrar/descifrar mensajes y dispositivos criptográficos para implementar este método**

(30) Prioridad:

03.12.2014 EP 14196089

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
26.06.2019

(73) Titular/es:

NAGRAVISION S.A. (100.0%)
22-24, route de Genève
1033 Cheseaux-sur-Lausanne, CH

(72) Inventor/es:

WYSEUR, BRECHT

(74) Agente/Representante:

TOMAS GIL, Tesifonte Enrique

DESCRIPCIÓN

Método criptográfico por bloques para cifrar/descifrar mensajes y dispositivos criptográficos para implementar este método

CAMPO TÉCNICO

La presente invención se refiere al campo del cifrado de mensajes, usando un modo de operación de cifrado por bloques para el cifrado autenticado (AE), y el descifrado de mensajes usando el modo de operación inverso. Más específicamente, esta invención se refiere a esquemas de cifrado autenticado y, en particular, se refiere a una mejora del llamado esquema de retroalimentación de cifrado de contador (CCFB). En una variante, esta invención comprende también una extensión de los esquemas de cifrado autenticado con datos asociados (AEAD), en particular, a una mejora de un esquema CCFB+H. Finalmente, esta invención también se refiere a la operación criptográfica inversa que utiliza el mismo esquema.

ANTECEDENTES

Cuando debe enviarse un mensaje confidencial desde una primera entidad a una segunda entidad, es común proteger la privacidad del mensaje y su integridad/autenticidad. En el pasado, estos dos objetivos de seguridad se han manejado por separado, debido a que se consideró que la protección de la integridad, típicamente manejada usando códigos de autenticación de mensaje (MAC) o firmas digitales, fue un objetivo que debía de alcanzarse por medios completamente diferentes.

Después de observar que puede ser difícil implementar un modo de confidencialidad con un modo de autenticidad separado y puede ser propenso a errores, se sugirió suministrar modos de operación que combinen tanto confidencialidad como integridad/autenticidad en un solo esquema criptográfico. Un modo de operación puede considerarse como una técnica para mejorar el efecto de un algoritmo criptográfico, en particular cualquier algoritmo basado en un esquema de cifrado simétrico. Los esquemas de cifrado autenticado (AE) están diseñados para proteger simultáneamente la privacidad y la integridad/autenticidad de los mensajes procesados en un solo modo de operación compacto. Como ejemplo, cuando un proveedor desea enviar información confidencial a un cliente, se requiere la privacidad de los datos para garantizar que la información sigue siendo confidencial. Además, se requiere además la integridad y autenticidad para garantizar que la entidad que ha enviado la información es realmente el proveedor mencionado anteriormente y que la información no se modificó durante su tránsito.

Los modos AE pueden clasificarse de acuerdo con el número de pases por los datos que requieren. Algunos modos AE utilizan únicamente un pase por los datos, mientras que otros utilizan dos pases. Los modos de un solo pase proporcionan el cifrado autenticado justo después del procesamiento único de un mensaje. Los modos de dos pases procesan datos en dos fases (usando aún una clave tanto para el cifrado como para la autenticación).

Cuando se cifra un mensaje por medio de tal esquema de cifrado autenticado "integrado", el texto cifrado puede enviarse junto con información adicional. Esto significa que puede adjuntarse un encabezado sin cifrar al mensaje de texto cifrado. Tal encabezado de paquete (encabezado de texto sin cifrar unido al mensaje de texto plano) necesita autenticarse junto con el mensaje, pero no necesita cifrarse. De hecho, típicamente en un paquete de red, la carga útil debe cifrarse y autenticarse, pero el encabezado únicamente debe autenticarse (es decir, debe permanecer sin cifrar). La razón es que los enrutadores deben poder leer los encabezados de los paquetes a fin de saber cómo enrutarlos adecuadamente. Algunos modos de operación son específicamente adecuados para autenticar eficientemente los datos asociados con el mensaje de texto cifrado. Dichos modos se denominan como cifrado autenticado con datos asociados (AEAD). Estos permiten que se incluyan los datos asociados incluidos como entrada en sus esquemas.

Una de las desventajas de los esquemas AE y AEAD es que son propensos a ataques de análisis de energía diferencial (DPA) de primer orden. El análisis de energía es una forma de ataque de canal lateral (ataque no invasivo) en la que el hacker estudia el consumo de energía de un dispositivo criptográfico. Tal dispositivo puede referirse a una tarjeta de microprocesador, una credencial, una clave, un chip de circuito integrado de una tarjeta de circuito impreso, un módulo de seguridad dentro de un módulo decodificador o pueden ser funciones implementadas únicamente en forma de software. Al estudiar la entrada y salida de corriente y tensión de un circuito (o la sincronización del dispositivo o incluso las emisiones electromagnéticas), puede ser posible descubrir la información confidencial tal como las claves criptográficas utilizadas durante las operaciones normales del dispositivo. De hecho, algunas operaciones requieren más consumo de energía.

Por lo tanto, usando un osciloscopio digital y un ordenador personal convencional, el análisis de las variaciones y picos puede extraer información valiosa para el criptoanalista.

El análisis simple de energía (SPA) recupera las claves confidenciales a través de la observación directa del consumo de energía del dispositivo, mientras que los ataques DPA emplean técnicas estadísticas para extraer la información de la clave confidencial a partir de múltiples mediciones del consumo de energía. DPA es una forma avanzada del análisis de energía que permite computar valores intermedios dentro de los cálculos criptográficos analizando estadísticamente los datos recopilados de múltiples operaciones criptográficas, típicamente de miles de operaciones. Los ataques DPA tienen propiedades de procesamiento de señal y de corrección de errores que son capaces de extraer información confidencial de las mediciones que contienen demasiado ruido para analizarse utilizando SPA. Usando un ataque DPA, un hacker puede obtener claves confidenciales analizando las mediciones del consumo de energía de múltiples operaciones criptográficas realizadas en sistemas criptográficos vulnerables.

Para abordar dichos ataques de canal lateral, se desarrollaron contramedidas. Algunas de estas contramedidas implican modificaciones del algoritmo criptográfico, por ejemplo, realizando operaciones criptográficas sobre los datos que se refieren al valor real mediante la relación matemática que sobrevive a la operación criptográfica. Un enfoque implica parámetros de obstrucción para aleatorizar datos manipulados por el sistema criptográfico mientras aún se produce el resultado correcto. La información impredecible se incorpora en valores intermedios para reducir la correlación entre las mediciones del consumo de energía (canales laterales) y los valores intermedios originales. De acuerdo con otro enfoque, es posible enmascarar la señal original reduciendo intencionadamente la amplitud de la señal para disminuir la relación señal-ruido (S/N). En una variante, se puede añadir la amplitud o el ruido temporal en las mediciones del consumo de energía para disminuir la relación S/N. Otras contramedidas pueden implicar modificaciones de hardware, por ejemplo, al variar la frecuencia del reloj interno de un chip para desincronizar las señales eléctricas, o puede implicar modificaciones del protocolo criptográfico, por ejemplo, renovando y actualizando la información confidencial durante la vida útil del sistema criptográfico.

Sin embargo, evitar los ataques DPA supone un gran reto, ya que incluso pequeñas variaciones en el consumo de energía pueden conducir a debilidades aprovechables. Además, con frecuencia no existe un libre acceso a dichas contramedidas debido a la protección conferida por las patentes. Por consiguiente, existe la necesidad de proporcionar un nuevo esquema AE a fin de superar, al menos parcialmente, estas desventajas.

El documento XP047030081 titulado *"Two-Pass Authenticated Encryption Faster Than Generic Composition"* (de Stefan Lucks), se refiere a esquemas de cifrado autenticados por CCFB (con o sin datos asociados H) y, más específicamente, propone otro esquema de dos pases con una separación diferente de tareas entre los pases. El primer pase es para la privacidad y la autenticación "local", mientras que el segundo calcula una única etiqueta de autenticación "global" de las locales. En cada ronda, solo una parte (por ejemplo, un segmento de 8 bits) del criptograma proporcionado por el algoritmo de cifrado por bloques se conserva para combinarse sucesivamente con los bloques de mensajes resultantes de la división del mensaje de texto simple por bloques; descartándose la otra parte (segmento).

RESUMEN DE LA INVENCIÓN

En lugar de proporcionar nuevas contramedidas para proteger a los sistemas criptográficos contra ataques DPA, la presente invención sugiere un modo de operación mejorado, mientras se mantienen en mente los ataques DPA. Para este fin, la presente invención se refiere a un modo de operación, basándose en el esquema CCFB, que es más adecuado para mitigar los ataques DPA. Más particularmente, la presente invención sugiere un método de cifrado por bloques que utiliza repetición de rondas para proporcionar, a partir de un mensaje de texto simple, un mensaje de texto cifrado y una etiqueta global para autenticar dicho mensaje de texto cifrado. El mensaje de texto simple se convierte (en particular dividiendo y rellenando si lo hubiera) en una pluralidad de bloques de texto simple ordenados que tienen todos la misma longitud de bits. Cada uno de estos bloques de texto simple se procesa sucesivamente de una forma ordenada como un bloque de texto simple normal durante una ronda. El método comprende las siguientes acciones:

- cargar los primeros datos en un registro,
- ejecutar la ronda realizando las siguientes etapas:
- una primera etapa para introducir datos de entrada en un algoritmo de cifrado por bloques, comprendiendo dichos datos de entrada, datos del registro y datos reproducibles, por ejemplo, los datos de contador o cualquier otra serie de datos reproducibles,

- una segunda etapa para computar, dentro del algoritmo de cifrado por bloques, un criptograma encriptando los datos de entrada usando una sola clave criptográfica, comprendiendo este criptograma un primer segmento y un segundo segmento, definiéndose éste último como una etiqueta local de autenticación,
- una tercera etapa para computar un bloque de texto cifrado realizando una primera operación usando como operandos: el primer segmento, el bloque de texto simple actual, y el segundo segmento,
- siempre y cuando no se hayan procesado todos los bloques de texto simple mencionados anteriormente en tal ronda, volver a cargar el registro con el último bloque de texto cifrado computado y aplicar una función de actualización, por ejemplo, una función de aumento, para los datos reproducibles antes de ejecutar una nueva ronda iniciando de nuevo dicha primera etapa,
- determinar el mensaje de texto cifrado al concatenar los bloques de texto cifrado proporcionados por todas las rondas de una forma ordenada, preferiblemente en la forma ordenada mencionada anteriormente, después
- volver a cargar el registro con el último bloque de texto cifrado computado y aplicar la función de actualización a los datos reproducibles antes de realizar solamente las primeras dos etapas de la ronda, después
- computar la etiqueta global mediante una segunda operación usando todas las etiquetas locales de autenticación computadas como operandos.

