(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 774 041**

(51) Int. CI.:

H04L 29/06 (2006.01)
G06F 21/10 (2013.01)
H04N 21/433 (2011.01)
H04N 21/4405 (2011.01)
H04N 5/913 (2006.01)
H04N 21/4408 (2011.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **10.07.2015 PCT/FR2015/051928**

(87) Fecha y número de publicación internacional: **21.01.2016 WO16009140**

(96) Fecha de presentación y número de la solicitud europea: **10.07.2015 E 15759510 (9)**

(97) Fecha y número de publicación de la concesión europea: **01.01.2020 EP 3170296**

(54) Título: **Método para acceder a un contenido multimedia protegido por un terminal**

(30) Prioridad:

16.07.2014 FR 1456806

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
16.07.2020

(73) Titular/es:

VIACCESS (100.0%)
Les Collines de l'Arche Tour Opéra C 76, Route de la Demi-Lune
92057 Paris La Défense Cedex, FR

(72) Inventor/es:

DOGUI, AMINE

(74) Agente/Representante:

LEHMANN NOVO, María Isabel

DESCRIPCIÓN

Método para acceder a un contenido multimedia protegido por un terminal

La invención se refiere a un método para acceder a un contenido multimedia protegido, mediante un terminal que comprende un descifrador, un decodificador y una memoria compartida. La invención también se refiere a un terminal y un soporte de registro de información para este método de acceso a un contenido multimedia protegido.

El método considerado puede ponerse en práctica después de la obtención del contenido multimedia protegido de cualquier servicio de suministro en línea para contenido multimedia protegido, en cualquier sistema de suministro en línea para contenido multimedia protegido en donde un encabezado de red proporciona protección de los contenidos y su transmisión a una pluralidad de terminales.

A continuación, un usuario del sistema utiliza un terminal para acceder a un contenido protegido para reproducirlo. Acceder a contenido multimedia protegido significa en el presente documento:

- cargarlo en memoria, para, a continuación
- levantar la protección, sobre la marcha mientras la recibe, o en un soporte de registro en donde se ha registrado previamente, a continuación
- decodificarlo, y a continuación
- transmitirlo a un dispositivo multimedia capaz de reproducirlo, registrarlo o hacer cualquier otro uso ofrecido por el servicio para proporcionar contenido multimedia protegido.

En el presente documento, "levantar la protección sobre la marcha" significa el hecho de que los fragmentos de contenido multimedia se procesan a medida que se reciben, sin esperar el contenido multimedia completo, es decir, el conjunto de sus fragmentos, haya sido completamente recibido.

Para este propósito, el terminal incluye un descifrador, un decodificador y una memoria compartida por los dispositivos precedentes.

Los contenidos proporcionados son:

- contenidos audiovisuales, por ejemplo, programas de televisión,
- únicamente contenido de audio, tal como un programa de radio, o
- más en general, cualquier contenido digital que contenga vídeo y/o audio, tal como una aplicación informática, un juego, una presentación de diapositivas, una imagen o cualquier conjunto de datos.

Entre estos contenidos, se considerará más en particular en la cadena de los contenidos denominados temporales. Un contenido multimedia temporal es un contenido multimedia cuya reproducción es una sucesión temporal de sonidos, en el caso de contenido de audio temporal, o de imágenes, en el caso de contenido de vídeo temporal, o de sonidos y de imágenes temporalmente sincronizadas entre sí en el caso de un contenido audiovisual multimedia. El contenido multimedia temporal también puede incluir componentes temporales interactivos sincronizados temporalmente con sonidos o imágenes.

Para ser proporcionado, dicho contenido se codifica primero, es decir, comprimido, de modo que su transmisión requiera menos ancho de banda.

Para este propósito, el componente de vídeo del contenido se codifica de conformidad con un formato de vídeo, tal como MPEG-2. El lector interesado podrá encontrar una presentación completa de este formato en el documento publicado por la Organización Internacional de Normalización bajo la referencia ISO/IEC 13818-2: 2013 y el título "Tecnología de la información - Codificación genérica de imágenes en movimiento y su sonido asociado - Parte 2: Datos de vídeo". De manera alternativa, se pueden utilizar muchos otros formatos, tal como MPEG-4 ASP, MPEG-4 Parte 2, MPEG-4 AVC (o Parte 10), HEVC (Codificación de Vídeo de Alta Eficiencia) o WMV (Windows Media Vídeo), y están basados en los mismos principios. Por lo tanto, todo lo siguiente también se aplica a estos otros formatos de vídeo que se basan en el mismo principio que la codificación MPEG-2.

La codificación MPEG-2 utiliza métodos generales de compresión de datos. Para las imágenes fijas, utiliza en particular la redundancia espacial interna de una imagen, la correlación entre los puntos próximos y la menor sensibilidad del ojo a los detalles. Para imágenes animadas, utiliza la fuerte redundancia temporal entre imágenes sucesivas. La utilización de esta última hace posible codificar algunas imágenes del contenido, aquí denominadas deducidas, con referencia a otras, denominadas fuentes, por ejemplo, mediante predicción o interpolación, de modo

que su decodificación solamente sea posible después de la de dichas imágenes fuentes. Otras imágenes, aquí denominadas iniciales, se codifican sin referencia a dichas imágenes fuente, es decir, cada una contiene, cuando se codifican, toda la información necesaria para su decodificación y, por lo tanto, se pueden decodificar completamente de manera independiente de las demás imágenes. Las imágenes iniciales son, por lo tanto, el punto de entrada obligatorio al acceder al contenido. Por lo tanto, el contenido codificado resultante no incluye los datos necesarios para decodificar cada una de las imágenes independientemente de las demás, sino que consiste en "secuencias" de conformidad con la terminología MPEG-2. Una secuencia comprime al menos un "grupo de imágenes" (o GOP, por Group Of Pictures, en MPEG-2).

Un grupo de imágenes es una cadena de imágenes consecutivas en las que cada imagen es:

- ya sea inicial y fuente para al menos una imagen deducida contenida en la misma secuencia de imágenes consecutivas,
- deducida y tal que cada una de las imágenes de origen necesarias para su decodificación pertenezcan a la misma cadena de imágenes consecutivas.

Un grupo de imágenes no contiene una cadena de imágenes consecutivas más pequeñas y que posean las mismas propiedades que las anteriores. El grupo de imágenes es, por lo tanto, la parte más pequeña de contenido a la que se puede acceder sin tener que decodificar previamente otra parte de este contenido.

Una secuencia está delimitada por un "encabezado" y un "final", cada uno identificado por un primer código específico. El encabezado incluye parámetros que caracterizan las propiedades esperadas de las imágenes decodificadas, tal como tamaños horizontales y verticales, relación, frecuencia. El estándar recomienda repetir el encabezado entre los grupos de imágenes en la secuencia, de modo que sus presencias sucesivas estén separadas aproximadamente en algunos segundos en el contenido codificado.

A modo de ejemplo, un grupo de imágenes normalmente incluye más de 5 a 10 imágenes y, por lo general, menos de 12 o 20 o 50 imágenes. A modo de ejemplo, en un sistema a 25 imágenes por segundo, un grupo de imágenes, por lo general representa un tiempo de reproducción mayor a 0,1 o 0,4 segundos y, por lo general, menor a 0,5 o 1 o 10 segundos.

Un contenido multimedia temporal puede comprender varios componentes de vídeo. En este caso, cada uno de estos componentes está codificado tal como se describió con anterioridad.

La componente de audio del contenido también se codifica de conformidad con un formato de audio tal como MPEG-2 Audio. El lector interesado puede encontrar una presentación completa de este formato en el documento publicado por la Organización Internacional de Normalización bajo la referencia ISO/IEC 13818-3: 1998 y el título "Tecnología de la información - Codificación genérica de imágenes en movimiento e información de audio asociada - Parte 3: Sonido". De manera alternativa, se pueden utilizar muchos otros formatos, tal como MPEG-1 Layer III, mejor conocido como MP3, AAC (codificación de audio avanzada), Vorbis o WMA (Windows Media Audio), y se basan en los mismos principios. Por lo tanto, todo lo que sigue también se aplica a estos otros formatos de audio que se basan en los mismos principios que la codificación de MPEG-2 Audio.

