(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 658 311**

(51) Int. CI.:

H04N 21/4405 (2011.01)
H04N 21/6437 (2011.01)
H04N 21/6334 (2011.01)
H04N 21/835 (2011.01)
H04N 21/266 (2011.01)
H04L 9/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **22.03.2016 E 16161668 (5)**

(97) Fecha y número de publicación de la concesión europea: **18.10.2017 EP 3073752**

(54) Título: **Procedimiento de generación de un vector de inicialización para el cifrado de un contenido de vídeo**

(30) Prioridad:

23.03.2015 FR 1500564

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
09.03.2018

(73) Titular/es:

THALES (100.0%)
Tour Carpe Diem, Place des Corolles, Esplanade Nord
92400 Courbevoie, FR

(72) Inventor/es:

BOYADJIS, BENOIT;
BERGERON, CYRIL y
LECOMTE, SÉBASTIEN

(74) Agente/Representante:

CARPINTERO LÓPEZ, Mario

DESCRIPCIÓN

Procedimiento de generación de un vector de inicialización para el cifrado de un contenido de vídeo

La invención se refiere a un procedimiento y un dispositivo que permite principalmente generar un vector de inicialización único y que puede sincronizarse que, combinado con una clave de cifrado, permite el cifrado de un flujo de datos de vídeo comprimidos y el descifrado a partir de una trama cualquiera del flujo de datos después del filtrado. Se aplica en el marco del cifrado selectivo, en unas aplicaciones de procesamiento de flujos de vídeo comprimidos, en unas aplicaciones de seguridad de los datos de vídeo comprimidos por cifrado selectivo. El cifrado selectivo es una técnica de cifrado que no cifra la totalidad del mensaje o de un flujo de datos de vídeo comprimido a transmitir.

En un sistema de cifrado, existen, entre otras, dos problemáticas a tener en cuenta: por una parte la transmisión con seguridad de la clave de cifrado y por otra parte la gestión del vector de inicialización. La clave de cifrado y el vector de inicialización son necesarios para la construcción, mediante un generador pseudoaleatorio (GPA) de tipo bloque, de una secuencia pseudoaleatoria que permita controlar el cifrado. Los mecanismos que garantizan un intercambio con seguridad de la clave de cifrado son conocidos para el experto en la materia. Igualmente, el experto en la materia conoce los modos de operación genéricos para la gestión de los mensajes claros y cifrados en el seno de un algoritmo de cifrado por bloques. Con referencia al vector de inicialización, y desde un punto de vista de seguridad, es importante no utilizar el mismo vector de inicialización para el conjunto del flujo de datos con el fin de evitar y/o de minimizar unos ataques eventuales. Además, es esencial gestionar la sincronización de una modificación de este vector de inicialización en casos de pérdidas eventuales de datos o de errores que puedan acaecer en el descifrado. Este procedimiento de sincronización no debe generar latencias suplementarias en el procesamiento de los datos, ni necesitar el envío de datos suplementarios (aumento de la velocidad de la comunicación).

La Patente US 8160157 del presente Solicitante describe un procedimiento de cifrado selectivo que selecciona unas palabras de código para realizar el cifrado selectivo del flujo de datos.

La publicación de Z.Shahid titulada “Fast Protection of H.264/AVC by selective encryption of CAVLC and CABAC for I and P frames”, IEEE transactions on CSVT, 2011 describe un procedimiento de cifrado y descifrado que implica, en caso de error de transmisión, una desincronización prohibitiva del cifrado del flujo.

El cifrado selectivo permite proteger un flujo de vídeo comprimido de manera no detectable mediante un análisis de dicho flujo. En efecto, el procedimiento garantiza la conformidad del flujo de cifrado con el formato del flujo inicial. El cifrado selectivo no debe por otra parte necesitar ninguna transmisión de información complementaria con el fin de garantizar la discreción de su aplicación; prueba suplementaria de seguridad.

El documento US 2005/0207580 se refiere a un procedimiento y un sistema de sincronización en el que se asocia un emplazamiento de memoria a un dato cifrado.

La publicación de Danang Tri Massandy et ál., XP032275867, “Secured video streaming development on smartphones with Android platform” se refiere a un procedimiento de cifrado de datos basado en el algoritmo estándar AES y en el protocolo de transporte utilizado para la transmisión de datos, el protocolo RTP.

El documento de Zhuo Wei et ál., titulado “A scalable and format-compliant encryption scheme for H.264/SVC bitstreams”, XP055211540, enseña a utilizar un generador de claves para cifrar un flujo de datos.