A partir del método anterior, debe apreciarse que el nuevo modo de operación de la presente invención se basa en un esquema de retroalimentación de cifrado de contador (CCFB). En sentido estricto, el esquema CCFB es un esquema AE de dos pases cuya estructura se asemeja mucho a las soluciones de un pase. El método innovador divulgado adopta preferiblemente una combinación inusual de dos modos de operación, es decir un modo de contador (CTR) y un modo de retroalimentación de cifrado (CFB), para proporcionar tanto el cifrado como la autenticación. De acuerdo con esta forma de realización, una etiqueta local de autenticación, que es resultado del cálculo del segundo segmento del criptograma, no se utiliza únicamente para computar una etiqueta global durante el segundo pase, sino que también se utiliza ventajosamente durante el primer pase para computar cada bloque de texto cifrado. Por lo tanto, se propone que pueden lograrse mejoras en el funcionamiento de los sistemas informáticos seguros, incluyendo, pero sin limitación, resistencia contra ataques conocidos.

Las formas de realización adicionales describen dispositivos para implementar los métodos mencionados anteriormente. Preferiblemente, uno de estos dispositivos está especialmente dedicado a cifrar mensajes, mientras que el otro dispositivo esta dedicado al proceso inverso. Dado que comprenden los mismos componentes, ambos de estos dispositivos son muy similares y se describirán con más detalle en la siguiente descripción detallada.

El objetivo y las ventajas de la presente invención se logran gracias al método criptográfico consistente con la materia objeto de las reivindicaciones del método independientes y gracias a los dispositivos consistentes con la materia objeto de las otras reivindicaciones del dispositivo independientes.

Otras ventajas y formas de realización se presentarán en la siguiente descripción detallada.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

La presente invención se entenderá mejor gracias a las figuras adjuntas, en las que:

La Figura 1 representa esquemáticamente el proceso de conversión de un mensaje de texto simple en un mensaje de texto cifrado,

la Figura 2 es un diagrama de flujo que muestra una forma de realización ejemplar del método criptográfico de la presente invención, más particularmente de la fase de cifrado,

la Figura 3 muestra otra forma de realización de acuerdo con una ilustración parcial del método en base a la Figura 2,

la Figura 4 es completamente similar a la Figura 2, pero se refiere a la fase de descifrado,

la Figura 5 muestra una forma de realización de un dispositivo para implementar cualquier forma de realización del método criptográfico de la presente invención, en particular para

implementar cualquier fase (cifrado o descifrado) de este método.

DESCRIPCIÓN DETALLADA

Haciendo referencia a la Figura 1, lo anterior muestra esquemáticamente un mensaje de texto simple (M) 10 procesado por la presente invención con los datos de entrada 20 adicionales, las etapas de conversión principales de este mensaje de texto simple 10 con dichos datos de entrada 20, y un mensaje de texto cifrado (C) 50 producido por el proceso con datos complementarios 52. Opcionalmente, el mensaje de texto simple 10 puede proporcionarse con datos asociados (AD) 15 que pueden relacionarse típicamente con un encabezado del mensaje de texto simple 10.

De acuerdo con una forma de realización preferida, los datos de entrada mencionados anteriormente 20 comprenden los primeros datos 21, por ejemplo, un vector de inicialización (IV), y datos reproducibles 22 que pueden inicializarse (reiniciar) a cualquier valor predeterminado. Estos datos reproducibles son preferiblemente los datos de contador (CDt), concretamente, datos tales como un valor proporcionado por un contador. Sin embargo, ya que no se requiere limitar dichos datos a los datos proporcionados por un contador en el estricto sentido, esto se refiere a los denominados datos reproducibles. La palabra "reproducible" excluye cualquiera de los datos aleatorios a menos que se hayan predefinido antes de utilizarse. Por lo tanto, debe entenderse que los datos reproducibles mencionados anteriormente se refieren a cualquiera de los datos tales como un valor que puede reproducirse gracias a un proceso o función reproducible que pueda considerarse como una función de actualización o una función de renovación. Por ejemplo, tal proceso o función puede referirse a una función hash o a una función de aumento o disminución aplicada a un valor inicial. También puede referirse a una lista predefinida o una serie de números aleatorios o pseudo-aleatorios tomados sucesivamente, en un cierto orden, como datos reproducibles. Preferiblemente, todos estos números serán diferentes para evitar cualquier repetición. En la presente descripción, la expresión "datos de contador" se utiliza de acuerdo con una forma de realización preferida, concretamente, como ejemplo no limitante. Los datos complementarios 52 se refieren a una etiqueta, en particular a una etiqueta global (T_G) que se determina de una pluralidad de etiquetas locales, como se explica además de aquí en adelante. La etiqueta global 52 se usa como datos de autenticación para autenticar el mensaje de texto cifrado 50 al final del proceso.

Como se muestra en la Figura 1, el mensaje de texto simple 10 se convierte en una pluralidad de bloques de texto simple (m_1, m_2, m_3,… m_i). En esta Figura, cada uno de estos bloques de texto simple se identifica por el número de referencia común 11. El proceso de conversión utilizado para convertir el mensaje de bloque de texto simple 10 en una pluralidad de bloques de texto simple 11, pretende principalmente dividir el mensaje de texto simple 10 en bloques que tengan la misma longitud de bits. En el caso en el que el mensaje de texto simple 10 tenga una longitud de bits que no sea un múltiplo del número de bits de los bloques de texto simple, el proceso de conversión comprenderá además una etapa para rellenar el menos uno de los dos bloques de texto simple terminales (por ejemplo, el último bloque de texto simple y/o el primero) con al menos un bit predeterminado (por ejemplo, con un bit igual a cero). El objetivo de la etapa de relleno es proporcionar al bloque terminal la misma longitud de bits que los otros bloques.

Los bloques de texto simple 11 han de procesarse en un orden específico, de manera que el mismo orden pueda aplicarse durante el proceso inverso con el objetivo de descifrar el mensaje de texto cifrado. Por consiguiente, los bloques de texto simple 11 se disponen de una forma ordenada. Como alternativa, a cada bloque se le asigna una posición identificable para determinar tal orden específico. Gracias al orden de estos bloques, será posible reconstruir el mensaje de texto simple 10 del mensaje de texto cifrado 50. Preferiblemente, cuando el mensaje de texto simple 10 se divide en bloques 11, los últimos no se mezclan ente sí, sino se conserva el orden en el que se han dividido. Conservar tal orden puede lograrse, por ejemplo, almacenando en un registro tanto el contenido del bloque como su posición dentro del mensaje 10. En una variante, puede asociarse un índice al contenido de cada bloque, para que pueda ser posible recuperar la posición apropiada de cualquier bloque m_1, m_2, m_3,…m_i, gracias a sus índices 1, 2, 3,… i.

Después, cada uno de estos bloques de texto simple 11 se procesa sucesivamente, como el denominado bloque de texto simple actual, en tal forma ordenada durante una ronda R del método. Al final de cada ronda, se obtiene un bloque de texto cifrado 51 en correspondencia con el bloque de texto simple actual 11. Por ejemplo, el bloque de texto simple m_1 mostrado en la Figura 1 proporciona el bloque de texto cifrado c_1 a la salida de la ronda.

Cada ronda proporciona también una etiqueta, en particular una etiqueta local 32 en correspondencia con bloque de texto cifrado 51 producido por la misma ronda. El conjunto de iteraciones que permiten computar

los bloques de texto cifrado 51 constituye un primer pase del método. En una forma de realización, el segundo pase tiene el objeto de determinar la etiqueta global 52 usando todas las etiquetas locales 32 que se han computado.

El mensaje de texto cifrado 50 se obtiene combinando entre sí todos los bloques de texto cifrado 51, en particular al concatenarlos para formar una cadena de acuerdo con la forma ordenada mencionada anteriormente, para conservar las posiciones respectivas entre los bloques 11 dentro del mensaje de texto simple 10 y los bloques correspondientes 51 dentro del mensaje de texto cifrado 50. Dado que los datos asociados 15 pueden ser típicamente el encabezado del mensaje 10, por lo tanto, permanecen sin cifrar y se adjuntan simplemente al mensaje de texto cifrado 50. Al final del proceso, el mensaje de texto cifrado 50, sus datos asociados 15 (si los hubiera) y la etiqueta global 52 forman entre sí tres entidades relacionadas unidas.

Un método preferido que implementa la presente invención se explicará ahora en más detalle gracias a la Figura 2 que muestra, a través de una pluralidad de iteraciones (IT_1, IT_2, IT_3,… IT_i), las acciones y etapas adoptadas durante la fase de cifrado. Cada una de estas iteraciones proporciona un bloque de texto cifrado 51 al final de un conjunto de etapas y acciones.

La primera acción es cargar los primeros datos 21 en un registro 23. Preferiblemente, y como se muestra en la forma de realización representada en la Figura 2, los primeros datos 21 de la primera iteración IT_1 son un vector de inicialización IV que puede generarse por cualquier generador aleatorio o pseudoaleatorio. La siguiente acción pretende ejecutar la ronda del método realizando las siguientes tres etapas principales.

La primera etapa de la ronda se utiliza para introducir los datos de entrada 20 en un algoritmo de cifrado por bloques (E) 35. Este algoritmo es un esquema de cifrado por bloques genérico que puede referirse a DES, IDEA, AES, o a cualquier otro algoritmo de cifrado por bloques. Los datos de entrada 20 comprenden datos cargados en el registro 23, es decir, los primeros datos 21 y los datos de contador (CDt_i) 22. Los datos de contador 22 se refieren a un valor proporcionado por un contador (CTR) 25. Este contador puede considerase como un actualizador (es decir, una unidad para actualización) ya que se actualiza, concretamente, aumenta (o disminuye), cada vez que se emprende una nueva iteración IT_n. Por consiguiente, cada ronda se inicia con nuevos datos de contador (CDt_1, CDt_2, CDt_3,… CDt_i, CDt_{i+1}). Por lo tanto, los datos de contador se usan como el denominado "una vez", concretamente como "número utilizado una vez". De hecho, cada valor proporcionado por el contador es único a través de todos los bloques procesados 11.

Durante la segunda etapa de la ronda se computa un criptograma 30 después de producirse a partir del algoritmo de cifrado por bloques 35 cifrando los datos de entrada 20 usando una sola clave criptográfica (K) 34. Este criptograma 30 comprende un primer segmento (S1) 31 y un segundo segmento 32. El último se define como una etiqueta de autenticación (T_n), en particular como la etiqueta local que autentica el primer segmento 31. Preferiblemente, la etiqueta local 32 se almacena en una memoria, ya sea hasta que se haya calculado la etiqueta global 52, o al menos temporalmente hasta que se realice un cálculo intermedio con la siguiente etiqueta local 32 de la siguiente ronda, en el caso en que la etiqueta local se determina progresivamente durante cada iteración.

La tercera etapa de la ronda se dedica a computar el bloque de texto cifrado 51 del bloque de texto simple actual 11. Para este fin, el bloque de texto cifrado 51 se determina realizando una primera operación 41 utilizando como operandos: el primer segmento 31, el bloque de texto simple actual 11 y la etiqueta local 32. Por lo tanto, el bloque de texto cifrado 51 es el resultado de una operación realizada sobre tres datos.