La codificación de MPEG-2 Audio obedece los mismos principios descritos con anterioridad para el contenido de vídeo temporal. El contenido codificado resultante está, por lo tanto, compuesto análogamente de "tramas". Una trama es el análogo, en audio, de un grupo de imágenes en vídeo. Por lo tanto, la trama es en particular la parte más pequeña del contenido de audio al que se puede acceder sin tener que decodificar otra parte de este contenido de audio. La trama también contiene el conjunto de las informaciones útiles en su decodificación.

Una trama suele incluir más de 100 o 200 muestras, cada una de las cuales codifica un sonido y, por lo general, menos de 2000 o 5000 muestras. En condiciones normales, cuando se reproduce en un dispositivo multimedia, una trama dura más de 10 ms o 20 ms y, en general, menos de 80 ms o 100 ms. A modo de ejemplo, una trama tiene 384 o 1152 muestras, cada una de las cuales codifica un sonido. Dependiendo de la frecuencia de muestreo de la señal, esta trama representa un tiempo de reproducción de 8 a 12, o de 24 a 36 milisegundos.

Un contenido multimedia temporal puede incluir varios componentes de audio. En este caso, cada uno de estos componentes está codificado tal como se describió con anterioridad.

Los componentes codificados del contenido, también calificados como flujos de datos elementales, se multiplexan a continuación, es decir, en particular, se sincronizan en el tiempo y a continuación se combinan en un solo tren o flujo de datos.

Dicho contenido, especialmente cuando está sujeto a derechos tales como derechos de autor o derechos relacionados, se proporciona protegido por un sistema de protección de contenido multimedia. Este sistema garantiza el cumplimiento de las condiciones de acceso al contenido que surgen de estos derechos.

Por lo general, se suministra cifrado como protección por un sistema de gestión de derechos digitales, o DRM, por Digital Rights Management, en inglés. Este cifrado se suele realizar por medio de una clave de cifrado, mediante un algoritmo simétrico. Se aplica al flujo resultante de la multiplexación o, antes de la multiplexación, a los componentes del contenido codificado.

Un sistema DRM es de hecho un sistema de protección de contenido multimedia. La terminología del campo de los sistemas de gestión de derechos digitales se utiliza así en el resto de este documento. El lector interesado puede, por ejemplo, encontrar una presentación más completa en el documento: DRM Architecture, versión Borrador 2.0, OMA-DRM-ARCH-V2_0-20040518-D, Open Mobile Alliance, 18 de mayo de 2004.

En dicho sistema de gestión de derechos digitales, la obtención de una licencia permite que un terminal acceda al contenido multimedia protegido.

De estructura bien conocida, dicha licencia incluye al menos una clave, denominada de contenido, necesaria para el descifrado de contenido multimedia protegido por un algoritmo de cifrado simétrico.

La clave de contenido se suele insertar en la licencia bajo la forma de un criptograma obtenido mediante el cifrado de la clave de contenido con una clave de cifrado, denominada "de terminal", específica del terminal o por el mismo conocida.

Para acceder al contenido, el terminal extrae de la licencia la clave de contenido, descifrando su criptograma utilizando su clave de terminal.

A continuación, el descifrador del terminal desaleatoriza, es decir descifra, el contenido mediante la clave de contenido así extraída de la licencia, levantando así la protección. El descifrador genera así un contenido multimedia sin cifrar que comprende al menos una cadena temporal de secuencias de vídeo o grupos de imágenes, o tramas de audio. Este contenido multimedia puede reproducirse mediante un dispositivo multimedia conectado al terminal. En este caso, por el término "no cifrado", se refiere al hecho de que el contenido multimedia ya no necesita ser descifrado para ser reproducido, por un dispositivo multimedia, de una manera directamente perceptible e inteligible por un ser humano. El término "dispositivo multimedia" también indica cualquier dispositivo capaz de reproducir el contenido multimedia no cifrado, tal como por ejemplo un televisor o un reproductor multimedia.

A continuación, el descifrador transfiere el contenido no cifrado en la memoria compartida.

Más adelante, el decodificador del terminal efectúa la lectura del contenido no cifrado de la memoria compartida y lo decodifica.

Por último, el terminal transmite el contenido así descodificado a un dispositivo multimedia.

Más concretamente, en el caso de un contenido multimedia temporal, la recepción, el procesamiento y a continuación la transmisión a un dispositivo multimedia, por el terminal, del contenido, tal como se describió con anterioridad, se realiza por fragmentos. Un fragmento es una parte restringida del flujo multimedia no cifrado, cuya reproducción es más corta que la del flujo multimedia completo. Por lo tanto, un fragmento incluye una parte restringida de cada componente de vídeo y de audio del flujo multimedia no cifrado, cuya reproducción tiene la misma duración menor que la reproducción del flujo multimedia completo. Estas partes restringidas de los componentes se sincronizan en el flujo para reproducirse de manera simultánea. Por lo tanto, un fragmento comprende la parte restringida de la cadena temporal de secuencias de vídeo o de grupos de imágenes, o de tramas de audio que llevan a cabo la codificación de esta parte restringida del componente del flujo multimedia no cifrado. Esta parte restringida por lo general consiste en una pluralidad de secuencias de vídeo o grupos de imágenes, o de tramas de audio sucesivas.

En el modo de acceso al contenido por el terminal tal como se describió con anterioridad, el contenido descifrado se transfiere a la memoria compartida del terminal por su descifrador, para a continuación ser objeto de lectura por su decodificador. Por lo tanto, el contenido no cifrado está presente en esta memoria, al menos entre el instante en el que se deposita por el descifrador y cuando es objeto de lectura por el decodificador. Entonces es fácil, en un entorno tan abierto, leer el contenido descifrado, y así obtener un acceso ilegítimo al contenido. Por lo tanto, es necesaria una protección adicional del contenido durante su presencia en la memoria compartida.

Se conocen soluciones criptográficas para este problema, que se basan en la utilización compartida de claves por el descifrador y el decodificador. En estas soluciones, el dispositivo cifrador cifra el contenido descifrado, antes de transferirlo a la memoria compartida. A continuación, el decodificador lee el contenido así cifrado en la memoria compartida, luego lo descifra y, antes de que posiblemente sea registrado en una memoria del terminal, lo decodifica y lo transmite a un dispositivo multimedia capaz de reproducirlo. Estas soluciones criptográficas son seguras, pero tienen el inconveniente de ser complejas y requieren recursos informáticos importantes para llevar a cabo los cálculos criptográficos implicados.

Se conocen otras soluciones que comparten este inconveniente, que consisten, por ejemplo, en la traducción de contenido a otros formatos de contenido.

También se conocen soluciones que consisten en la puesta en práctica de mecanismos de seguridad tales como la verificación, por el módulo DRM del terminal, de la aplicación de destino del contenido no cifrado. Estas soluciones tienen la desventaja de que esta puesta en práctica es poco segura en entornos abiertos tal como terminales.

La patente EP2268020 protege una solución alternativa que consiste en una pequeña alteración del contenido descifrado. Esta solución se describe como un requisito para lograr un compromiso entre los recursos de cálculo requeridos por su puesta en práctica y su seguridad. Para presentar una ganancia importante en términos de recursos de cálculo requeridos, por lo tanto, debe ir acompañada de una pérdida notable en términos de seguridad.

La presente invención tiene como objetivo ofrecer una solución alternativa más segura que la de la aplicación EP2268020 y al menos tan económica en tiempo de cálculo.

La técnica anterior también se da a conocer por el documento US6747580B1. El documento EP1947854 se refiere al problema según el cual la copia de contenido no autorizado es posible cuando el contenido se descifra en un momento dado. La solución propuesta es la siguiente: "envenenar" los datos cifrados para que un decodificador estándar ya no pueda leerlos, modificar algunos bits predeterminados del contenido, dejándolos inutilizables a menos que el dispositivo del usuario pueda realizar el cálculo inverso en el momento de la lectura; sustituir los valores de bits con bits predeterminados, o simplemente cambiarlos. La modificación de bits en áreas particulares de los datos hace que los datos sean inútiles.