Las soluciones descritas en la técnica anterior conocida por el presente solicitante no toman verdaderamente en cuenta la especificidad del cifrado selectivo:

- Necesitan la transmisión de un vector de inicialización como complemento del flujo de datos comprimido cifrado si se modifica este vector de inicialización,
- No permiten identificar unos puntos de resincronización adaptados a la aplicación del citado selectivo sobre unos flujos de vídeo comprimidos en tiempo real; no está soportada por tanto la resincronización del descifrado en caso de pérdida de datos,
- No proponen una arquitectura adaptada a la transmisión inalámbrica, por ejemplo sobre un soporte en tiempo real RTP (Real-Time Transport Protocol),
- No es posible con los procedimientos actuales comenzar el descifrado a partir de una trama cualquiera del vídeo comprimido,
- No proponen solución para una utilización en tiempo real sobre un enlace de transmisión con pérdidas o errores.

En lo que sigue de la descripción, se utilizarán las abreviaturas siguientes:

AVC: codificación de vídeo avanzada o en inglés Advanced Video Coding,
SVC: codificación de vídeo adaptable o en inglés Scalable Video Coding,
HEVC: codificación de vídeo de alta eficacia o en inglés High Efficiency Video Coding,
NALU: Unidad de capa de abstracción de red o Network Abstraction Layer Unit,
VCL: capa de codificación de vídeo o Video Coding Layer

RTP: protocolo de datos sometidos a restricciones de tiempo o Real-time Protocol Transport,
MMTP: protocolo de transporte de datos multimedia MPEG o MPEG Media Transport Protocol,

La descripción utilizará las siguientes definiciones:

Secuencia para designar una serie de bits.
Semilla = secuencia utilizada para inicializar una herramienta de generación de series de bits o claves,
Contador = secuencia que representa una variable incrementada con un paso fijo en cada utilización,
Vector de inicialización (IV) = secuencia utilizada a la entrada de un generador pseudoaleatorio (o bloque de cifrado) como complemento de una clave de cifrado para producir una secuencia de cifrado,
Clave de cifrado = secuencia compartida utilizada para el cifrado y el descifrado del mensaje,
Secuencia de cifrado = secuencia pseudoaleatoria, generada por un bloque de cifrado, utilizada para cifrar una secuencia del flujo a cifrar,
Secuencia a cifrar = secuencia procedente de un mensaje a transmitir y destinada a ser protegida mediante una operación de cifrado,
Secuencia cifrada = secuencia obtenida mediante una operación XOR entre una secuencia de cifrado y una secuencia a cifrar,
Generador de IV = función que permite generar un vector IV de inicialización a partir de la semilla.
Generador pseudoaleatorio (o bloque de cifrado) GPA = función que permite generar una serie de cifrado a partir de una clave de cifrado y de un vector IV de inicialización,
Operador XOR = operador lógico o exclusiva,

Unidad autónoma Ui para designar de manera genérica una parte del flujo de vídeo en el que la semilla permanece constante.

La invención se refiere a un procedimiento de sincronización de un mecanismo de cifrado y de descifrado de datos contenidos en un flujo Fc de vídeo comprimido constituido por datos cifrables y datos no cifrables, en el seno de un sistema de transmisión de seguridad que utiliza un protocolo de transporte y un modo de codificación de los datos, caracterizado porque incluye al menos las etapas siguientes:

a) Extraer las informaciones protocolarias vinculadas al protocolo de transporte en tiempo real RTP con el fin de generar una semilla K o secuencia de bits, utilizando el identificador de sincronización de origen SSRC contenido en un encabezado RTP, un parámetro representativo del instante de presentación relativa de una NALU contenida en el paquete RTP, el orden de presentación de las imágenes con relación a una imagen de referencia de una agrupación de imágenes o GOP en curso de codificación, el número de la capa de mejora, y la dirección de comienzo de la “slice” en el seno de una imagen,
b) Generar un vector IV de inicialización para el cifrado a partir de la semilla K y actualizar el vector de inicialización con cada cambio de uno de los elementos constitutivos de la semilla K,
c) Especificar un intervalo de aplicación del vector IV de inicialización sobre el flujo de datos y cifrar los elementos cifrables utilizando una secuencia de cifrado procedente de un generador pseudoaleatorio que tiene en cuenta una clave de cifrado y el vector IV de inicialización generado en la etapa b).