Dado que esta primera operación 41 usa tres operandos, por lo tanto, se realiza en dos etapas, concretamente, realizando una primera sub-operación $OP1_1$ que utiliza un primer operador y después realizando una segunda sub-operación $OP1_2$ que utiliza un segundo operador. Típicamente, las dos sub-operaciones $OP1_1$ y $OP1_2$ son operaciones lógicas (álgebra booleana), típicamente operaciones conmutativas. Aún preferiblemente, al menos uno de los operadores es una operación OR exclusiva (operación XOR).

Además, debido a la propiedad conmutativa de las sub-operaciones XOR, debe apreciarse que pueden procesarse los tres operandos en cualquier orden dentro de la primera operación 41. Como se muestra en el ejemplo de la Figura 1, el bloque de texto simple actual 11 se procesa con el primer segmento 31 mediante la primera sub-operación $OP1_1$ y después el resultado de esta primera sub-operación se procesa con la etiqueta local 32 mediante la segunda sub-operación $OP1_2$. Como resultado, la segunda sub-operación $OP1_2$ proporciona el bloque de texto cifrado 51 y termina la primera iteración IT_1.

En lugar de utilizar la operación Booleana, tal como la operación XOR, debe apreciarse que la segunda sub-operación $OP1_2$ puede basarse en cualquier otra función que se capaz de mezclar las distribuciones de los dos operandos utilizados como entradas, de tal manera que, cuando no se conoce uno de estos dos operandos, no es posible computar el otro operando para un bloque cifrado dado 51 (es decir, para un resultado dado). No obstante, obviamente la función debe permanecer reversible; de otro modo, el cifrado no sería factible. Por ejemplo, si se usa una adición modular como operación durante la fase de cifrado, entonces se usará una resta modular en lugar de la adición modular durante la fase de descifrado. Por lo tanto, las operaciones realizadas durante la fase de descifrado serán inversas de las realizadas durante la fase de cifrado.

Aunque la operación XOR sigue siendo la elección más natural, debe apreciarse que la adición modular o multiplicación modular también puede utilizarse para realizar tales operaciones durante la fase de cifrado si se utiliza una resta modular o división modular respectivamente para la fase de descifrado. Sin embargo, dichas operaciones modulares son menos adecuadas debido a que pueden liberar información (por ejemplo, el bit de arrastre en la adición modular) que puede usarse por otros ataques de canal lateral.

Siempre que los bloques de texto simple 11 no se hayan procesado completamente, el registro 23 se vuelve a cargar entonces con el bloque de texto cifrado 51 (es decir, con el bloque de texto cifrado actual, o último computado, 51 como se muestra en la Figura 2) y se aplica una función de actualización, por ejemplo, una función de aumento, a los datos de contador 22 antes de ejecutar una nueva ronda (es decir, antes de iniciar de nuevo la primera etapa de la ronda mencionada anteriormente). Por consiguiente, los primeros datos 21 de la segunda iteración IT_2 corresponderán al bloque de texto cifrado 51 transmitido a partir de la primera iteración IT_1, y los datos de contador CDt_2 de la segunda iteración corresponderán al resultado de la función de aumento aplicada a los datos de contador CDT_1 de la primera iteración IT_1. En la presente descripción, la expresión "función de aumento" se utilizará como una forma de realización preferida de la "función de actualización". Por lo tanto, debe entenderse que la "función de aumento" puede remplazarse por cualquier "función de actualización". De manera similar, el contador 25 mencionado en la presente descripción también puede considerarse como un "actualizador", como ya se ha mencionado anteriormente.

Cuando se han procesado todos los bloques de texto simple 11, el registro 23 se vuelve a cargar con el último bloque de texto cifrado computado 51 y se aplica la función de aumento del contador a los datos de contador (actuales) CDT_i, en cuanto a las iteraciones anteriores. Sin embargo, únicamente se realizan entonces las dos primeras etapas de la ronda a fin de obtener el criptograma 30 de los datos de entrada 20 (es decir, de c_i y CDt_{i+1}) que se procesa dentro del algoritmo de cifrado por bloques 35 utilizando la clave criptográfica 34. Por consiguiente, estas acciones no se refieren a una iteración completa (como las realizadas anteriormente), sino más bien deben considerarse como una iteración parcial. A partir de este último criptograma 30, la etiqueta local (T_{i+1}) 32 se utiliza entonces para computar la etiqueta global 52 mediante una segunda operación (OP2) 42 utilizando todas las etiquetas locales computadas, concretamente, todas las etiquetas locales computadas desde la primera iteración IT_1. Debe apreciarse que únicamente se utiliza la etiqueta local (segundo segmento 32) de esta iteración parcial. Por lo tanto, la determinación del primer segmento 31 permanece opcional, pero se hace implícita a través del cálculo del criptograma 30 que se procesa como un solo dato.

El resultado de la segunda operación (OP2) proporciona la etiqueta global 52. Este resultado se computa de una forma similar que al determinar el resultado de la primera operación (OP1), concretamente, mediante una pluralidad de sub-operaciones ($OP2_1$ a $OP2_i$). Preferiblemente, cada sub-operación utiliza como operandos la etiqueta local T_n de la iteración IT_n relacionada y el resultado de la sub-operación anterior determinado en la iteración anterior IT_{n-1}. La naturaleza (es decir, el tipo) de la segunda operación OP2 es similar o idéntica a la de la primera operación OP1. Lo mismo es cierto con respecto al orden en el que se utilizan los operandos para determinar la etiqueta global 52. La determinación de la etiqueta global 52 de todas las etiquetas locales corresponde al segundo pase del modo de operación.

El mensaje de texto cifrado 50 se determina al combinar entre sí (concatenar) todos los bloques de texto cifrado 51 en un orden específico que se reutilizará para recuperar el mensaje de texto simple 10 del mensaje de texto cifrado 50 durante un proceso inverso que se describirá en lo sucesivo. Por ejemplo, tal orden puede ser el mismo que aquel en el que se han procesado los bloques de texto simple 11. En una variante, puede asignarse un índice a cada uno de los bloques de texto cifrado 51 a fin de identificar el orden en el que se han procesado estos bloques.

Ventajosamente, realizando la primera operación OP1 usando tres operandos en lugar de dos, el bloque de texto cifrado 51 no es resultado de una sola operación, sino que es resultado de dos operaciones sucesivas,

concretamente, la primera sub-operación $OP1_1$, y la segunda sub-operación $OP1_2$. La primera sub-operación usa dos de los tres operandos como entradas, y la segunda sub-operación usa como entradas el tercer operando y el resultado de la primera sub-operación. Un atacante con el objetivo de realizar ataques DPA sobre tal proceso ejecutando el algoritmo de cifrado por bloques sobre muchas entradas (por ejemplo, 100.000 veces para observar 100.000 rastros de energía), no tendrá acceso al resultado intermedio determinado sobre la base de los primeros dos operandos. Más bien únicamente puede observar el resultado proporcionado después de la segunda sub-operación $OP1_2$. Por consiguiente, necesitará primero atacar la segunda sub-operación antes de poder analizar la primera sub-operación $OP1_1$. Sin embargo, la segunda sub-operación se basa en dos valores desconocidos; ambos distribuidos uniformemente y no correlacionados. Como resultado el DPA no funciona, o será al menos mucho más difícil de aplicar tal ataque en la materia objeto de la presente invención que en uno de los procesos de la técnica anterior.

De acuerdo con una forma de realización, el método comprende además una etapa preliminar para restaurar la longitud de bits común entre los primeros datos 21 (es decir, datos cargados dentro del registro 23) y los datos de contador 22. Esta etapa preliminar se realiza en el caso en que los primeros datos 21 y los datos de contador 22 no tienen la misma longitud de bits. Preferiblemente, esta etapa preliminar se adopta antes de introducir dichos datos de entrada 20 en el algoritmo de cifrado por bloques 35 o dentro del propio algoritmo 35. Preferiblemente, el primer segmento 31 y la etiqueta local 32 tienen la misma longitud de bits. Si no, el método puede comprender además una etapa adicional (etapa de relleno) para restaurar la misma longitud de bits entre estos dos segmentos. Aún preferiblemente, los primeros datos 21 (por ejemplo, IV), los datos de contador 22, el primer segmento 31 y el segundo segmento 32 (etiqueta local) tienen todos la misma longitud de bits, es decir, la longitud de medio bits de los datos de entrada 20.

De acuerdo con una forma de realización mostrada en la Figura 3, los primeros datos 21 pueden ser el resultado de una operación inicial (OP0) 43 utilizando los datos asociados (AD) de texto simple 24 como operando. Como se muestra como ejemplo en esta Figura, puede utilizarse un vector de inicialización IV como segundo operando de la operación inicial 43. Los datos asociados 24 típicamente pueden relacionarse con un encabezado del mensaje de texto simple 10. En este caso, debe entenderse que el encabezado (es decir, los datos asociados) no se procesa de la misma forma que el propio mensaje, sino permanece como un apéndice del mensaje. La naturaleza de la operación inicial OP0 es similar o idéntica a la de la primera operación OP1 o a la de la segunda operación OP2. Normalmente, se añaden los datos asociados al vector de inicialización.

Haciendo referencia a la Figura 4, se muestra la fase de descifrado del mensaje de texto cifrado 50 obtenido de acuerdo con la fase de cifrado mostrada en la Figura 2. Dado que el modo de operación para cifrar un mensaje de texto simple 10 (Figura 2) se refiere a un proceso de cifrado reversible, el método de descifrado mostrado en la Figura 4 es muy similar al método de cifrado de la Figura 2.

Más específicamente, la Figura 4 representa un método de descifrado por bloques que utiliza repetición de rondas para proporcionar, a partir de un mensaje de texto cifrado 50, un mensaje de texto simple 10 y una etiqueta global 52 para autenticar dicho mensaje de texto simple. El mensaje de texto cifrado 50 se convierte en una pluralidad de bloques de texto cifrado ordenados 51; teniendo todos ellos la misma longitud de bits (tal operación no se muestra en la Figura 4 pero es similar a la mostrada en la Figura 1). Más particularmente, esta longitud de bits es la misma que la que se ha definido para obtener los bloques de texto simple 11 a partir del mensaje de texto simple 10 que estuvo en el origen de este mensaje de texto cifrado 50 durante la fase de cifrado (Figura 2). Como se explicó con respecto a la fase de cifrado, si la hubiera, puede aplicarse también una etapa de relleno a los bloques terminales (primer o último bloque, dependiendo del bloque que se ha rellenado durante la fase de cifrado relacionada). Cada uno de los bloques de texto cifrado 51 se procesa sucesivamente de una forma ordenada como un bloque de texto cifrado actual durante una ronda. En particular, esta forma ordenada permite conservar las posiciones respectivas entre los bloques 11 dentro del mensaje de texto simple 10 y los bloques correspondientes 51 dentro del mensaje de texto cifrado 50.