La invención se refiere así a un método para acceder a un contenido multimedia protegido de conformidad con la reivindicación 1.

Las formas de realización de este método para acceder a un contenido multimedia protegido pueden incluir características de las reivindicaciones dependientes.

Estas formas de realización de este método para acceder a un contenido multimedia protegido también tienen las siguientes ventajas:

- El uso de una primera tabla de correspondencia hace posible poner en práctica una selección de la cadena de N bits sustituidos en la primera tabla en lugar de su cálculo en función del valor de la cadena de M bits originales. Ello presenta una ventaja adicional en términos de seguridad, al dificultar el criptoanálisis del proceso y, más concretamente, al garantizar que la observación externa, por ejemplo, del tiempo de cálculo o del consumo eléctrico, del sistema durante la puesta en práctica del método, con respecto a los valores de las cadenas de bits originales y sustituidos, no proporciona información sobre la naturaleza de la sustitución de bits realizada para obtener el fragmento modificado. Además, la puesta en práctica de una operación de entrada-salida en la primera tabla de correspondencia en lugar de una operación de cálculo, también tiene la ventaja de reducir de manera importante el tiempo de cálculo.

- El uso de una segunda tabla de correspondencia permite poner en práctica una selección precedida por la búsqueda de la cadena de N bits sustituidos en la segunda tabla en lugar de una búsqueda en la primera tabla de la cadena de M bits originales con la cual está asociada la cadena de N bits sustituidos del fragmento modificado. Ello presenta la ventaja de reducir la complejidad de la búsqueda efectuada y, por lo tanto, el tiempo de cálculo.

- La clasificación de las líneas de la segunda tabla de correspondencia en orden de valores de M bits de índices dados, permite que estos M bits se utilicen como índice de la segunda tabla de correspondencia, con el fin de seleccionar directamente, en la segunda tabla de correspondencia, la cadena de M bits originales asociada con la cadena de N bits sustituidos del fragmento modificado. Esta puesta en práctica de una operación de entrada-salida en la segunda tabla de correspondencia en lugar de una operación de búsqueda, tiene la ventaja anterior de reducir de manera importante el tiempo de cálculo.

- El uso de varias primeras tablas de correspondencia distintas permite aumentar la seguridad del método al dificultar su criptoanálisis. Más concretamente, ello permite variar las sustituciones de bits realizadas para obtener un fragmento modificado y reestablecer el fragmento no cifrado.

- El uso de las primeras tablas de correspondencia que difieren entre sí por el valor de M o por el valor de N, hace posible aumentar la seguridad del método al dificultar su criptoanálisis. Más concretamente, esto hace posible variar las longitudes de las cadenas de bits originales o de las cadenas de bits sustituidas.

- La extracción aleatoria o pseudoaleatoria de un número permite aumentar la seguridad del método al dificultar su criptoanálisis. Más concretamente, esto hace posible variar de manera aleatoria la sucesión de las sustituciones de bits llevadas a cabo para obtener un fragmento modificado y reestablecer el fragmento no cifrado.

- La toma igual a una potencia estrictamente positiva de dos del número de las primeras tablas de correspondencia distintas o de los bytes del número extraído aleatoria o pseudo-aleatoriamente hace posible reducir el tiempo de cálculo. Más concretamente, ello simplifica la selección en la etapa c) y la determinación en la etapa f), en donde tienen lugar las divisiones de módulo de estos números. De hecho, estas divisiones vuelven a producir desviaciones de bits si estos números son potencia de dos.

La invención también se refiere a un soporte de registro de información que comprende instrucciones para la puesta en práctica del método anterior de acceso a un contenido multimedia protegido, cuando estas instrucciones son ejecutadas por un ordenador electrónico.

La invención también se refiere a un terminal que comprende un descifrador, un decodificador y una memoria compartida para poner en práctica el método anterior de acceder a un contenido multimedia protegido.

La invención se entenderá mejor al leer la descripción que sigue, proporcionada únicamente a modo de ejemplo no limitativo, y realizada con referencia a los dibujos en los que:

- La Figura 1 es una representación esquemática de un terminal que comprende un descifrador, un decodificador y una memoria compartida.

- La Figura 2 es una representación esquemática parcial de una primera y de una segunda tablas de correspondencia correspondientes, y

- La Figura 3 es una representación esquemática de un método para acceder a un contenido multimedia protegido utilizando el terminal de la Figura 1.

En estas figuras, se dan las mismas referencias para designar los mismos elementos.

En la siguiente descripción, las características bien conocidas por los expertos en esta técnica no se describen en detalle.

La Figura 1 muestra un terminal capaz de acceder a un contenido multimedia protegido. Para este propósito, este terminal incluye un descifrador 1, un decodificador 3, una memoria 5 compartida por el descifrador y el decodificador, un ordenador electrónico programable 7 y una memoria 9.

El descifrador 1 puede memorizar información en la memoria compartida 5 y, tal como el decodificador 3, extraer información de la misma.

El ordenador electrónico 7 es capaz de ejecutar instrucciones memorizadas en la memoria 9. La memoria 9 incluye las instrucciones necesarias para la ejecución del método de la Figura 3.

Las memorias compartidas 5 y 9 pueden ser independientes, tal como en la Figura 1, o combinadas, o también la memoria compartida 5 puede ser una parte de la memoria 9.

La Figura 2 muestra una parte de una primera tabla de correspondencia 12 y una parte de una segunda tabla de correspondencia 18. En esta figura, un byte se da en forma hexadecimal y el símbolo "|" representa la operación de concatenación.

La primera tabla 12 incluye una línea para cada valor de byte posible. Por lo tanto, comprende una primera columna que contiene todas las cadenas posibles 14 de ocho bits originales. El número de bits de la cadena 14, indicado M a continuación, es en este caso igual a ocho. Por lo tanto, existe 2^M cadenas 14 cuyos valores están comprendidos entre 0x01 y 0xFF. Cada línea de la tabla 12 asocia en su cadena 14, una concatenación de un primer y un segundo byte, es decir, una cadena 16 de dieciséis bits sustituidos. El número de bits de la cadena 16, indicado N a continuación, se considera en este caso, igual a dieciséis. Por lo tanto, existe 2^N cadenas 16 posibles, pero la tabla 12 contiene solamente 2^M. Cada cadena 16 es distinta de las otras cadenas 16. Las cadenas 16 están contenidas en una segunda columna de la tabla 12.

Las líneas de la tabla 12 se clasifican por orden de valores de las cadenas 14 para constituir un índice de esta tabla. Este índice hace posible acelerar la búsqueda y la selección de una línea de la tabla 12 que contenga una cadena 14 dada. Además, el valor de este índice varía en una etapa regular de un valor al siguiente, lo que permite identificar directamente la línea de la tabla 12 que contiene este índice sin tener que leer el contenido de otras líneas. En este caso, el valor del índice es igual al número de la línea de la tabla 12 que contiene esta cadena 14. Ello, por lo tanto, hace posible buscar con mayor rapidez una cadena 14 dada. Para simplificar la ilustración, la Figura 2 representa

solamente las ocho líneas relacionadas con las cadenas 14 de valores, respectivamente, 0xA0 a 0xA7. A modo de ejemplo, la primera línea representada en la tabla 12 asocia la cadena 16 de dieciséis bits sustituidos 0xD4 | 0x03 con la cadena 14 de ocho bits originales 0xA0. La cadena 0xA0 está en la línea 0xA0, es decir la línea 160ª línea de la tabla 12 a partir de la primera línea de esta tabla.

Además, en esta tabla 12, el conjunto de los valores de los segundos bytes entre los bytes concatenados, es decir, los ocho bits de índices 9 a 16 de cada cadena 16, es igual al conjunto de cadenas 14. En la Figura 2, los segundos bytes visibles de las cadenas 16 están comprendidos entre 0x01 y 0x08.