La invención se refiere también a un procedimiento de sincronización de un mecanismo de cifrado y de descifrado de datos contenidos en un flujo Fc de vídeo comprimido constituido por datos cifrables y datos no cifrables, en el seno de un sistema de transmisión de seguridad que utiliza un protocolo de transporte y un modo de codificación de datos, caracterizado porque incluye al menos las etapas siguientes:

d) Extraer las informaciones protocolarias vinculadas al protocolo de transporte MMTP con el fin de generar una semilla K o secuencia de bits, utilizando un identificador del paquete “packet_id”, un identificador “packet_sequence_number” y un parámetro “timestamp” representativo del instante de presentación, contenidos en un encabezado MMTP, el orden de presentación de las imágenes con relación a la imagen de referencia de una agrupación de imágenes o GOP en curso de codificación, el número de la capa de mejora y la dirección de comienzo de la “slice” en el seno de la imagen,
e) Generar un vector IV de inicialización para el cifrado a partir de la semilla K y actualizar el vector de inicialización con cada cambio de uno de los elementos constitutivos de la semilla K,
f) Especificar un intervalo de aplicación del vector IV de inicialización sobre el flujo de datos y cifrar los elementos cifrables utilizando una secuencia de cifrado procedente de un generador pseudoaleatorio que toma en la entrada una clave de cifrado y el vector IV de inicialización generado en la etapa b).

El flujo de datos puede componerse de varias unidades y cuando el número de elementos a cifrar contenidos en una unidad del flujo de vídeo es superior al tamaño de la secuencia de cifrado generada por el generador de pseudo-aleatoriedad, GPA, el procedimiento genera en un primer tiempo una primera secuencia de cifrado a partir de un primer vector de inicialización, y posteriormente en un segundo tiempo genera una nueva secuencia de cifrado con un modo de operación criptográfico.

El algoritmo de cifrado utilizado para la implementación del procedimiento es el modo AES.

El modo de operación de cifrado elegido puede ser el modo contador (CTR), y se asigna un campo contador de 16

bits inicializado a cero.

En el caso de una aplicación H.264/AVC-SVC o HEVC/SHVC, se introduce un punto de resincronización en cada inicio de la NALU, siendo desencadenado el punto de resincronización por un cambio de semilla.

En la inicialización de una comunicación con un protocolo RTP, el procedimiento incluye, por ejemplo, una etapa de inicialización de los parámetros “timestamp” y SSRC aparte de una etapa de colocación en paquetes de los datos de vídeo.

En el curso de la comunicación que utiliza un protocolo RTP, el procedimiento genera, por ejemplo, una aleatoriedad suplementaria sobre un número de bits de peso reducido del “timestamp”.

En la inicialización de una comunicación con un protocolo MMTP, el procedimiento puede incluir una etapa de inicialización de los parámetros “timestamp”, “packet_sequence_number” y “packet_id” aparte de una etapa de colocación en paquetes de los datos de vídeo.

En el curso de la comunicación que utiliza un protocolo MMTP, el procedimiento incluye, por ejemplo, una etapa de generación de una aleatoriedad suplementaria sobre un número de bits de peso reducido del “timestamp”.

Según una variante de implementación, el modo de operación es el modo CTR y se genera también una aleatoriedad sobre al menos una parte del campo reservado al contador.

Según una variante de realización, el vector de inicialización se genera sobre 128 bits.

La invención se refiere también a un dispositivo de generación pseudoaleatoria de elementos de cifrado y de punto de sincronización para un flujo de datos de vídeo que puede descomponerse en varias unidades autónomas caracterizado porque incluye al menos los elementos siguientes:

- Un módulo de extracción de elementos no cifrables del flujo de datos (mensaje de vídeo comprimido e informaciones de nivel superior) con el fin de constituir una semilla K según las etapas del procedimiento de la invención,
- Un módulo de generación de vectores de inicialización, que utiliza una semilla K para generar una secuencia de bits, caracterizado porque el módulo de generación de IV está adaptado para desencadenar un punto de resincronización cada vez que cambia la semilla utilizada, y porque el vector IV de inicialización se recalcula en cada punto de resincronización.
- Un generador pseudoaleatorio GPA adaptado para generar una secuencia de cifrado utilizando un vector de inicialización y una clave de cifrado.

El algoritmo de cifrado es, por ejemplo el algoritmo AES.

El modo de operación utilizado en el dispositivo es, por ejemplo, el modo CTR, y el protocolo de transporte es el protocolo RTP.

Según otra variante, el modo de operación utilizado es el modo CTR, y el protocolo de transporte es el protocolo MMTP.

Surgirán mejor otras características y ventajas de la presente invención con la lectura de la descripción que sigue, de ejemplos de realización, anexada por las figuras que representan:

- la figura 1, un recordatorio de la colocación en paquetes de las tramas de vídeo por el protocolo RTP,
- la figura 2, el desarrollo de las etapas que permiten el cifrado de una secuencia,
- la figura 3, un ejemplo de esquema para el servidor de cifrado,
- la figura 4, un ejemplo de esquema para un cliente descifrador sincronizado,
- la figura 5, un ejemplo de dispositivo de descompresión de un flujo de datos,
- la figura 6, un ejemplo de implementación del vector de inicialización, y
- la figura 7, un esquema que describe un ejemplo de etapas implementadas en el procedimiento según la invención.