Hablando en general, se adoptan las mismas acciones que las realizadas durante la fase de cifrado, concretamente, se cargan los primeros datos 21 en el registro 23, después se ejecuta la ronda realizando las siguientes etapas:

Una primera etapa para introducir los datos de entrada 20 en un algoritmo de descifrado por bloques 35, comprendiendo estos datos de entrada datos cargados en el registro 23 y los datos de contador 22.

Una segunda etapa para computar, dentro del algoritmo de descifrado por bloques 35, una

primera salida 30 al descifrar los datos de entrada 20 utilizando la única clave criptográfica 34 (es decir, la misma clave K que la utilizada para la fase de cifrado). Esta primera salida 30 comprende un primer segmento 31 y un segundo segmento 32, definiéndose el último como una etiqueta local de autenticación.

Una tercera etapa para computar un bloque de texto simple 11 realizando una primera operación (OP1) 41 utilizando como operandos: el primer segmento 31, el bloque de texto cifrado 51 actual y el segundo segmento 32 (es decir, la etiqueta local proporcionada por la iteración actual). Por consiguiente, el bloque de texto simple 11 es el resultado de una operación realizada sobre tres datos. Dado que esta primera operación 41 utiliza tres operandos, por lo tanto, esta se realiza en dos etapas como ya se ha explicado con respecto a la fase de cifrado.

Debe apreciarse que la operación (OP1) 41 realizada durante la fase de descifrado (Figura 4) es la inversa de la operación (OP1) 41 realizada durante la fase de cifrado (Figura 2). Además, ya que cada una de estas operaciones 41 comprende dos sub-operaciones sucesivas, concretamente, la primera $OP1_1$ y después $OP1_2$, debe entenderse que la primera sub-operación $OP1_1$ realizada durante la fase de descifrado corresponde a la inversa de la segunda sub-operación $OP1_2$ realizada en la fase de cifrado. De manera similar, la segunda sub-operación $OP1_2$ realizada durante la fase de descifrado corresponde a la inversa de la sub-operación $OP1_1$ realizada primero durante la fase de descifrado.

Siempre que todos los bloques de texto cifrado 51 no se hayan procesado completamente, se vuelve a cargar el registro durante una siguiente iteración con el bloque de texto cifrado actual 51 (es decir, con el último bloque de texto cifrado computado 51, como se muestra en la Figura 4) y se aplica una función de aumento, por el contador 25, sobre los datos de contador de la iteración anterior para obtener nuevos datos de contador 22. Estos nuevos datos de contador 22 se utilizarán cuando ejecute la nueva ronda, en la nueva iteración actual, al iniciar de nuevo la primera etapa del presente método de descifrado. En particular, la función de aumento es la misma función que la utilizada durante la fase de cifrado. Además, los primeros datos de contador (CDt_1) utilizados como el valor de inicio (por ejemplo, por la primera iteración IT_1') durante la fase de descifrado son los mismos que los utilizados como el valor de inicio (por la primera iteración IT_1) durante la fase de cifrado. Por esta razón, los primeros datos de contador (CDt_1) de la fase de cifrado pueden transmitirse al dispositivo de descifrado que se encuentra a cargo de descifrar el mensaje de texto cifrado 50. Típicamente, tales datos de contador (CDt_1) pueden transmitirse junto con el mensaje de texto cifrado 50, por ejemplo, en el encabezado (no cifrado) del mensaje de texto cifrado 50. Además, estos datos de contador (CDt_1) pueden también almacenarse dentro del dispositivo de cifrado, al menos temporalmente, en el caso en que estos datos deben volverse a enviar al dispositivo de descifrado por cualquier razón (por ejemplo, para el propósito de la resincronización entre el remitente y el receptor). Lo mismo es cierto con respecto a los primeros datos 21 (IV). De acuerdo con otro modo, los datos de contador iniciales (CDt_1) pueden también ajustarse a un valor predeterminado (por ejemplo, pueden ajustarse a 1) de manera que no necesiten comunicarse. De hecho, los mismos primeros datos 21 tienen que cargarse dentro del registro durante las primeras iteraciones IT_1 e IT_1'. Por consiguiente, los primeros datos 21 (IV) pueden también transmitirse de la misma forma que para los primeros datos de contador y pueden también almacenarse dentro del dispositivo de cifrado por la misma razón, o pueden ajustarse a un valor predeterminado sin la necesidad de comunicarse. En una variante, los primeros datos de contador (CDt_1), los primeros datos 21 (IV) y otros datos que pueden variar, tal como la clave criptográfica 34 (K) y/o la función de aumento del contador 25, pueden compartirse al menos una vez, entre el dispositivo de cifrado y el dispositivo de descifrado durante la fase de iniciación. Tal fase de inicio podría también reproducirse después, por ejemplo, para restablecer el sistema o actualizarlo.

El mensaje de texto simple 10 se determina al combinar (concatenar) los bloques de texto simple 11 obtenidos durante todas las rondas en el mismo orden, como se ha mencionado antes, a fin de recuperar el mensaje de texto simple que se utilizó originalmente durante la fase de cifrado para proporcionar el mensaje de texto cifrado 50.

Después, el registro 23 se vuelve a cargar con el bloque de texto cifrado 51 (por ejemplo, con el último bloque de texto cifrado 51 que se ha procesado durante la última iteración IT_1'), y se aplica la función de aumento una o más veces sobre los últimos datos de contador 22 antes de realizar solo las primeras dos etapas de la ronda (de manera similar a lo que se realizó durante la fase de cifrado cuando se refiere a la iteración parcial). A partir de esta última primera salida 30, la etiqueta local (T_{i+1}) 32 se utiliza entonces para computar la etiqueta global 52 (T_G) mediante una segunda operación (OP2) 42 que utiliza todas las etiquetas locales de autenticación computadas 32 como operandos. Como ya se ha explicado con respecto a la fase de cifrado, la

etiqueta global 52 resultante de la segunda operación OP2 se calcula de una forma similar que para determinar el resultado de la primera operación (OP1), concretamente, mediante una pluralidad de sub-operaciones ($OP2_1$ a $OP2_i$). Puede aplicarse la misma clase de operaciones y variantes que las menciones durante la fase de cifrado, durante la fase de descifrado.

Independientemente de la fase de cifrado o de descifrado, debe apreciarse que las sub-operaciones ($OP2_1$ a $OP2_i$) no se computan necesariamente una vez que se han procesado todos los bloques 11, 51. De hecho, estas sub-operaciones pueden computarse de manera progresiva, una por una durante cada iteración, al determinar un resultado intermedio. Este resultado intermedio puede almacenarse en una memoria hasta que se procese como un operando con la siguiente etiqueta local (segundo operando) proporcionada por la siguiente iteración (o iteración parcial al final del proceso). El mismo principio puede aplicarse para determinar el mensaje 10, 50 de los bloques 11, 51 respectivos.

Además, debido a la propiedad inversa del algoritmo 35, debe apreciarse que el algoritmo de cifrado por bloques mencionado en la fase de cifrado es el mismo que el algoritmo de descifrado por bloques de la fase de descifrado. Además, debe apreciarse que el resultado proporcionado por el algoritmo 35 se define como la denominada "primera salida" 30 en la fase de descifrado. Este término se ha elegido en lugar de la palabra criptograma, debido a que esta salida debe considerarse más bien como datos descifrados (aunque, hablando técnicamente, el algoritmo 35 no hace ninguna diferencia entre la denominada primera salida y el denominado criptograma).

De acuerdo con una forma de realización, el método de descifrado por bloques comprende además una etapa preliminar para restaurar una longitud de bits común entre los primeros datos 21 cargados en el registro 23 y los datos de contador 22. Esta etapa preliminar se usará en el caso de que los primeros datos 21 y los datos de contador 22 no tengan la misma longitud de bits. Tal etapa preliminar se adoptará antes de introducir los datos de entrada 20 en el algoritmo de cifrado por bloques 35.

Como ya se ha mencionado con respecto a la fase de cifrado, los primeros datos 21 pueden ser un vector de inicialización IV, en particular, un solo bloque que tiene la misma longitud de bits que la de los datos de contador 22.

De acuerdo con una forma de realización, y de forma similar a la que se muestra en la Figura 3, los primeros datos 21 pueden ser el resultado de una operación inicial OP0 que utiliza los datos asociados con el texto simple 24 como un operando. Preferiblemente, esta operación inicial OP0 y la primera y la segunda operaciones OP1, OP2 son operaciones OR exclusivas lógicas. Sin embargo, y como se ha mencionado anteriormente, pueden también utilizarse otras clases de operaciones.

Además, debe apreciarse que el mensaje de texto cifrado 50 del presente método de descifrado puede asociarse además con los datos de autenticación (en particular, la etiqueta global obtenida durante la fase de cifrado mediante un método de cifrado por bloques relacionado). En tal caso, el presente método de descifrado puede comprender además una etapa para verificar si estos datos de autenticación asociados son idénticos a la etiqueta global 52 determinada por el segundo pase del método de descifrado. Si la etiqueta global 52 no es idéntica a los datos de autenticación asociados con el mensaje de texto cifrado 50, ésta última se declarará como no auténtica.

Aunque la fase de cifrado y la fase de descifrado se han presentado respectivamente como un método de cifrado y como un método de descifrado, la presente invención puede también referirse a un solo método que comprende tanto la fase de cifrado como la fase de descifrado.

La invención también se refiere al un dispositivo criptográfico 60, 60' para implementar uno de los métodos o formas de realización que se divulgan en la presente descripción. Con referencia a las Figuras 1, 2 y 5, el primer dispositivo criptográfico 60 se dedica a la implementación de un método de cifrado por bloques usando la repetición de rondas para proporcionar, a partir de un mensaje de texto simple 10, un mensaje de texto cifrado 50 y una etiqueta global 52 para autenticar el mensaje de texto cifrado. Para este fin, el dispositivo criptográfico 60 comprende los siguientes componentes:

- un registro 23, típicamente en la forma de una memoria (por ejemplo, una celda de memoria) para recibir los primeros datos 21,
- un actualizador 25 para actualizar los datos reproducibles 22 de acuerdo con una función de actualización f; por ejemplo, puede referirse a un contador 25 para aumentar los datos de contador 22 de acuerdo con una función de aumento (f),

- una interfaz 61 para recibir al menos el mensaje de texto simple 10,
- una unidad de conversión 62 para convertir y preferiblemente almacenar el mensaje de texto simple 10 en una pluralidad de bloques de texto simple ordenados 11 (m_1, m_2,… m_i) que tienen todos la misma longitud de bits,
- una unidad de procesamiento 64 para procesar sucesivamente cada uno de los bloques de texto simple 11 de una forma ordenada como un bloque de texto simple actual durante una ronda, comprendiendo además la unidad de procesamiento una unidad criptográfica 65, que aloja un algoritmo de cifrado por bloques 35, y una memoria 66 que comprende una sola clave criptográfica 34 (K).