La segunda tabla 18 de la Figura 2 incluye una línea para cada cadena 16 de dieciséis bits sustituidos. Cada línea asocia, a esta cadena 16, la cadena 14 a la cual esta cadena 16 está asociada por la tabla 12. Por lo tanto, se dice que estas tablas 12 y 18 son "correspondientes". Las cadenas 16 y 14 están contenidas, respectivamente, en la primera y segunda columnas de la tabla 18. La primera línea representada de la tabla 18 asocia, por ejemplo, la cadena 14 de ocho bits originales 0xA4 con la cadena 16 de dieciséis bits sustituidos 0x5b | 0x01.

Además, las líneas de la tabla 18 se clasifican en orden de valores de los segundos bytes, es decir, los ocho bits de los índices 9 a 16 de cada cadena 16. Los valores de estos segundos bytes se ejecutan en el conjunto de valores posibles entre 0x01 y 0xFF. Por lo tanto, constituyen un índice de esta segunda tabla. Tal como se mencionó con anterioridad, este índice hace posible acelerar la búsqueda y la selección de una línea de la tabla 18 que contiene una cadena 16 dada. De hecho, puesto que el valor de este índice varía en una etapa regular de un valor al siguiente, ello permite identificar directamente la línea de la tabla 18 que contiene este índice sin tener que leer el contenido de otras líneas. En este caso, el valor del índice es igual al número de la línea de la tabla 18 que contiene esta cadena 16. En particular, ello permite un ahorro de tiempo en comparación con el caso en que los valores del índice no se distribuyen uniformemente entre sus valores extremos. En la Figura 2, los segundos bytes representados toman los valores comprendidos entre 0x01 y 0x08 y ocupan, respectivamente y en orden ascendente, las líneas uno a ocho de la tabla 18.

El funcionamiento del terminal de la Figura 1 se describirá a continuación con referencia al método de la Figura 3.

El método comienza con una fase de inicialización. Esta fase de inicialización incluye:

- etapas 110 y 112 durante las cuales se establecen y memorizan las tablas 12, 18, y
- una etapa 114 de establecimiento de un número S de inicialización.

Esta fase de inicialización se activa, por ejemplo, cada vez que se inicia el terminal. En este caso, preferiblemente, las tablas restablecidas y el número S se memorizan en una memoria volátil, tal como una memoria RAM (memoria de acceso aleatorio), para borrarse automáticamente cuando se desactiva el terminal.

Más concretamente, durante la etapa 110, el terminal se establece así y memoriza la tabla 12 en una memoria accesible para el descifrador 1. Esta memoria puede ser una memoria específica para este último, una parte de la memoria 9 o la memoria 5 compartida. A modo de ejemplo, la tabla 12 se establece así realizando las siguientes operaciones para cada posible cadena 14:

a) el terminal extrae, de manera aleatoria, una cadena 16 en el conjunto de 2^N cadenas 16 posibles, y luego

b) si la cadena 16 así extraída al azar no se ha asociado ya con otra cadena 14, entonces está asociada en la tabla 12 con esta cadena 14 y, en caso contrario, se repiten las operaciones a) y b).

A continuación, durante una etapa 112, el terminal se establece y memoriza la tabla 18 correspondiente en una memoria accesible para el decodificador 3. Esta memoria puede ser una memoria específica para este último, una parte de la memoria 9 o la memoria 5 compartida. A modo de ejemplo, durante esta etapa 112, la tabla 18 se establece a partir de la tabla 12 previamente registrada durante la etapa 110. Para ello, por ejemplo, se intercambian los contenidos de la primera y segunda columnas de la tabla 12 y luego las líneas de esta nueva tabla se clasifican en orden ascendente o descendente de los valores de los segundos bytes de la primera columna. En esta forma de realización, las líneas se clasifican en orden ascendente de los valores de los segundos bytes.

La etapa 110 se pone en práctica, un número predeterminado t_1 de veces, de manera que memorice t_1 distintas tablas 12. Preferiblemente, el número t_1 es igual a una potencia estrictamente positiva de dos. Asimismo, la etapa 112 se ejecuta t_2 veces para memorizar t_2 distintas tablas 18. Los tiempos t_1 y t_2 son números enteros positivos o nulos, y t_1 es mayor o igual que t_2. En esta forma de realización, t_1 y t_2 son iguales. En este caso, durante cada ejecución de la etapa 110, los números M y N se modifican respetando siempre el hecho de que el número N es estrictamente mayor que el número M. A modo de ejemplo, en cada iteración de la etapa 110, el número M se extrae de forma aleatoria o pseudoaleatoria dentro de un rango predeterminado de valores posibles. A continuación, lo mismo se hace con el número N. En condiciones normales, el número M varía entre 4 y 16 bits y el número N varía entre 8 y 24 bits. Las tablas 12 que difieren entre sí se obtienen así, además, por los valores de los números M y N.

Cada una de las tablas memorizadas 12 se identifica por medio de un índice i comprendido entre 0 y t_1-1. Asimismo, cada una de las tablas memorizadas 18 se identifica mediante un índice. El índice de las tablas 18 se elige de modo que las tablas correspondientes 12 y 18 tengan el mismo índice i. A continuación, se indican por T [i] y L [i], respectivamente, las tablas correspondientes 12 y 18 del índice i.

En la etapa 114, el terminal extrae, aleatoria o pseudo-aleatoriamente, un número entero estrictamente positivo de k bytes. Este número entero constituye el número S. De manera preferible, este número S se extrae aleatoria o pseudo-aleatoriamente de un conjunto que contiene solamente potencias estrictamente positivas de dos. El número k es un número predeterminado o se extrae al azar o pseudo-aleatoriamente de un conjunto predeterminado de valores. De manera similar, en esta forma de realización, el terminal también extrae de manera aleatoria o pseudo-aleatoriamente un número entero u estrictamente positivo. Este número u representa un número de usos consecutivos de la misma tabla 12. A continuación, los números S y u así establecidos se ponen a disposición del descifrador 1 y del decodificador 3. Para este propósito, el terminal memoriza los números S y u así tal como la longitud k en bytes del número S en la memoria compartida 5. Además, en esta forma de realización, el número t_1 también se memoriza en la memoria 5.

Antes, en paralelo con o cerca de la fase de inicialización, durante una etapa 100, el descifrador 1 recibe un fragmento cifrado de un contenido multimedia y una clave de descifrado para este fragmento cifrado. De una manera conocida por un experto en esta técnica, el fragmento cifrado y la clave de descifrado pueden recibirse conjuntamente, es decir al mismo tiempo y a través de la misma red de comunicación o un mismo soporte de registro interno o externo al terminal. El fragmento cifrado y la clave de descifrado también se pueden recibir por separado, es decir, en instantes separados o a través de redes de comunicación o soportes de registro, internos o externos al terminal, distintos. En este caso, tal como se describió con anterioridad, el terminal recibe la clave de descifrado del fragmento encriptado en una licencia DRM.

Después de la etapa 100 y de la fase de inicialización, durante una etapa 120, el descifrador 1 descifra el fragmento con la clave de descifrado, ambos recibidos en la etapa 100, obteniendo así un fragmento no cifrado.

Después de la etapa 120, y antes de que el fragmento descifrado se registre en la memoria 5, durante una etapa 130, el descifrador sustituye los bits originales del fragmento de texto sin cifrar con distintos bits sustituidos para obtener un fragmento modificado.

Más concretamente, el descifrador 1 sustituye, por bloques formados etapa a etapa, el conjunto de los bits originales del fragmento no cifrado por bits sustituidos. Cada uno de estos bloques de bits se denominará en adelante Px. El índice "x" indica el número de orden o el rango del bloque P_x. En este caso, el número de orden del primer bloque de bits del fragmento no cifrado es "1" y el del último bloque de bits es un número entero "X" desconocido a priori, y determinado al final de la etapa 130. El índice "x" está por lo tanto comprendido entre 1 y X. La concatenación del conjunto de los bloques P_x en el orden de los índices x es igual al fragmento no cifrado.

Para ello, para cada uno de estos bloques P_x, el descifrador 1 pone en práctica una operación 132, y luego, una operación 134.