Con el fin de comprender mejor el procedimiento implementado en la presente invención, se da el ejemplo que sigue a título ilustrativo y en ningún caso limitativo en el caso de un flujo de vídeo protegido mediante cifrado selectivo con el algoritmo de cifrado AES-CTR y transmitido sobre un enlace RTP.

Se realiza un recordatorio en relación con la figura 1. Las normas conocidas por el experto en la materia H.264 (AVC/SVC) y HEVC (SHVC) se destinan a múltiples utilizaciones tales como la difusión (cable, satélite, modulador/demodulador o módem,…), el almacenamiento, el vídeo bajo demanda, etc. Para paliar esta variedad de las aplicaciones objetivo, las normas especifican la noción de unidades de base independientes y autónomas o NALU. Cada unidad de NAL 4 está así dotada con un encabezado 5 o NAL Header, cuyas informaciones pueden utilizarse por unos protocolos de más alto nivel y de datos o RSBP 6. Coexisten dos tipos de NALU 8: las unidades VCL 1 que contienen los datos de vídeo codificados 2 y las unidades no VCL que contienen las informaciones necesarias para

el parametrizado del decodificador. La figura 1 ilustra el paso de la capa de codificación de vídeo VCL al protocolo RTP.

El empaquetado del flujo de vídeo comprimido mediante el protocolo RTP se basa en la colocación en paquetes de aproximadamente 1400 octetos correspondientes frecuentemente a una única NALU del flujo de vídeo de origen. Cada paquete RTP, 9, está dotado de un encabezado 10 (RTP header) constituido por informaciones que permiten la reconstrucción del flujo: el origen de la emisión (synchronization source identifier), el número de paquete (sequence number), el instante de presentación de la NALU empaquetada (timestamp), etc. y de datos 11. Unas normas específicas, tales como RFC 6184 para H.264, detallan la implementación del protocolo RTP para los flujos de datos multimedia.

Entre los procedimientos de protección de datos, el cifrado selectivo es un procedimiento que no modifica más que ciertos elementos del flujo cifrado con el fin de impedir la reconstrucción de la imagen a unos usuarios no autorizados (contenido protegido por degradación de la imagen) mientras conserva el formato y la sintaxis del flujo, permitiendo su legibilidad por no importa qué decodificador. Una manera de cifrar el flujo puede ser la descrita en la publicación de B.Boyadjis et ál., “A real-time ciphering transcoder for H.264/AVC and HEVC streams”, IEEE ICIP, proceedings 2014, para la transcodificación de datos comprimidos.

En el ejemplo que se va a dar a título ilustrativo y en ningún caso limitativo, el algoritmo de cifrado propuesto utiliza la versión modificada del AES-CTR. La norma AES-CTR (Advanced Encryption Standard Counter Mode) es ampliamente utilizada hoy en día por su simplicidad de utilización y su robustez. La AES es un cifrado de bloques que tiene como entrada una clave de cifrado y un vector IV de inicialización. La clave de cifrado es de 128, 192 o 256 bits, el vector IV de inicialización es una secuencia pseudoaleatoria de 128 bits. El modo de operación CTR específica un medio para generar una serie de secuencias de cifrado mediante la utilización sucesiva de diferentes vectores de inicialización. La principal debilidad conocida del AES en modo CTR es que en caso de reutilización del vector de inicialización, ya no puede asegurarse la seguridad de los mensajes. De ese modo, el procedimiento recomendado generalmente para implementar este modo de cifrado es utilizar una semilla regularmente incrementada. El AES proporciona, a partir de la clave de cifrado y del vector de inicialización, una secuencia pseudoaleatoria de 128 bis, denominada secuencia de cifrado en el marco de la presente patente. Con el fin de operar las modificaciones de bits en el seno de un flujo de datos, cada bit de la secuencia de cifrado se combina mediante una operación XOR, por ejemplo, con un bit original del flujo a cifrar.

Cuando se consume totalmente la secuencia de cifrado, se incrementa el vector de inicialización para producir un nuevo vector de inicialización, que permita la generación de una nueva secuencia de cifrado de 128 bits. En el marco de un envío sobre un canal de pérdida de flujo protegido por un procedimiento de ese tipo, la pérdida de datos puede provocar la desincronización del incremento e impedir el descifrado de los datos. Es necesario un procedimiento de compartición o sincronización de los vectores de inicialización. Ventajosamente, su implementación no necesita el envío de datos suplementarios en el seno del protocolo RTP.