Cada ronda comprende:

- una primera etapa para introducir (por ejemplo, a través de la unidad de procesamiento 64 como se muestra en la Figura 5) datos de entrada 20 en el algoritmo de cifrado por bloques 35, comprendiendo estos datos de entrada 20 los datos 21 del registro 23 y los datos reproducibles (datos de contador 22),
- una segunda etapa para computar, dentro del algoritmo de cifrado por bloques 35, un criptograma 30 al cifrar los datos de entrada 20 utilizando una sola clave criptográfica 34; comprendiendo el criptograma 30 un primer segmento 31 y un segundo segmento 32, definiéndose el último como una etiqueta local de autentificación y almacenándose en una memoria, por ejemplo, en la memoria 66 que puede ubicarse en o sin la unidad de procesamiento 64,
- una tercera etapa para computar y almacenar, por ejemplo, en tal memoria 66, un bloque de texto cifrado 51 realizando una primera operación OP1 utilizando, como operandos, el primer segmento 31, el bloque de texto simple actual 11 y el segundo segmento 32 (por ejemplo, la etiqueta local).

Siempre que todos los bloques de texto simple 11 no se hayan procesado completamente, la unidad de procesamiento 64 se configura además para volver a cargar el registro 23 con el bloque de texto cifrado actual 51 (es decir, el último bloque de texto cifrado computado 51) y para aumentar los datos de contador 22 (es decir, para actualizar los datos reproducibles) antes de ejecutar una nueva ronda al iniciar de nuevo la primera etapa.

La unidad de conversión 62 se configura además para determinar el mensaje de texto cifrado 50 al concatenar los bloques de texto cifrados 51, (c_1, c_2,… c_i) de todas las rondas de una forma ordenada, en particular, en la forma ordenada mencionada anteriormente, a fin de conservar las posiciones respectivas entre los bloques 11 dentro del mensaje de texto simple 10 y los bloques correspondientes 51 dentro del mensaje de texto cifrado 50.

La unidad de procesamiento 64 se configura además para volver a cargar el registro 23 con el último bloque de texto cifrado computado 51 y para aplicar la función de aumento (f) a los datos de contador 22 a fin de aumentarlos antes de realizar solo las primeras dos etapas de la ronda. Después, la unidad de procesamiento 64 se configura para computar la etiqueta global 52 mediante una segunda operación OP2 al utilizar, por ejemplo, desde la memoria 66, todas las etiquetas locales de autenticación computadas (32) como operandos.

La interfaz 61 (o cualquier otra interfaz) se configura además para emitir el mensaje de texto cifrado 50 y la etiqueta global 52.

La operación de conversión realizada por la unidad de conversión 62 puede relacionarse para dividir el mensaje de texto simple 10 en bloques 11 que tienen la misma longitud de bits y, si la hubiera, para una operación de relleno aplicada al menos a uno de los dos bloques terminales 11 de este mensaje de texto simple una vez dividido.

El dispositivo criptográfico 60 puede incluir además un generador 67 (o un pseudogenerador) para generar un vector de inicialización (IV) que puede usarse para implementar el método de acuerdo con cualquier forma de realización divulgada en la presente descripción. Tal generador (u otro) puede también utilizarse para inicializar el contador 25 con un valor inicial, típicamente con el fin de restablecimiento. Además, debe apreciarse que el registro 23, el contador 25 y, si lo hubiera, el generador 67 pueden situarse dentro de la unidad de procesamiento 64.

Cuando se actúa como un primer dispositivo, en particular en el lado del emisor, el dispositivo de cifrado 60 típicamente se pretende que se use para cifrar los mensajes de texto simple 10 que se descifrarán después

por un dispositivo de descifrado 60', que actúa como un segundo dispositivo, en particular en el lado del receptor. Dado que la fase de descifrado corresponde al proceso inverso de la fase de cifrado, los componentes electrónicos que forman el dispositivo de cifrado son idénticos o similares a los del dispositivo de descifrado. Solo las funciones asociadas con algunos componentes del dispositivo de descifrado pueden diferir de las del dispositivo de cifrado. Por consiguiente, el dispositivo de descifrado de la presente invención también se describirá de aquí en adelante con referencia a los componentes de la Figura 5.

Por consiguiente, este segundo dispositivo se refiere a un dispositivo criptográfico 60' para implementar un método de descifrado por bloques que utiliza la repetición de rondas para proporcionar, a partir de un mensaje de texto cifrado 50, un mensaje de texto simple 10 y una etiqueta global 52 para autenticar el mensaje de texto simple. Este dispositivo criptográfico 60' comprende:

- un registro 23, típicamente en la forma de una memoria (o una celda de memoria) para recibir los primeros datos 21,
- un actualizador 25 para actualizar los datos reproducibles 22 de acuerdo con una función de actualización f; por ejemplo, puede referirse a un contador 25 para aumentar los datos de contador 22 de acuerdo con una función de aumento (f),
- una interfaz 61 para recibir al menos el mensaje de texto cifrado 50,
- una unidad de conversión 62 para convertir y preferiblemente almacenar el mensaje de texto cifrado 50 en una pluralidad de bloques de texto cifrado ordenados 51 que tienen todos la misma longitud de bits,
- una unidad de procesamiento 64 para procesar sucesivamente cada uno de los bloques de texto cifrado 51 de una forma ordenada como un bloque de texto cifrado actual durante una ronda, comprendiendo además la unidad de procesamiento 64 una unidad criptográfica 65, que aloja un algoritmo de descifrado por bloques 35, y una memoria 66 que comprende una sola clave criptográfica 34 (K).

Cada ronda comprende:

- Una primera etapa para introducir los datos de entrada 20 en el algoritmo de descifrado por bloques 35, comprendiendo estos datos de entrada datos 21 cargados en el registro 23 y datos reproducibles (datos de contador 22),
- una segunda etapa para computar, dentro del algoritmo de cifrado por bloques 35, una primera salida 30 al procesar los datos de entrada 20 utilizando una sola clave criptográfica 34, comprendiendo la primera salida 30 un primer segmento 31 y un segundo segmento 32, definiéndose este último como una etiqueta local de autenticación y estando almacenado (al menos temporalmente) en una memoria, por ejemplo, la memoria 66,
- una tercera etapa para computar y después almacenar (por ejemplo, en tal memoria 66), un bloque de texto simple 11 realizando una primera operación OP1 utilizando, como operandos, el primer segmento 31, el bloque de texto cifrado actual 51 y el segundo segmento 32 (es decir, la etiqueta local).

Siempre que todos los bloques de texto cifrado 51 no se hayan procesado completamente, la unidad de procesamiento 64 se configura además para volver a cargar el registro 23 con el bloque de texto cifrado actual (es decir, el último bloque de texto cifrado computado) y para actualizar los datos reproducibles (por ejemplo, para aumentar los datos de contador 22 por medio de la función de aumento del contador 25) antes de ejecutar una nueva ronda al iniciar de nuevo la primera etapa.

La unidad de conversión 62 se configura además para determinar el mensaje de texto simple 10 al concatenar los bloques de texto simple 11 de todas las rondas de una forma ordenada, en particular en la forma ordenada mencionada anteriormente a fin de conservar las posiciones respectivas entre los bloques 11, dentro del mensaje de texto simple 10, y los bloques correspondientes 51, dentro del mensaje de texto cifrado 50.

La unidad de procesamiento 64 se configura además para volver a cargar el registro 23 con el bloque de texto cifrado actual (es decir, el último bloque de texto cifrado que se ha procesado) y para aplicar la función de aumento (f) a los datos de contador 22 a fin de aumentarlos antes de realizar solo las primeras dos etapas de la ronda. Después, la unidad de procesamiento 64 se configura para computar la etiqueta global 52 mediante una segunda operación OP2 al utilizar, por ejemplo, desde la memoria 66, todas las etiquetas locales de autenticación computadas (32) como operandos.

Preferiblemente, la unidad de procesamiento 64 del dispositivo criptográfico 60' se configura además para verificar si la etiqueta global 52 es idéntica a los datos de autenticación proporcionados junto con el mensaje de texto cifrado 50. Tales datos de autenticación estarán típicamente en la etiqueta global 52 que se determino durante la fase de cifrado mediante el dispositivo criptográfico 60. En el caso de que exista (para un mismo mensaje 50) una diferencia entre las etiquetas globales 52 proporcionadas por cada uno del dispositivo criptográfico 60, 60', esto significa que el mensaje de texto simple 10 (o el mensaje de texto cifrado 50) no es auténtico. Por consiguiente, la unidad de procesamiento 64 puede adoptar una acción apropiada, por ejemplo, puede activar un mensaje de advertencia, interrumpir la liberación del mensaje de texto simple 10 y/o puede enviar información como datos de reporte.

Por supuesto, la interfaz 61 se configura además para transmitir el mensaje de texto simple 10 una vez que se ha restituido.
Debe apreciarse que el algoritmo de cifrado del dispositivo criptográfico 60 es el mismo que el algoritmo de descifrado del dispositivo criptográfico 60'. De hecho, tal algoritmo 35 puede usarse para cifrar, así como para descifrar.

Además, en vista de determinar la etiqueta global 52, la memoria 66 (u otra memoria) se utilizará para almacenar todas las etiquetas locales 32 o los resultados intermedios en el caso de que la etiqueta global se determine de manera progresiva durante cada iteración. En una forma de realización, el mensaje de texto cifrado 50 o el mensaje de texto simple 10 pueden determinarse también de la misma forma.

La memoria 66 del dispositivo criptográfico 60, 60' puede ser una memoria segura. En una forma de realización, los componentes del dispositivo criptográfico 60, 60' se encuentran comprendidos en una unidad monolítica, por lo que no sería posible acceder físicamente a los componentes, en particular, al menos a los componentes confidenciales, sin destruir la unidad monolítica.

La clave criptográfica 34 se usa tanto para fines de cifrado como de descifrado mediante el dispositivo criptográfico 60, 60'.

Aunque las formas de realización de la presente divulgación se han descrito con referencia a las formas de realización ejemplares específicas, será evidente que pueden hacerse diversas modificaciones y cambios a estas formas de realización sin apartarse del alcance más amplio de estas realizaciones. Por consiguiente, la memoria descriptiva y los dibujos deben considerarse como ilustrativos en lugar de un sentido restrictivo. Los dibujos adjuntos que forman parte de la misma, muestran a modo de ilustración, y no de limitación, las formas de realización específicas en las que puede ponerse en práctica la materia objeto. Las formas de realización ilustradas se describen con suficiente detalle para permitir que los expertos en la técnica pongan en práctica las enseñanzas divulgadas en el presente documento. Pueden utilizarse otras formas de realización y derivarse de las mismas, de tal manera que pueden hacerse sustituciones y cambios estructurales y lógicos sin apartarse del alcance de esta divulgación. Por lo tanto, esta Descripción Detallada no debe tomarse en un sentido limitante, y el alcance de las diversas formas de realización se define únicamente por las reivindicaciones adjuntas, junto con un rango completo de equivalentes sobre las que tales reivindicaciones tienen derecho.