Durante la operación 132, el descifrador 1 selecciona en primer lugar una de las tablas 12 memorizadas durante la etapa 110 con el fin de utilizarla para sustituir los bits originales de este bloque P_x con bits sustituidos.

Para este propósito, en este caso, el descifrador 1 extrae de la memoria 5 el número S y su longitud k en bytes, el número u, así como el número t_1 de las tablas 12 memorizadas en la etapa 110. A continuación, el índice i de la tabla seleccionada T [i] se determina utilizando la siguiente fórmula: i = S [((x-1) div u) módulo k] módulo t_1, en donde

- S [j] es la función que devuelve el byte del índice j en el número S,
- div designa la operación de división euclidiana.

Una vez que se ha seleccionado la tabla T [i], se conoce el número M de bits de las cadenas 14 contenidas en la primera columna de esta tabla T [i].

A continuación, durante la operación 134, el descifrador 1 realiza la sustitución de los bits originales del bloque P_x por bits sustituidos por medio de la tabla T [i] seleccionada durante la ejecución previa de la operación 132. Más concretamente, si permanece al menos M bits originales del fragmento de texto sin cifrar a sustituir, el descifrador 1 constituye el bloque P_x utilizando los M primeros bits del fragmento de texto sin cifrar que quedan por sustituir. Por lo tanto, este bloque P_x constituye una cadena de M bits originales consecutivos del fragmento no cifrado. La sustitución consiste entonces en:

- buscar y leer, en la tabla seleccionada 12, la cadena 16 asociada con la cadena 14 que tiene el mismo valor que este bloque P_x, y

- concatenar, en orden de lectura, esta cadena 16 leída a las cadenas 16 anteriores leídas.

En esta forma de realización, la búsqueda es particularmente simple y rápida porque el número de línea de la tabla T [i] que contiene la cadena 16 a leer puede establecerse directamente a partir del valor del bloque P_x y sin consultar el contenido de otras líneas en esta tabla. A modo de ejemplo, la cadena 16 se lee en la línea del número P_x de la tabla T [i].

Al repetir las operaciones 132 y 134, se establecer así un fragmento modificado etapa a etapa en donde cada bloque P_x se sustituye por una cadena 16 respectiva.

Si quedan menos de M bits originales del fragmento de texto sin cifrar que se van a sustituir, los bits originales del fragmento de texto sin cifrar que quedan por sustituirse, se complementan arbitrariamente con bits adicionales para constituir un bloque de M bits consecutivos para procesarse como los anteriores.

A modo de ejemplo, en este caso, si la tabla T [i] del índice i es la de la Figura 2, y si el bloque P_x del fragmento no cifrado es igual a la cadena 14 de ocho bits originales 0xA4, entonces el bloque del índice x del fragmento modificado es la cadena 16 de dieciséis bits sustituidos 0x5B | 0x01.

A continuación, cuando las operaciones 132 y 134 se han puesto en práctica para el conjunto de los bloques P_x, durante una etapa 140, el descifrador 1 memoriza solamente el fragmento modificado así obtenido en la memoria compartida 5.

A continuación, durante una etapa 150, el decodificador 3 extrae el fragmento modificado de la memoria compartida 5.

A continuación, durante una etapa 160, el decodificador 3 sustituye los bits sustituidos del fragmento modificado con los bits originales para reestablecer el fragmento no cifrado a partir del fragmento modificado.

Más concretamente, el decodificador 3 sustituye, por bloques formados etapa a etapa, el conjunto de los bits sustituidos del fragmento modificado por los bits originales. Cada uno de estos bloques de bits se denominará en adelante P_x*. El índice "x" designa el número de orden o el rango del bloque P_x*. En este caso, el número de orden del primer bloque de bits del fragmento modificado es "1" y el del último bloque de bits es un número entero "X" a priori desconocido, y se determina al final de la etapa 160. El índice "x" está por lo tanto comprendido entre 1 y X. La concatenación de los bloques P_x* en el orden de los índices x es igual al fragmento modificado.

Para ello, para cada uno de estos bloques P_x*, el decodificador 3 pone en práctica una operación 162, y luego, una operación 164.

Durante la operación 162, el decodificador 3 determina en primer lugar el índice i de la tabla T [i] seleccionada y utilizada en la etapa 130. A continuación, selecciona la tabla L [i] que forma con la tabla T [i] un par de tablas correspondientes.

Para este propósito, en este caso, el decodificador 3 extrae de la memoria 5 el número S y su longitud k en bytes, el número u, así como el número t_1 de tablas 12 memorizadas en la etapa 110. A continuación, el índice i de la tabla L [i] se determina utilizando la siguiente fórmula: i = S [((x-1) div u) módulo k] módulo t_1, donde x es el número de orden del bloque P_x* que se procesará durante la siguiente operación 164. A continuación, el decodificador 3 selecciona la tabla L [i] del índice i. A partir de este momento, se conoce el número N de bits de las cadenas 16 de la tabla L [i].

A continuación, durante la operación 164, el decodificador 3 realiza la sustitución de los bits sustituidos del bloque P_x* por bits originales por medio de la tabla L [i] seleccionada durante la ejecución previa de la operación 162. Para ello, establece el bloque Px* con los primeros N bits del fragmento modificado que queda por sustituirse. El bloque P_x* es, por lo tanto, una cadena de N bits sustituidos consecutivos. La sustitución consiste entonces en:

- leer, en la tabla L [i] seleccionada, la cadena 14 asociada con la cadena 16 cuyo valor es igual al del bloque P_x*, y

- concatenar, en el orden de lectura, esta cadena 14 leída con las anteriores cadenas 14 leídas.

La lectura de la cadena 14 en la tabla L [i] es, en este caso, particularmente rápida porque el segundo byte de la cadena 16 hace posible restablecer directamente el número de la línea de la tabla L [i] que contiene la cadena 14 a leer sin consultar el contenido de otras líneas en esta tabla. A modo de ejemplo, la cadena 14 se lee en la línea de la tabla L [i] cuyo número de línea es igual al valor del segundo byte de la cadena 16.

Al repetir las operaciones 162, 164, el decodificador reestablece gradualmente el fragmento no cifrado.

Durante la operación 164, si el decodificador 3 detecta que se trata del último bloque P_x* para procesar del fragmento modificado, entonces el decodificador 3 sustituye el último bloque P_x* como los anteriores. A continuación, solamente concatenará con las cadenas 14 precedentes leídas más que los primeros bits útiles de la cadena 14. Para determinar el número de bits útiles de esta cadena 14, el decodificador 3 utiliza, por ejemplo, una información de longitud contenida en la parte del fragmento no cifrado ya obtenido. La posición de esta información de longitud en el fragmento de texto sin cifrar se fija mediante el formato de codificación conocido utilizado para los fragmentos de texto sin cifrar. Esta información de longitud también puede ser una información predeterminada, relacionada con la estructura de cada fragmento e impuesta por el formato de codificación utilizado.

A modo de ejemplo, si la tabla L [i] es la de la Figura 2, y si el bloque P_x* es la cadena de dieciséis bits sustituidos 0x5B | 0x01, entonces el bloque P_x reestablecido a partir del fragmento no cifrado es la cadena 14 de ocho bits originales 0xA4.

A continuación, cuando se han puesto en práctica las operaciones 162 y 164 para el conjunto de los bloques P_x*, y antes de que el fragmento no cifrado se guarde posiblemente en una memoria del terminal, durante una etapa 170, el decodificador 3 decodifica el fragmento no cifrado obtenido. En este caso, el fragmento no cifrado obtenido por el decodificador nunca se registra en una memoria compartida y, en particular, en la memoria 5.

Por último, durante una etapa 180, el decodificador 3 transmite el fragmento decodificado a un dispositivo multimedia capaz de reproducir este fragmento decodificado del contenido multimedia para que sea directamente perceptible y entendible por un ser humano.

Son posibles muchas otras formas de realización de la invención. A modo de ejemplo, el contenido multimedia se proporciona protegido por un sistema de acceso condicional, o CAS, en inglés Conditional Access System. A continuación, se utiliza la terminología del dominio de los sistemas de acceso condicional. El lector interesado puede, por ejemplo, encontrar una presentación más completa en el documento: "Modelo funcional de un sistema de acceso condicional", EBU Review, Technical European Broadcasting Union, Bruselas, BE, N° 266, 21 de diciembre de 1995. En este caso, la clave de contenido se suele designar como la palabra de control y se recibe en un mensaje de control de los títulos de acceso, o ECM, en inglés Entitlement Control Message.