La figura 2 esquematiza las etapas para el cifrado de una secuencia de datos. Se extraen las informaciones protocolarias vinculadas al protocolo de transporte y las informaciones no cifrables del flujo Fc de datos, con el fin de generar una semilla K o secuencia de bits. Se genera un vector IV de inicialización (128 bits) por un generador 20 del vector de inicialización a partir de la semilla K, el vector de inicialización se actualiza con cada cambio de uno de los elementos de la semilla K. Se especifica un intervalo de aplicación del vector IV de inicialización y se cifran los elementos cifrables utilizando una secuencia de cifrado Sc procedente de un generador 21 pseudoaleatorio que toma en la entrada una clave kp de cifrado y el vector IV de inicialización generado en la etapa b). La secuencia de cifrado Sc se utiliza para cifrar una secuencia F de datos a cifrar en el seno de un módulo operador XOR, 22, para obtener una secuencia Fc de cifrado. Una semilla está constituida a partir de elementos no cifrables y no cifrados extraídos del flujo de datos y de extracciones del encabezado de la capa de transporte del protocolo de transmisión utilizado; estos elementos pueden combinarse mediante cualquier operación del tipo concatenación, adición, XOR.

Las figuras 3 y 4 representan unos esquemas de servidor 30 de cifrado y de cliente 40 descifrador.

La figura 3 ilustra un ejemplo de arquitectura de un servidor de cifrado del flujo de datos Fv, F_{RTP}. Un servidor 30 de cifrado recibe en una o varias entradas 31, un flujo de datos no comprimidos Fnc recuperado por ejemplo en tiempo real de un captador de tipo cámara (no representado en la figura), que transmite a un módulo 32 de compresión y cifrado selectivo adaptado para cifrar los datos del flujo utilizando una clave kp de cifrado y un vector IV de inicialización. El flujo Fcc de datos comprimidos y cifrados se transmite a un módulo 33 de colocación en paquetes RTP antes de transmitirse a través de la salida 34 sobre una red de transmisión de flujos de vídeo no representada por razones de simplificación, el IV se transmite implícitamente. El vector IV de inicialización se genera por medio de un generador 34 del vector IV de inicialización, que toma en la entrada una semilla constituida a partir de informaciones locales, y/o de nivel superior. El generador de IV construye un vector IV de inicialización que se utiliza por el algoritmo de cifrado, como se detalla a continuación, por ejemplo.

La figura 4 ilustra un ejemplo de arquitectura de un sistema 40 del lado de recepción. Un cliente que va a descifrar el flujo Fc de vídeo codificado incluye un primer módulo 42 que desempaqueta los datos del flujo recibido en una entrada 41; el flujo comprimido y protegido se transmite a un módulo 43 de descompresión y de cifrado selectivo,

mientras que los datos vinculados a la construcción de la semilla se transmiten al generador de IV 44, el flujo de datos se descomprime y descifra según el algoritmo de cifrado selectivo que recibe también la clave de cifrado Pk utilizada en la codificación y que proporcionará un flujo Fv, F_{RTP}, 45, de datos de vídeo decodificados y descifrados.

La figura 5 ilustra un ejemplo de dispositivo que realiza una transcodificación y un cifrado selectivo 50. Se transmite un flujo Fc comprimido a un módulo 51 de transcodificación y de cifrado selectivo que recibe una clave de cifrado y un vector IV de inicialización proporcionado por un generador 53 de IV. El flujo transcodificado y cifrado/descifrado de manera selectiva se envía después hacia una red de transmisión no representada. En el caso de un flujo RTP, el dispositivo 50 incluirá un módulo 55 de desempaquetado antes de la transmisión al módulo de transcodificación y cifrado selectivo y un módulo 56 de colocación en paquetes en el formato RTP.

La figura 6 de un ejemplo de elementos constitutivos del vector de inicialización tal como se define en la presente implementación de la invención. Para construir un vector de inicialización, IV, el procedimiento generará una semilla K únicamente a partir de los elementos apropiados de la capa protocolaria, con o sin utilizar unas informaciones no cifradas y no cifrables del mensaje de vídeo comprimido.

Un ejemplo posible de elección de los elementos constitutivos de la semilla es por tanto el siguiente:

61: SSRC (32 bits); elemento que representa la información del identificador de sincronización de origen, que está contenido en el encabezado RTP y que sirve para identificar el emisor de la comunicación.
62: Timestamp (32 bits); información que está contenida en el encabezado RTP; el timestamp representa el instante de presentación relativo de la NALU contenida en el paquete RTP.
63: POC (16 bits): información vinculada al codificador/decodificador CÓDEC, POC designa la noción de Pic Order Count. El POC designa el orden de presentación de las imágenes con relación a la imagen de referencia I(DR) de la agrupación de imágenes (GOP) en curso de codificación.
64: Layer_ID (16 bits); información vinculada al codificador-decodificador o CÓDEC, el Layer_ID da, para las extensiones “escalables” de las normas AVC (SVC) y HEVC (SHVC), el número de la capa de mejora a la que pertenece la NALU en curso de procesamiento.
65: ST_Addr (16 bits); información vinculada al CÓDEC, ST_Addr designa la dirección (es decir la posición con relación a la imagen global) del comienzo de la trama más conocida bajo el término inglés “slice” en el seno de la imagen.
66: Int_Count (16 bis); contador interno o espacio reservado del vector de inicialización para la aplicación del modo de operación CTR en el seno de una imagen. La sincronización de este campo contador entre cliente y servidor está vinculada directamente a la simetría de ciertos dispositivos de transcodificación tal como se describe en la publicación antes citada de Boyadijis et ál. El contador se dimensiona de manera que garantice el procesamiento de los datos entre dos puntos de sincronización.

Según un ejemplo de realización, el módulo 20 de generación de IV, realiza una concatenación de los elementos anteriores. La arquitectura propuesta para la construcción de la semilla identifica la parte reservada a las informaciones vinculadas a la imagen y la parte reservada al protocolo de transporte. De ese modo, el procedimiento puede adaptarse fácilmente a otras capas de transporte. Se asociarán los primeros bits del encabezado del protocolo de transporte a esta parte reservada del IV con el fin de implementar el procedimiento según la invención, 64 bits en este ejemplo.

La presente invención designa un punto de resincronización Prs como un evento desencadenado por un cambio de la semilla. Un punto de resincronización reinicializa el vector IV de inicialización: se constituye la nueva semilla, y se genera el IV correspondiente (campo contador Int_Count repuesto a cero). Una construcción de ese tipo permite satisfacer la condición de unicidad de cada vector IV de inicialización, con la única condición de que la semilla construida sea única. La presente invención aprovecha esta propiedad construyendo una semilla única K(NALU) para cada NALU del flujo Fcp comprimido protegido. En el interior de una NALU, si la secuencia de cifrado generada con el vector de inicialización así construido se consume totalmente, se genera un nuevo vector IV de inicialización incrementando el contador C, y se produce una nueva secuencia de cifrado.

En el ejemplo explicado anteriormente, es necesario que cada NALU VCL transcodificada disponga de su vector IV de inicialización (NALU VCL) específico y único. Si dos NALU VLC no forman parte de la misma imagen, su marca de tiempos o en inglés el timestamp es diferente, salvo en caso de cierre evocado un poco más adelante en la descripción, y de que el vector de inicialización sea distinto. En el caso en el que dos NALU VCL pertenecen a la misma imagen, tienen el mismo timestamp y se presentan dos casos. Es decir la imagen se descompone en tramas o slices en inglés/ segmentos/tejas y la información de ST_Addr permite asociar a cada NALU su propio vector de inicialización. La otra posibilidad es la presencia de una o varias capas de mejora, en el caso de la información de capa de información o Layer_ID asociada a cada capa de mejora de los vectores de inicialización distintos. En el caso de la tecnología MVC, codificación de vídeo multivía, se utilizará el campo view_Id conocido por el experto en la materia.

Según otra variante de realización, el flujo de datos es un flujo de datos de vídeo comprimido que puede descomponerse en varias unidades autónomas NALU, el protocolo de transporte es el protocolo MMTP, y el procedimiento construirá una semilla K utilizando el identificador de paquete “packet_id”, el identificador

"packet_sequence_number" un parámetro "timestamp" representativo del instante de presentación de contenidos en un encabezado MMTP, el orden de presentación de las imágenes con relación a una imagen de referencia de una agrupación de imágenes o GOP en curso de codificación, el número de la capa de mejora, y la dirección de inicio de la "slice" en el seno de la imagen. En la etapa de inicialización de una comunicación con un protocolo MMTP, el procedimiento incluye una capa de inicialización de los parámetros "timestamp", "packet_sequence_number" y "packet_id" aparte de una etapa de colocación en paquetes de los datos de vídeo. El procedimiento puede generar también una aleatoriedad suplementaria sobre un número de pistas de peso reducido del "timestamp".

La figura 7 esquematiza un ejemplo de etapas implementadas por el procedimiento según la invención.