Dichas formas de realización de la materia objeto de la invención pueden denominarse en el presente documento, individual y/o colectivamente, por el término "intervención" solo por conveniencia y sin pretender limitar voluntariamente el alcance de esta solicitud a cualquier concepto inventivo único si de hecho se divulga más de uno. Por lo tanto, aunque se han ilustrado y descrito en el presente documento formas de realización específicas, debe apreciarse que cualquier disposición calculada para alcanzar el mismo propósito puede sustituirse por las formas de realización específicas mostradas. Esta divulgación pretende incluir cualquiera y todas las adaptaciones o variaciones de las diversas realizaciones. Las combinaciones de las formas de realización anteriores, y otras formas de realización no descritas específicamente en el presente documento, serán evidentes para los expertos en la técnica tras la revisión de la descripción anterior.

REIVINDICACIONES

1. Método de cifrado por bloques que usa repetición de rondas para proporcionar, a partir de un mensaje de texto simple (10), un mensaje de texto cifrado (50) y una etiqueta global (52) para autenticar dicho mensaje de texto cifrado (50), convirtiéndose dicho mensaje de texto simple (10) en una pluralidad de bloques de texto simple ordenados (11) que tienen una longitud de bits predefinida, procesándose cada uno de dichos bloques de texto simple (11) sucesivamente de manera ordenada como un bloque de texto simple actual durante una ronda, comprendiendo dicho método las siguientes acciones:

- cargar los primeros datos (21) en un registro (23),
- ejecutar dicha ronda realizando las siguientes etapas:
- una primera etapa para introducir datos de entrada (20) en un algoritmo de cifrado por bloques (35), comprendiendo dichos datos de entrada (20) datos (21) de dicho registro (23) y datos reproducibles (22),
- una segunda etapa para calcular, dentro de dicho algoritmo de cifrado por bloques (35), un criptograma (30) cifrando los datos de entrada (20) utilizando una sola clave criptográfica (34), comprendiendo dicho criptograma (30) un primer segmento (31) y un segundo segmento (32), estando este último definido como una etiqueta local de autenticación,
- una tercera etapa para calcular un bloque de texto cifrado (51) realizando una primera operación (41) usando, como operandos, dicho primer segmento (31) y dicho bloque de texto simple actual (11),
- siempre que no se hayan procesado dichos bloques de texto simple (11) en una ronda, volver a cargar el registro (23) con dicho bloque de texto cifrado (51) y aplicar una función de actualización a dichos datos reproducibles (22) antes de ejecutar una nueva ronda comenzando de nuevo en dicha primera etapa,
- determinar dicho mensaje de texto cifrado (50) concatenando los bloques de texto cifrado (51) de todas las rondas de dicha manera ordenada, después
- volver a cargar el registro (23) con el último bloque de texto cifrado computado (51) y aplicar dicha función de actualización a dichos datos reproducibles (22) antes de realizar solo las dos primeras etapas de dicha ronda, después
- calcular dicha etiqueta global (52) mediante una segunda operación (42) utilizando todas las etiquetas locales de autenticación computadas (32) como operandos,

caracterizado por que la tercera etapa de dicha ronda utiliza además dicho segundo segmento (32) como un operando.

2. Método de cifrado por bloques de la reivindicación 1, que comprende además una etapa preliminar para restaurar una longitud de bits común entre los primeros datos (21) de dicho registro (23) y dichos datos reproducibles (22) si los primeros datos (21) y dichos datos reproducibles (22) no tienen la misma longitud de bit, adoptándose dicha etapa preliminar antes de introducir dichos datos de entrada (20) en el algoritmo de cifrado por bloques (35).

3. Método de cifrado por bloques de las reivindicaciones 1 o 2, en el que dichos primeros datos (21) son un vector de inicialización.

4. Método de cifrado por bloques de las reivindicaciones 1 o 2, en el que dichos primeros datos (21) son el resultado de una operación inicial (43) que utiliza datos asociados con texto simple (24) como un operando.

5. Método de cifrado por bloques de cualquiera de las reivindicaciones 1 a 4, en el que dicha operación inicial (43) y dichas primera (41) y segunda (42) operaciones son operaciones OR exclusivas lógicas.

6. Método de cifrado por bloques de cualquiera de las reivindicaciones 1 a 5, que comprende además una etapa para rellenar un bloque de texto simple terminal (11) con al menos un bit predeterminado en el caso en el que el mensaje de texto simple (10) tiene un número de bit que no es un múltiplo del número de bits de dichos bloques de texto simple (11).

7. Método de descifrado por bloques que usa repetición de rondas para proporcionar, a partir de un mensaje de texto cifrado (50), un mensaje de texto simple (10) y una etiqueta global (52) para autenticar dicho mensaje de texto simple, convirtiéndose dicho mensaje de texto cifrado (50) en una pluralidad de los bloques de texto cifrado (51) que tienen la misma longitud de bits, procesándose cada uno de dichos bloques de texto cifrado (51) sucesivamente de manera ordenada como un bloque de texto cifrado actual durante una ronda,

comprendiendo dicho método las siguientes acciones:

- cargar los primeros datos (21) en un registro (23),
- ejecutar dicha ronda realizando las siguientes etapas:
- una primera etapa para introducir datos de entrada (20) en un algoritmo de descifrado por bloques (35), comprendiendo dichos datos de entrada (20) datos (21) de dicho registro (23) y datos reproducibles (22),
- una segunda etapa para calcular, dentro de dicho algoritmo de descifrado por bloques (35), una primera salida (30) descifrando los datos de entrada (20) utilizando una clave criptográfica única (34), comprendiendo dicha primera salida un primer segmento (31) y un segundo segmento (32), definiéndose este último como una etiqueta local de autenticación,
- una tercera etapa para calcular un bloque de texto simple (11) realizando una primera operación (41) utilizando, como operandos, dicho primer segmento (31) y dicho bloque de texto cifrado actual (51),
- siempre que no se hayan procesado dichos bloques de texto cifrado (51) en una ronda, volver a cargar el registro (23) con dicho bloque de texto cifrado actual (51) y aplicar una función de actualización a dichos datos reproducibles (22) antes de ejecutar una nueva ronda comenzando de nuevo en dicha primera etapa,
- determinar dicho mensaje de texto simple (10) concatenando los bloques de texto simple (11) de todas las rondas de dicha manera ordenada, después
- volver a cargar el registro (23) con el último bloque de texto cifrado actual (51) y aplicar dicha función de actualización a dichos datos reproducibles (22) antes de realizar solo las dos primeras etapas de dicha ronda, después
- calcular dicha etiqueta global (52) mediante una segunda operación (42) utilizando todas las etiquetas locales de autenticación computadas (32) como operandos,

caracterizado por que la tercera etapa de dicha ronda comprende además dicho segundo segmento (32) como un operando.

8. Método de descifrado por bloques de la reivindicación 7, que comprende además una etapa preliminar para restaurar una longitud de bits común entre los primeros datos (21) de dicho registro (23) y dichos datos reproducibles (22) si dichos primeros datos (21) y dichos datos reproducibles (22) no tienen la misma longitud de bit, adoptándose dicha etapa preliminar antes de introducir dichos datos de entrada (20) en el algoritmo de cifrado por bloques (35).

9. Método de descifrado por bloques de las reivindicaciones 7 o 8, en el que dichos primeros datos (21) son un vector de inicialización.

10. Método de descifrado por bloques de las reivindicaciones 7 o 8, en el que dichos primeros datos (21) son el resultado de una operación inicial (43) que utiliza datos asociados con texto simple (24) como un operando.

11. Método de descifrado por bloques de cualquiera de las reivindicaciones 7 a 10, en el que dicha operación inicial (43) y dichas primera (41) y segunda (42) operaciones son operaciones OR exclusivas lógicas.

12. Método de descifrado por bloques de cualquiera de las reivindicaciones 7 a 11, en el que dicho mensaje de texto cifrado (50) está dotado además de datos de autenticación y dicho método comprende además una etapa para verificar si dichos datos de autenticación son idénticos a dicha etiqueta global (52), si no, el mensaje de texto cifrado (50) se declara como no auténtico.

13. Dispositivo criptográfico (60) para implementar un método de cifrado por bloques que usa repetición de rondas para proporcionar, a partir de un mensaje de texto simple (10), un mensaje de texto cifrado (50) y una etiqueta global (52) para autenticar dicho mensaje de texto cifrado (50), que comprende:

- un registro (23) para recibir los primeros datos,
- un actualizador (25) para actualizar datos reproducibles (22) de acuerdo con una función de actualización,
- una interfaz (61) para recibir al menos dicho mensaje de texto simple (10),
- una unidad de conversión (62) para convertir y almacenar dicho mensaje de texto simple (10) en una pluralidad de bloques de texto simple ordenados (11) que tienen la misma longitud de bits,
- una unidad de procesamiento (64) para procesar sucesivamente cada uno de dichos bloques de texto simple (11) de manera ordenada como un bloque de texto simple actual durante una ronda,

comprendiendo dicha unidad de procesamiento (64) una unidad de cifrado (65), que aloja un algoritmo de cifrado por bloques (35), y una memoria (66) que almacena una sola clave criptográfica (34),
- comprendiendo cada ronda:
- una primera etapa para introducir datos de entrada (20) en el algoritmo de cifrado por bloques (35), comprendiendo dichos datos de entrada (20) datos (21) de dicho registro (23) y datos reproducibles (22),
- una segunda etapa para calcular, dentro de dicho algoritmo de cifrado por bloques (35), un criptograma (30) cifrando los datos de entrada (20) utilizando dicha clave criptográfica única (34), comprendiendo dicho criptograma (30) un primer segmento (31) y un segundo segmento (32), estando este último definido como una etiqueta local de autenticación y almacenándose en dicha memoria (66),
- una tercera etapa para calcular y almacenar en dicha memoria (66) un bloque de texto cifrado (51) realizando una primera operación (41) usando, como operandos, dicho primer segmento (31) y dicho bloque de texto simple actual (11),
- estando dicha unidad de procesamiento (64) configurada además para volver a cargar el registro (23) con dicho bloque de texto cifrado (51) y para actualizar los datos reproducibles (22) antes de ejecutar una nueva ronda comenzando de nuevo dicha primera etapa, siempre que todos dichos bloques de texto simple (11) no se hayan procesado en una ronda,
- estando dicha unidad de conversión (62) configurada además para determinar dicho mensaje de texto cifrado (50) mediante la concatenación de los bloques de texto cifrado (51) de todas las rondas de dicha manera ordenada, y
- estando dicha unidad de procesamiento (64) configurada además para recargar el registro (23) con el último bloque de texto cifrado computado (51) y aplicar la función de actualización a dichos datos reproducibles (22) para actualizarlos antes de realizar solo las dos primeras etapas de dicha ronda, y después calcular dicha etiqueta global (52) mediante una segunda operación (42) utilizando, desde dicha memoria (66), todas las etiquetas locales de autenticación computadas (32) como operandos,
- estando dicha interfaz (61) configurada además para emitir dicho mensaje de texto cifrado (50) y dicha etiqueta global (52),

caracterizado por que dicha unidad de procesamiento (64) está configurada además para tomar también dicho segundo segmento (32) como un operando en la tercera etapa de dicha ronda.