En otra forma de realización, el contenido multimedia se proporciona protegido por cualquier otro tipo de sistema de protección de contenidos, tal como, por ejemplo, un sistema de protección de datos más convencional que no gestiona los derechos de acceso. En este caso, la clave de contenido se puede recibir en cualquier otro tipo de mensaje de transmisión.

La fase de inicialización puede ser activada por otros eventos. A modo de ejemplo, la ejecución de las etapas 110 y 112 se activa en respuesta a la ejecución de un módulo para acceder al contenido protegido del terminal. También se puede activar periódicamente o después de un número predeterminado de usos de las tablas 12 y 18 o del número S restablecido previamente.

De manera alternativa, durante las etapas 110 y 112, las tablas 12 y 18 pueden calcularse previamente, y luego, ser estáticas e integrarse en las instrucciones memorizadas en la memoria 9 tal como parte del desarrollo de estas instrucciones.

Como variante, los números M y N son constantes en cada ejecución de la etapa 110. A modo de ejemplo, el número M se elige igual a ocho y el número N se elige igual a un múltiplo entero de ocho mayor o igual que dieciséis. En otra variante, solamente el número N varía de una ejecución a otra de la etapa 110, mientras que el número M permanece constante.

Los M bits de cada cadena 16 de las tablas L [i] utilizadas para formar el índice de esta tabla no son necesariamente los últimos M bits de cada cadena 16. De hecho, estos M bits pueden ubicarse en cualquier lugar en la cadena 16 a partir del momento en que el decodificador 3 conoce la ubicación de estos M bits. A modo de ejemplo, estos M bits ocupan solamente las posiciones de índice par en la cadena 16.

En otra variante, la primera columna de la tabla L [i] simplemente se clasifica en orden ascendente o descendente de los valores de las cadenas 16 y no se utiliza ningún índice como el descrito previamente para la tabla 18.

En otra forma de realización, después de la puesta en práctica de la etapa 110, se omite la etapa 112. Solamente se memorizan las primeras tablas 12 para que estén disponibles, no solamente para el descifrador 1, sino también para el decodificador 3. En esta forma de realización, es por lo tanto la misma tabla 12 que se utiliza durante las operaciones 134 y 164. Durante la operación 164, el valor del bloque P_x* se busca en la segunda columna de la tabla 12, por ejemplo, consultando sucesivamente en un orden predeterminado las cadenas 16 presentes en la segunda columna de esta tabla.

En otra forma de realización, después de la puesta en práctica de la etapa 110, se pone en práctica la etapa 112 un número t_2 de veces, estrictamente positivo y estrictamente menor que t_1, de modo que solamente t_2 tablas 18 sean

memorizadas. A modo de ejemplo, para los valores del índice i comprendidos entre [0; t_2-1], se establece una tabla L [i] para cada tabla T [i] restablecida. Por el contrario, para los valores del índice i comprendidos entre [t_2; t_1-1], solamente se establece la tabla T [i]. En el último caso, la tabla T [i] se memoriza para que esté disponible no solamente para el descifrador 1, sino también para el decodificador 3. Posteriormente, si durante la etapa 162 el valor del índice i está comprendido entre [0; t_2-1], a continuación, durante la operación 164, el valor del bloque P_x* se busca en la tabla L [i] correspondiente a la tabla T [i]. Por el contrario, si durante la etapa 162, el valor del índice i está entre [t_2; t_1-1], durante la operación 164, el valor del bloque P_x* se busca en la segunda columna de la tabla T [i].

Como variante, durante la etapa 114, para hacer que el número o números S, k, u generados estén disponibles para el descifrador 1 y el decodificador 3, el terminal los memoriza, a la vez, en una memoria específica para el descifrador 1 y en una memoria específica del decodificador 3.

En una forma de realización simplificada, t_1 es igual a uno, es decir que se utiliza una única tabla 12 para crear la totalidad del fragmento o fragmentos modificados.

Las operaciones 132 y 162 pueden llevarse a cabo de manera distinta. A modo de ejemplo, se puede omitir el uso del número u. En este caso, el índice i de la tabla seleccionada 12 o 18 se determina usando la siguiente fórmula: i = S [(x-1) módulo k] módulo t_1. También es posible determinar el índice i sin usar el número S. A modo de ejemplo, en este último caso, se utiliza la siguiente fórmula: i = x módulo t_1. En este caso, durante la etapa 114, los números S, u y k no se establecen.

En otra forma de realización, se omiten las etapas 110, 112 y 114. Por lo tanto, ninguna tabla 12, 18 se memoriza antes del comienzo de la etapa 130. En este caso, durante la operación 132, se establece una tabla 12 a medida que se realiza la operación 132. A modo de ejemplo, en este caso, se supone que M y N son constantes conocidas y que la tabla 12 está inicialmente vacía. A continuación, si el valor del siguiente bloque P_x del fragmento no cifrado no corresponde a ninguna cadena 14 ya contenida en la tabla 12, se genera una cadena 16 y a continuación se memoriza en la tabla 12 en asociación con los bits originales del bloque P_x. A modo de ejemplo, se genera la cadena 16:

a) extrayendo aleatoria o pseudo-aleatoriamente una cadena 16 en el conjunto de las 2^N cadenas 16 posibles, y luego

b) si la cadena 16 así extraída al azar no se ha asociado ya con otra cadena 14, entonces se asocia en la tabla 12 con esta cadena 14 y, en caso contrario, se repiten las operaciones a) y b).

De esta forma, una denominada tabla dinámica se establece etapa a etapa. Esta tabla dinámica difiere de una tabla 12 solamente en que puede no contener todas las cadenas 14 posibles. En este caso, esta tabla dinámica contiene solamente las cadenas 14 necesarias para procesar el fragmento no cifrado. A continuación, durante la etapa 162, se establece una tabla dinámica correspondiente, por ejemplo, tal como se describió para la etapa 112.

En otra forma de realización, el uso de las tablas 12, 18 se sustituye por el uso de una función de cifrado F y de una función de descifrado F^{-1}. Más concretamente, la función F ejecuta varias operaciones aritméticas y lógicas en los bits de una cadena 14 para generar la cadena 16 que le está asociada. La función F^{-1} es la inversa de la función F. Por lo tanto, devuelve el valor de la cadena 14 asociada con una cadena 16. Por lo tanto, las funciones F y F^{-1} se utilizan en lugar de las tablas 12, 18 durante la ejecución, respectivamente, de las operaciones 134 y 164.

En otra forma de realización, el conjunto de los bits originales no se sustituye por bits sustituidos durante la etapa 130. Para este propósito, por ejemplo, durante la etapa 134, solamente una fracción predeterminada de los bloques P_x son objeto de la sustitución descrita con anterioridad. Para los otros bloques P_x, los bits originales se concatenan directamente con el resultado del procesamiento de los bloques anteriores. La fracción predeterminada de los bloques sustituidos se puede modificar para cada fragmento de texto no cifrado procesado. Por ejemplo, para los fragmentos no cifrados pares, el método aplica las operaciones 132 y 134 solamente a un bloque P_x de cada dos y, para fragmentos impares, solamente a un bloque P_x de cada tres. Durante la etapa 160, el decodificador 3 realiza las sustituciones recíprocas de las realizadas durante la etapa 130. Por lo tanto, si el conjunto de bits originales no se sustituye por bits sustituidos durante la etapa 130, durante la etapa 160, solamente una fracción de los bloques se identifica como bloques de bits sustituidos P_x*. Solamente estos bloques P_x* identificados están sujetos a la sustitución descrita con anterioridad, estando los demás bits directamente concatenados con el resultado del procesamiento de los bloques anteriores.