Uno o varios símbolos procedentes de la unidad de vídeo y/o del transporte se extraen del flujo F_0 de vídeo no cifrado. Los elementos o símbolos extraídos se determinan por ejemplo utilizando una tabla preestablecida que reagrupa los elementos de flujo que no pueden cifrarse, B_{NC}. Estos elementos 70, 71, a priori no cifrables y extraídos, los completa el procedimiento mediante un contador Count 72. Por ejemplo, el procedimiento concatenará los elementos poniéndoles uno a continuación del otro. El conjunto constituido por los elementos B_{NCS} seleccionados y por el contador Int_Count se transmite a un generador 73 de pseudo-aleatoriedad GPA que genera la una secuencia 77 de cifrado (uno o varios elementos de cifrado). La secuencia de cifrado se consume progresivamente por la extracción, para cada bit que puede cifrarse mediante cifrado selectivo del flujo de vídeo, 76 de un bit utilizado de la manera siguiente: bit_cifrado = bit_cifrable XOR bit_extraído. Cuando está vacía la secuencia de cifrado pseudoaleatoria inicialmente generada, se incrementa el vector de inicialización y se genera una nueva secuencia pseudoaleatoria. Esto llega por ejemplo, cuando el número de los bits que pueden cifrarse es mayor que la capacidad del generador de pseudo-aleatoriedad como se ilustra en la parte derecha de la figura. En un primer tiempo el procedimiento genera un primer vector IV_1 de inicialización para los N primeros bits cifrables como se acaba de describir, utilizando en el ejemplo dos símbolos extraídos del flujo en claro, 80, 81, y posteriormente generará un segundo vector IV_2 de inicialización utilizando los mismos dos símbolos extraídos 80, 81 e incrementando el contador haciéndolo pasar de cero a uno, 82. Se produce así un conjunto de símbolos de cifrado, eventualmente no totalmente consumido. Esto permite producir dos vectores de inicialización distintos el seno de una misma unidad autónoma de codificación.

La arquitectura elegida para generar un vector de inicialización según la invención permite, durante la selección del vector de inicialización, no utilizar el bucle de retorno para el mantenimiento de la sincronización. En efecto, el SSRC no evoluciona en el seno de una sesión RTP y el timestamp, iniciado aleatoriamente, es precalculable a continuación, no depende más que de la frecuencia de presentación del vídeo. De esa manera, es posible precalcular la semilla para una trama (o un conjunto de tramas), e iniciar el cifrado a partir de no importa qué NALU de flujo comprimido.

El procedimiento según la invención se implementa, por ejemplo, o bien utilizando un codificador de vídeo modificado, para cifrar el contenido y un decodificador de vídeo modificado en paralelo, o utilizando un transcodificador que realiza la operación de cifrado.

El procedimiento según la invención puede compartirse entre dos modos de funcionamiento, correspondiendo un primer modo de funcionamiento a la inicialización de la comunicación y un segundo modo de funcionamiento al régimen de funcionamiento estándar una vez instalada la comunicación.

En el primer modo de funcionamiento, en la inicialización, el servidor ha recibido un flujo de datos no comprimido, que debe retransmitir comprimido y cifrado en la red. La norma RFC 3550 da un ejemplo de generación de una aleatoriedad de ese tipo.

El primer vector IV de inicialización se constituye aparte del módulo de colocación en paquetes. En el ejemplo dado, las dos palabras SSCR y Timestamp de 32 bits se transmitirán a continuación al módulo de colocación en paquetes con el fin de utilizarse tal cual en el encabezado RTP. El resto del procedimiento de inicialización RTP no se modifica.

En el segundo modo de funcionamiento, una vez implementada la comunicación, el SSRC permanece constante hasta final de la sesión. El timestamp cambia con cada nueva imagen, en función de la frecuencia de presentación del vídeo de origen. Siendo precalculable el timestamp, se calculará por ejemplo por separado del módulo de colocación en paquetes con el fin de utilizarse en el transcodificador para la constitución del vector de inicialización.

En el transcurso de una comunicación de larga duración, con el fin de evitar la reutilización del mismo vector de inicialización y por tanto un riesgo de colisión, el procedimiento genera una aleatoriedad suplementaria sobre un número de bits Nf de peso reducido del timestamp, por ejemplo sobre los cinco de peso más reducido del timestamp y/o sobre los 16 bits reservados del contador. Esta aleatoriedad permite incrementar notablemente la robustez del procedimiento contra eventuales ataques y minimiza el riesgo de colisión. Esta variante se aplica a otros protocolos distintos del protocolo RTP.

En el lado del cliente, el procedimiento incluye unas etapas similares a las ejecutadas para el servidor de cifrado, pero más simples. El módulo de "desempaquetado" realiza por un lado la extracción del flujo, H.264/AVC-SVC o HEVC-SHEVC, por ejemplo, y por otro lado la asociación de las informaciones de SSRC y de timestamp

procedentes del encabezado RTP con cada NALU extraída. Estas informaciones permiten reconstruir los vectores de inicialización idénticamente a unos vectores de inicialización utilizados en el servidor de cifrado con el fin de realizar el descifrado simétrico según unas etapas conocidas por el experto en la materia en el dominio de vídeos con cifrado selectivo.