14. Dispositivo criptográfico (60') para implementar un método de descifrado por bloques que usa repetición de rondas para proporcionar, a partir de un mensaje de texto cifrado (50), un mensaje de texto simple (10) y una etiqueta global (52) para autenticar dicho mensaje de texto simple, que comprende:

- un registro (23) para recibir los primeros datos (21),
- un actualizador (25) para actualizar datos reproducibles (22) de acuerdo con una función de actualización,
- una interfaz (61) para recibir al menos dicho mensaje de texto cifrado (50),
- una unidad de conversión (62) para convertir y almacenar dicho mensaje de texto cifrado (50) en una pluralidad de bloques de texto cifrado ordenados (51) que tienen la misma longitud de bits,
- una unidad de procesamiento (64) para procesar sucesivamente cada uno de dichos bloques de texto cifrado (51) de manera ordenada como un bloque de texto cifrado actual durante una ronda, comprendiendo dicha unidad de procesamiento (64) una unidad de descifrado (65), que aloja un algoritmo de descifrado por bloques (35), y una memoria (66) que comprende una sola clave criptográfica (34),
- comprendiendo cada ronda:
- una primera etapa para introducir datos de entrada (20) en el algoritmo de descifrado por bloques (35), comprendiendo dichos datos de entrada (20) datos (21) de dicho registro (23) y datos reproducibles (22),
- una segunda etapa para calcular, dentro de dicho algoritmo de descifrado por bloques (35), una primera salida (30) procesando los datos de entrada (20) utilizando dicha clave criptográfica única (34), comprendiendo dicha salida (30) un primer segmento (31) y un segundo segmento (32), estando este último definido como una etiqueta local de autenticación y almacenándose en dicha memoria (66),
- una tercera etapa para calcular y almacenar en dicha memoria (66) un bloque de texto simple (11) realizando una primera operación (41) usando, como operandos, dicho primer segmento (31) y dicho bloque de texto cifrado actual (51),
- estando dicha unidad de procesamiento (64) configurada además para volver a cargar el registro

(23) con dicho bloque de texto cifrado actual (51) y para actualizar los datos reproducibles (22) antes de ejecutar una nueva ronda comenzando de nuevo dicha primera etapa, siempre que todos dichos bloques de texto cifrado (51) no se hayan procesado en una ronda,

- estando dicha unidad de conversión (62) configurada además para determinar dicho mensaje de texto simple (10) mediante la concatenación de los bloques de texto simple (11) de todas las rondas de dicha manera ordenada, y después
- estando dicha unidad de procesamiento (64) configurada además para recargar el registro (23) con el bloque de texto cifrado actual (51) y aplicar la función de actualización a dichos datos reproducibles (22) para actualizarlos antes de realizar solo las dos primeras etapas de dicha ronda, y después calcular dicha etiqueta global (52) mediante una segunda operación (42) utilizando, desde dicha memoria (66), todas las etiquetas locales de autenticación computadas (32) como operandos,
- estando dicha interfaz (61) configurada además para emitir dicho mensaje de texto simple (10),

caracterizado por que dicha unidad de procesamiento (64) está configurada además para tomar también dicho segundo segmento (32) como un operando en la tercera etapa de dicha ronda.

15. Dispositivo criptográfico (60') de la reivindicación 14, en el que dicha unidad de procesamiento (64) está configurada además para verificar si dicha etiqueta global (52) es idéntica a los datos de autenticación proporcionados junto con el mensaje de texto cifrado (50), y, si no es así, emprender una acción apropiada.

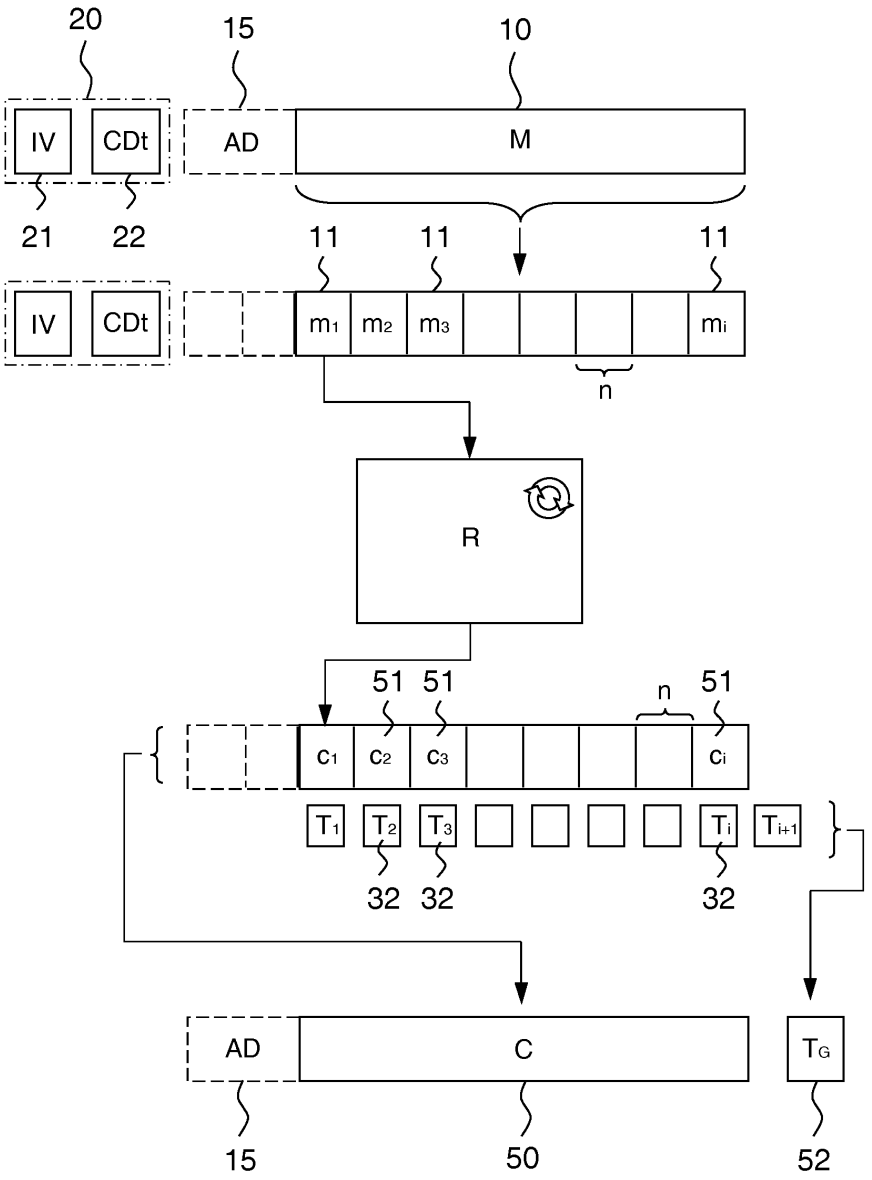
20
15
10
IV
CDt
AD
M
21
22
11
11
11
IV
CDt
m1
m2
m3
mi
n
R
51
51
n
51
c1
c2
c3
ci
T1
T2
T3
Ti
Ti+1
32
32
32
AD
C
TG
15
50
52

Fig. 1

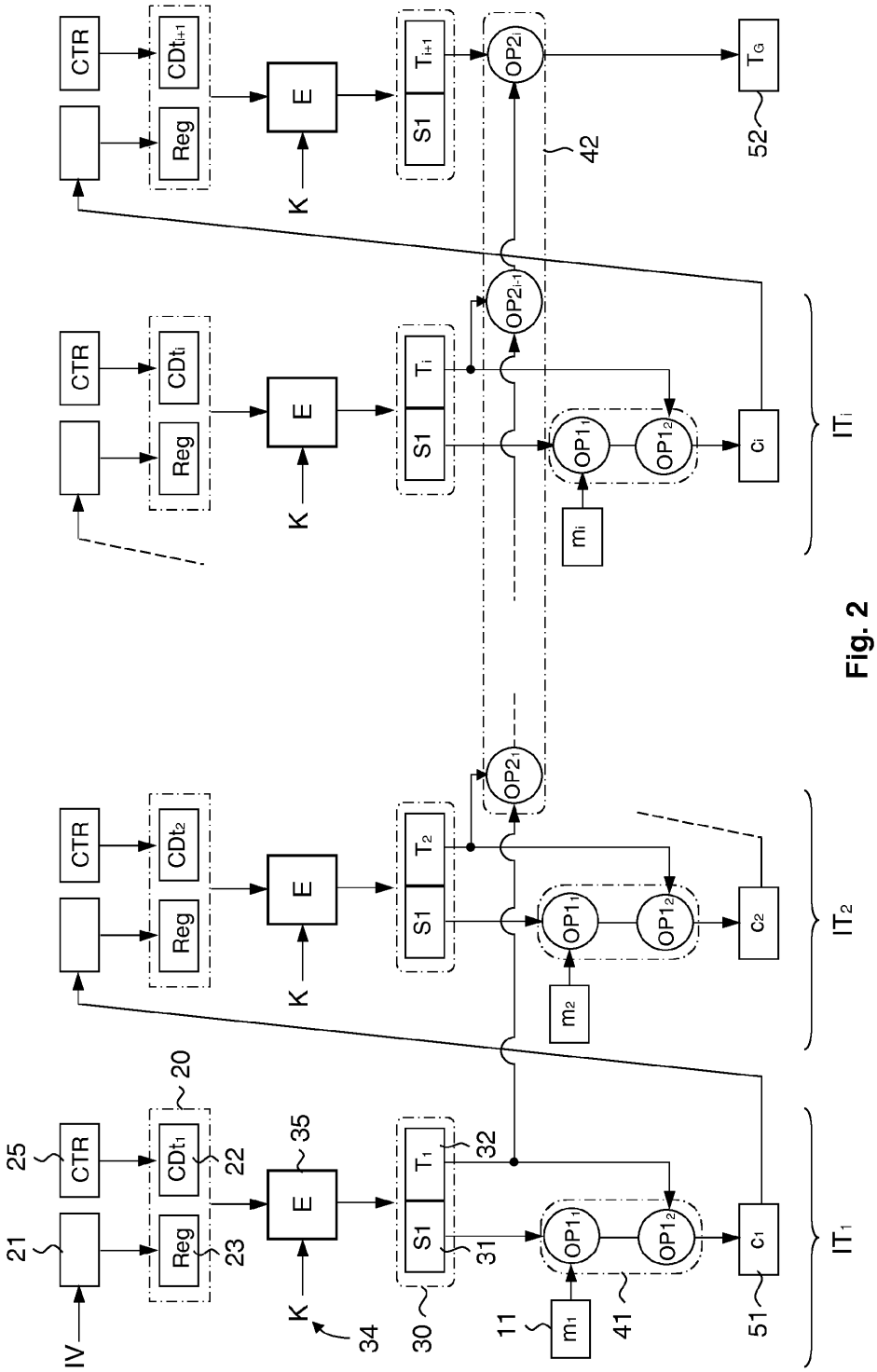
IV
21
25
CTR
Reg
CDt1
20
22
23
E
35
K
34
S1
T1
30
32
31
OP11
OP12
11
m1
41
c1
51
IT1
CDt2
T2
OP21
m2
c2
IT2
CDti
Ti
OP2i-1
mi
ci
ITi
CDti+1
Ti+1
OP2i
42
TG
52