Como variante, durante la etapa 134, el descifrador 1 constituye bloques P_x de longitud mayor que M, y solamente sustituye una cadena de M bits originales de este bloque P_x con una cadena 16. Esta cadena de M bits originales se encuentra en la misma posición predeterminada en cada bloque P_x. A modo de ejemplo, esta cadena de M bits originales se encuentra al comienzo del bloque P_x. Esto equivale a insertar, en el fragmento modificado, una cadena de bits originales entre cada cadena 16. Durante la etapa 164, el decodificador 3 constituye bloques P_x* de longitud predeterminada mayor que N, y solamente sustituye la parte predeterminada de estos bits correspondiente a la cadena 16. Los bits restantes del bloque P_x* permanecen sin cambios. La longitud del bloque P_x y la posición en el bloque P_x de la cadena de M bits originales sustituida puede ser cada una constante o variable.

De manera alternativa, asimismo, durante la operación 134, cuando quedan menos de M bits originales del fragmento de texto sin cifrar a sustituir, estos últimos bits originales no se sustituyen, sino que se concatenan directamente a las cadenas 16 anteriores leídas. En este caso, durante la operación 164, el último bloque P_x* tiene menos de M bits, y estos bits no se sustituyen, sino que se concatenan directamente a las cadenas 14 anteriores leídas.

En otra variante, durante la etapa 140, el fragmento modificado se memoriza en la memoria 5, junto con la longitud útil, en bits, del fragmento no cifrado o de su último bloque P_x. A continuación, el decodificador 3 extrae de la memoria 5, durante la etapa 150, el fragmento modificado, así como esta longitud útil. A continuación, durante la operación 164, el decodificador 3 sustituye el último bloque P_x* como los anteriores, pero solamente concatena, con las cadenas 14 anteriores leídas, solamente los primeros bits útiles de la cadena 14 así leídos. El número de primeros bits útiles se determina a partir de la longitud útil extraída durante la etapa 150.

REIVINDICACIONES

1. Método para acceder a un contenido multimedia protegido, mediante un terminal que comprende un descifrador, un decodificador y una memoria compartida, en donde:

- durante una fase de descifrado, el descifrador:

  a) recibe (100) un fragmento cifrado del contenido multimedia y una clave de descifrado para este fragmento cifrado,

  b) descifra (120) el fragmento recibido con la clave de descifrado recibida para obtener un fragmento no cifrado,

  c) sustituye (134) bits originales del fragmento de texto sin cifrar con distintos bits sustituidos para obtener un fragmento modificado, y luego

  d) memoriza (140) el fragmento modificado en la memoria compartida,

- durante una fase de decodificación, el decodificador:

  e) extrae (150) el fragmento modificado de la memoria compartida,

  f) sustituye (164) los bits sustituidos del fragmento modificado extraído por los bits originales para reestablecer el fragmento no cifrado,

  g) decodifica (170) el fragmento no cifrado reestablecido para obtener un fragmento decodificado, y luego

  h) transmite (180) el fragmento decodificado a un dispositivo multimedia capaz de reproducir este fragmento decodificado del contenido multimedia,

caracterizado por cuanto que:

- durante la etapa c) (130), una cadena (14) de M bits originales consecutivos del fragmento no cifrado se sustituye por una cadena (16) de N bits sustituidos consecutivos, donde N es estrictamente mayor que M y la cadena de N bits sustituidos es distinta para cada cadena distinta de M bits originales, y

- durante la etapa f) (160), la cadena de N bits sustituidos se sustituye por la cadena de M bits originales, y

- antes de la ejecución de las etapas b), c), d), e) y f), el método comprende:

  • la memorización (110), por el terminal, de una primera tabla de correspondencia (12) que comprende una línea para cada posible cadena de M bits originales, asociando cada línea, con una posible cadena de M bits originales, una cadena de N bits sustituidos diferentes de los asociados con las otras posibles cadenas de M bits originales, clasificándose las líneas de la primera tabla en orden de valores de las posibles cadenas de M bits originales para constituir un índice de esta primera tabla, y

- la etapa c) incluye:

  • la lectura, en la primera tabla de correspondencia, de la cadena de N bits sustituidos asociadas con la cadena de M bits originales del fragmento no cifrado, y luego

  • la sustitución de la cadena de M bits originales del fragmento no cifrado por la cadena de N bits sustituidos así seleccionada para obtener el fragmento modificado, y

- antes de la ejecución de las etapas b), c), d), e) y f), el método comprende:

  • la memorización (112), por el terminal, de una segunda tabla de correspondencia (18) que comprende una línea para cada cadena de N bits sustituidos, asociando esta línea, con esta cadena de N bits sustituidos, la cadena de M bits originales a la que esta cadena de N bits sustituidos estaba asociada con la primera tabla, siendo estas tablas primera y segunda denominadas "correspondiente" puesto que la segunda tabla permite reestablecer el fragmento no cifrado a partir del fragmento modificado obtenido utilizando la primera tabla ,

- la etapa f) comprende:

  • la lectura en la segunda tabla de correspondencia, correspondiente a la primera tabla, de la cadena de M bits originales asociada con la cadena de N bits sustituidos del fragmento modificado, y luego

- la sustitución de la cadena de N bits sustituidos del fragmento modificado con la cadena de M bits originales así leída para reestablecer el fragmento no cifrado, y

- el conjunto de los valores de M bits de índices dados, dos a dos distintos de cada cadena de N bits sustituidos, es igual al conjunto de los valores posibles de las cadenas de M bits, y las líneas de la segunda tabla se clasifican solamente por orden de valores de estos M bits de índices dados de cada cadena de N bits sustituidos, de modo que estos M bits de índices dados constituyan un índice de esta segunda tabla.

2. Método según la reivindicación 1, en donde el método comprende:

- antes de la ejecución de las etapas b), c), d), e) y f), la memorización, por el terminal, de varias primeras tablas de correspondencia distintas, y luego

- durante la ejecución de la siguiente etapa c), la selección (132) de una de estas primeras tablas, y luego el uso de la primera tabla seleccionada, y a continuación

- durante la ejecución de la siguiente etapa f), la determinación (162) de la primera tabla seleccionada en la etapa c).

3. Método según la reivindicación 2, en donde al menos dos de las primeras tablas de correspondencia memorizadas distintas difieren entre sí al menos en la longitud M de las cadenas de bits originales o por la longitud N de las cadenas de bits sustituidos.

4. Método según cualquiera de las reivindicaciones 2 o 3, en donde:

- el método comprende, antes de la ejecución de las etapas b), c), d), e) y f), la extracción 114 aleatoria o pseudoaleatoria de un número, por el terminal, y su puesta a disposición del descifrador y del decodificador, y

- la selección, en la etapa c), de una primera tabla de correspondencia, y luego la determinación, en la etapa f), de la primera tabla seleccionada en la etapa c), que se realizan en función de este número extraído.

5. Método según una cualquiera de las reivindicaciones 2 a 4, en donde el número de las primeras tablas de correspondencia distintas memorizadas o el número de bytes del número extraído de forma aleatoria o pseudoaleatoria, al menos, se toma igual a una potencia estrictamente positiva de dos.

6. Soporte (9) para registrar información, caracterizado porque incluye instrucciones para la puesta en práctica de un método de conformidad con cualquiera de las reivindicaciones anteriores, cuando estas instrucciones son ejecutadas por un ordenador (7) electrónico.

7. Terminal que comprende un descifrador (1), un decodificador (3) y una memoria compartida (5), caracterizado porque este terminal comprende un ordenador electrónico programado para ejecutar las etapas c) y f) de un método de acceso a un contenido multimedia protegido según cualquiera de las reivindicaciones 1 a 5.

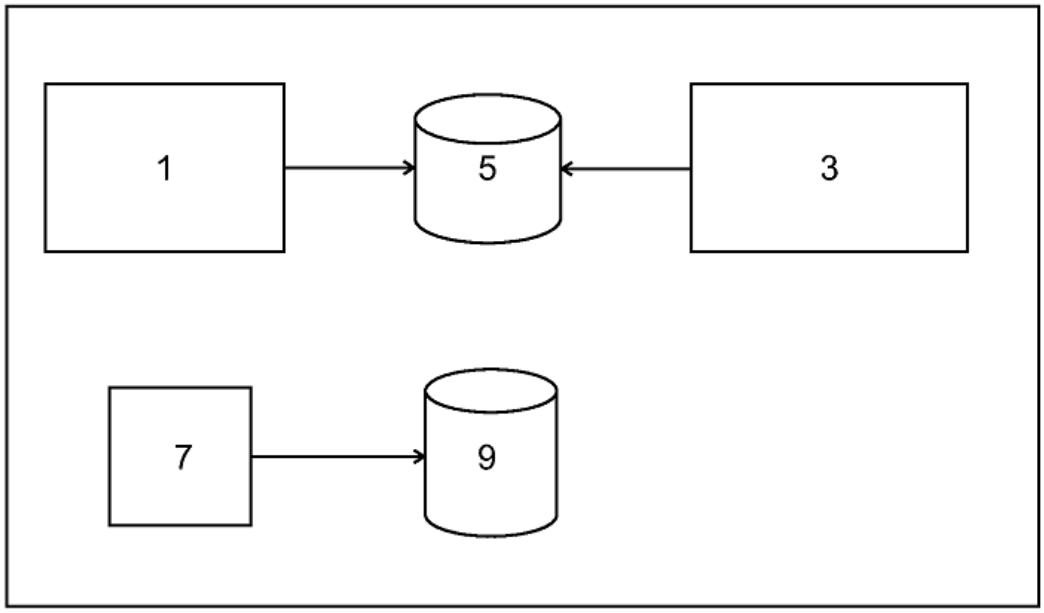
1
5
3
7
9

Fig. 1

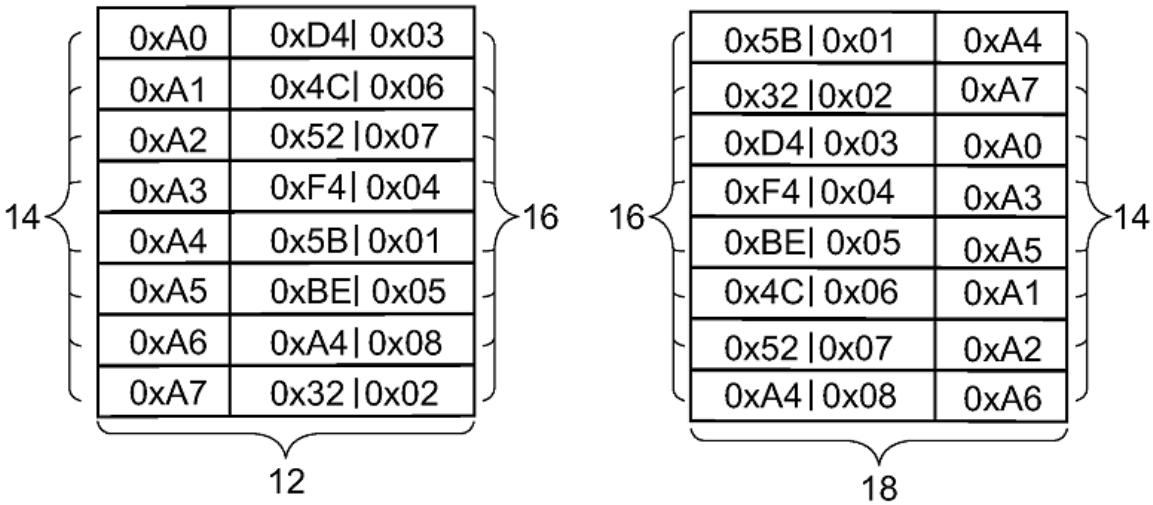
0xA0 0xD4| 0x03
0xA1 0x4C| 0x06
0xA2 0x52 |0x07
0xA3 0xF4| 0x04
0xA4 0x5B| 0x01
0xA5 0xBE| 0x05
0xA6 0xA4| 0x08
0xA7 0x32 |0x02
14
16
12
0x5B| 0x01 0xA4
0x32 |0x02 0xA7
0xD4| 0x03 0xA0
0xF4 |0x04 0xA3
0xBE| 0x05 0xA5
0x4C| 0x06 0xA1
0x52 |0x07 0xA2
0xA4| 0x08 0xA6
16
14
18

Fig. 2

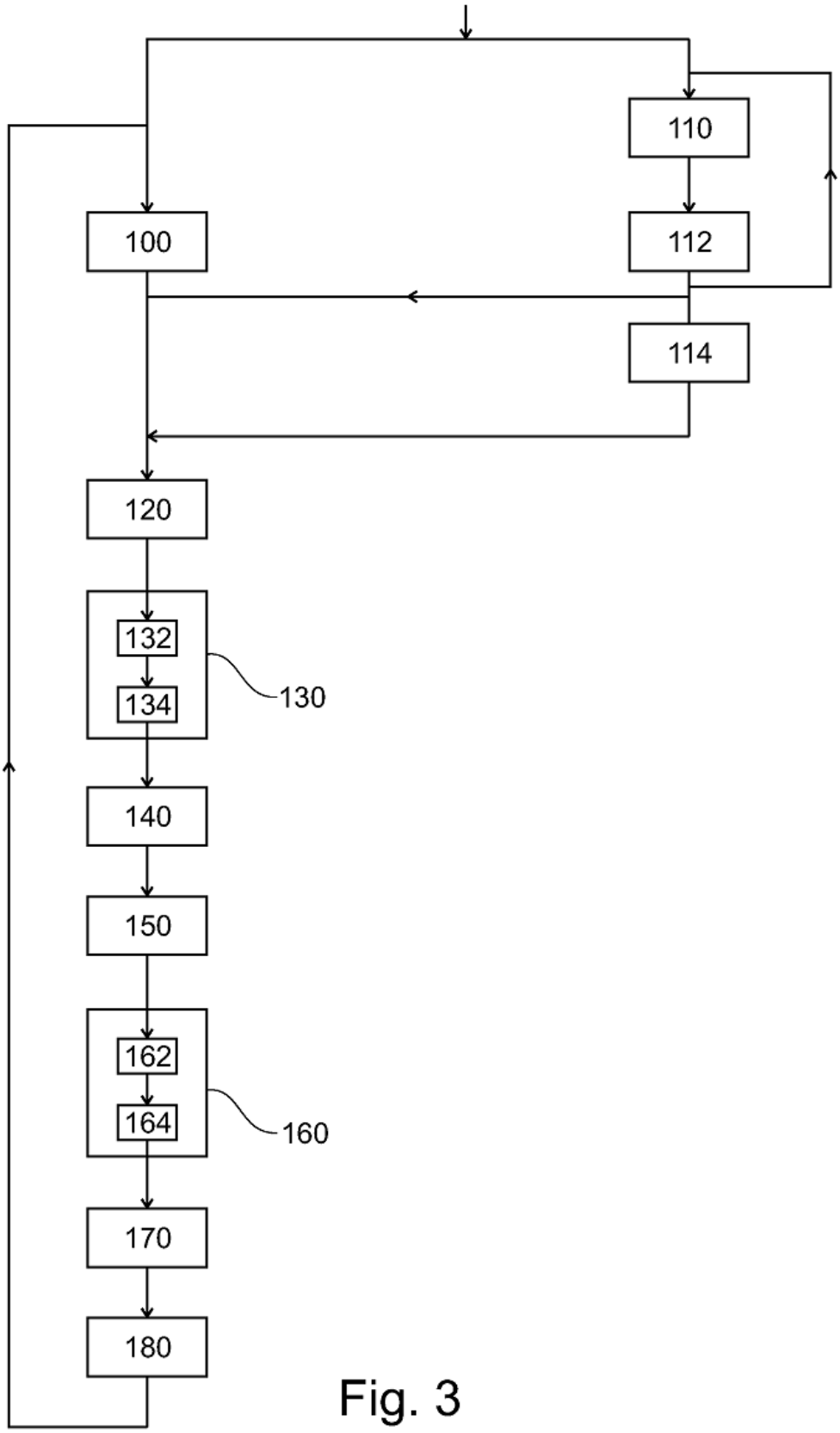

Fig. 3