Fig. 2

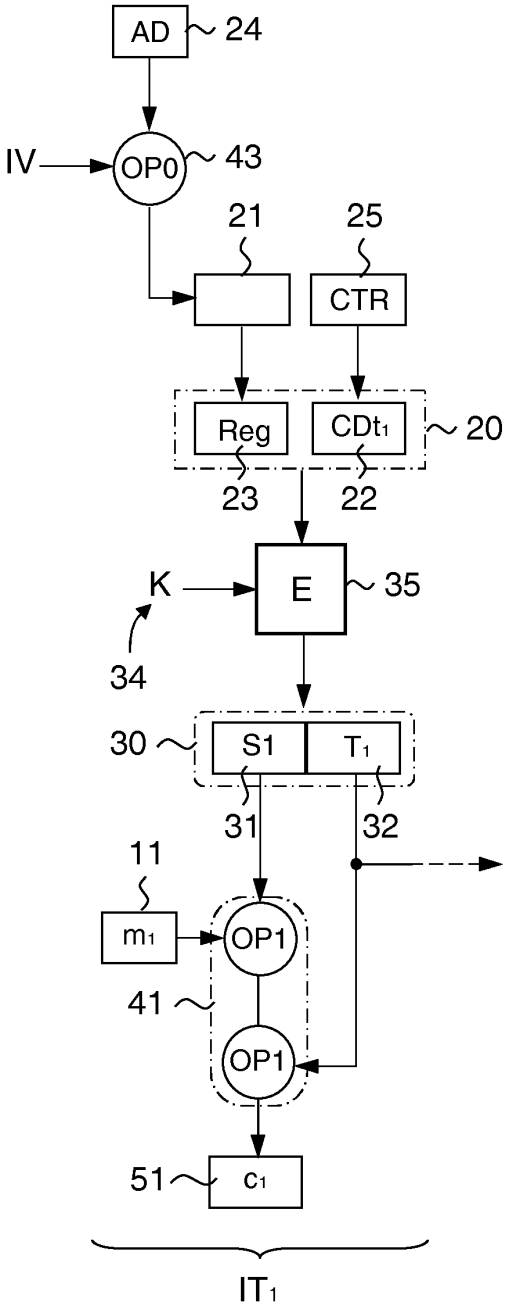
AD
24
IV
OP0
43
21
25
CTR
Reg
CDt1
20
23
22
K
E
35
34
30
S1
T1
31
32
11
m1
OP1
41
OP1
51
c1
IT1

Fig. 3

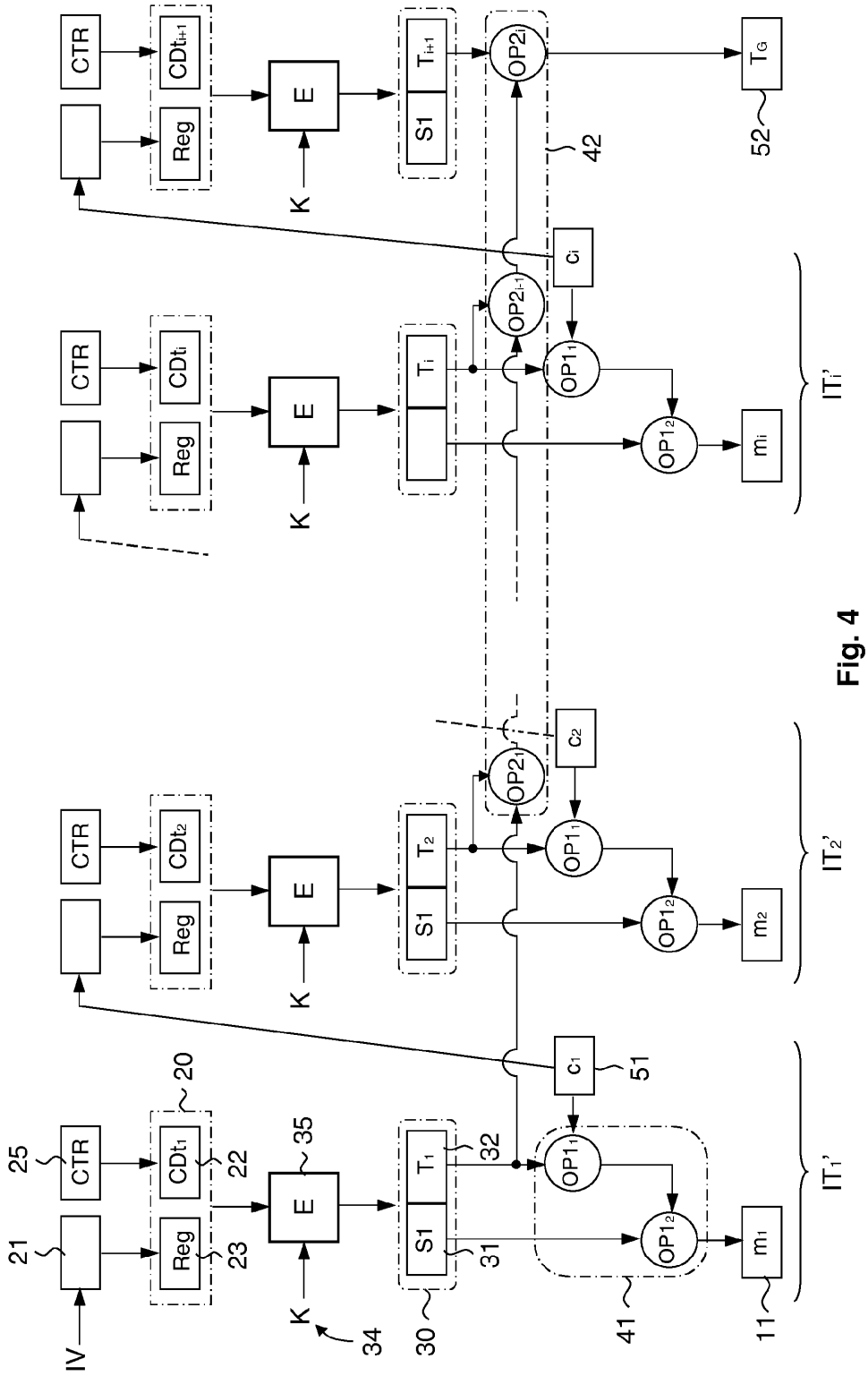
IV
21
25
CTR
Reg
CDt1
20
22
23
E
35
K
34
S1
T1
30
31
32
OP11
OP12
m1
11
41
c1
51
IT1'
OP21'
c2
T2
m2
IT2'
OP2i-1
ci
Ti
mi
ITi'
OP2i
Ti+1
42
TG
52

Fig. 4

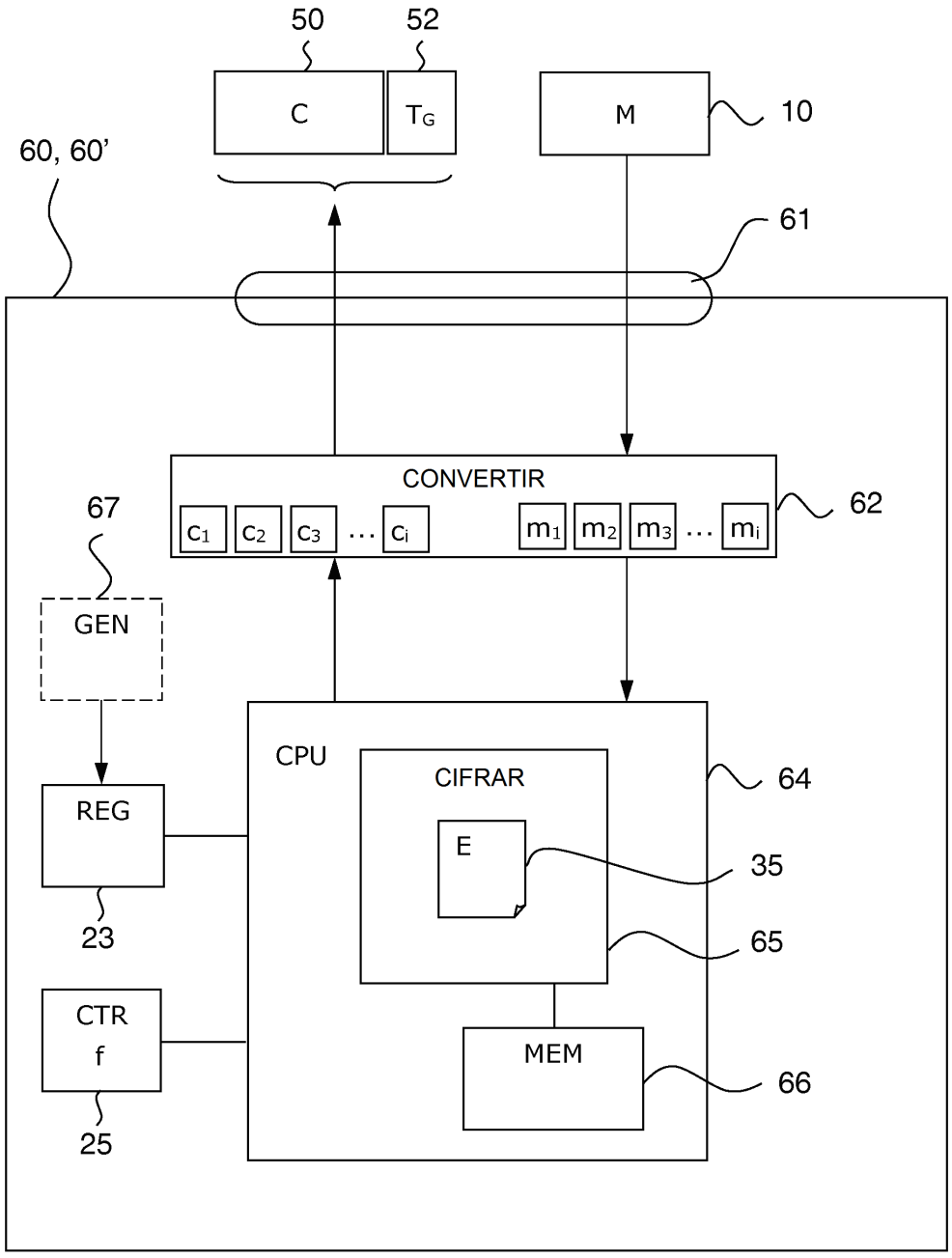
50
52
C
T_G
M
10
60, 60'
61
CONVERTIR
62
67
c_1 c_2 c_3 ... c_i
m_1 m_2 m_3 ... m_i
GEN
CPU
CIFRAR
64
REG
E
35
23
65
CTR
f
MEM
66
25

Fig. 5