En el caso de pérdidas de paquetes, las informaciones del SSRC y de timestamp se recuperarán en el encabezado RTP del paquete finalmente recibido, y estando contenidas las informaciones vinculadas al CÓDEC necesarias para la reconstrucción del vector de inicialización intrínsecamente en la NALU encapsulada, no hay impacto a la altura del cifrado. De ese modo, el protocolo o procedimiento propuesto permite la sincronización procedente de la estructura misma del protocolo RTP.

En el caso de paquetes agregados AU de tipo STAP abreviatura inglesa de “Single time Aggregation Packet” o MTAP “Multiple time Aggregation Packet”, el procedimiento funciona sin especificación suplementaria. En efecto, incluso si la parte vinculada al protocolo RTP del vector de inicialización es eventualmente idéntica, por ejemplo para los STAP, la parte relativa al CÓDEC del vector de inicialización permite obtener un vector de inicialización único y no equívoco para cada capa NALU VCL.

El procedimiento según la invención se aplica para diferentes tipos de normas de compresión de vídeo, tales como H.264/AVC, H.265/HEVC, SVC, SHVC, MVC, VP8-9, las expresiones “escalables”, etc., en el campo MPEG.

Constituir una semilla según la invención permite no tener que transmitir información complementaria a aquella contenida en el flujo a transmitir o en los encabezados de la capa de transporte que permite la transmisión de este flujo (= transparencia de la transmisión de la semilla). Si evolucionan los elementos mantenidos para constituir la semilla, las semillas se actualizan para crear un punto de resincronización.

El procedimiento según la invención permite así la resincronización de cualquier punto de un vídeo comprimido utilizando unas informaciones vinculadas al contenido de vídeo y sin utilización de información suplementaria para asegurar la sincronización de cifrado/descifrado. Se adapta fácilmente a cualquier tipo de protocolo de transporte utilizado en el campo del vídeo comprimido. Es una solución robusta que no necesita el envío de datos suplementarios en el seno de la sesión RTP, y no genera latencia en el procesamiento del flujo. En caso de pérdidas de datos, el decodificador podrá descifrar correctamente al resto de los datos recibidos debido a la frecuente reactualización de la semilla utilizada por el algoritmo de cifrado, dimensionada para el procesamiento de flujos de vídeo comprimidos por las normas H.264/AVCSVC, HEVC-SHVC, por ejemplo. El procedimiento permite controlar la robustez de transmisión, estando directamente vinculada la frecuencia de los puntos de resincronización a la frecuencia de cambio de los elementos constitutivos de la semilla.

REIVINDICACIONES

1. Procedimiento de sincronización de un mecanismo de cifrado y de descifrado de datos contenidos en un flujo Fc de vídeo comprimido constituido por elementos cifrables y datos no cifrables, pudiendo descomponerse el flujo de datos de vídeo comprimido en varias unidades autónomas NALU, en el seno de un sistema de transmisión de seguridad que utiliza un protocolo de transporte y un modo de codificación de los datos, **caracterizado porque** incluye al menos las etapas siguientes:

a) Concebir una semilla (K), secuencia de bits, únicamente a partir de las informaciones protocolarias vinculadas al protocolo de transporte en tiempo real RTP, utilizando el identificador (61) de sincronización de origen SSRC contenido en un encabezado RTP, un parámetro (62) representativo del instante de presentación relativa de una unidad autónoma NALU contenida en el paquete RTP, el orden (63) de presentación de las imágenes con relación a una imagen de referencia de una agrupación de imágenes o GOP en curso de codificación, el número de la capa de mejora, y la dirección (65) de comienzo de la “slice” en el seno de una imagen,
b) Generar (20) un vector IV de inicialización para el cifrado a partir de la semilla K y actualizar el vector de inicialización con cada cambio de uno de los elementos constitutivos de la semilla K,
c) Especificar un intervalo de aplicación del vector IV de inicialización sobre el flujo de datos y cifrar (22) los elementos cifrables utilizando una secuencia de cifrado Sc procedente de un generador (21) pseudoaleatorio que tiene en cuenta una clave kp de cifrado y el vector IV de inicialización generado en la etapa b).

2. Procedimiento de sincronización de un mecanismo de cifrado y de descifrado de datos contenidos en un flujo Fc de vídeo comprimido constituido por elementos cifrables y datos no cifrables, pudiendo descomponerse el flujo de datos de vídeo comprimido en varias unidades autónomas NALU, en el seno de un sistema de transmisión de seguridad que utiliza un protocolo de transporte y un modo de codificación de datos, **caracterizado porque** incluye al menos las etapas siguientes:

a) Concebir una semilla (K), secuencia de bits, únicamente a partir de las informaciones protocolarias vinculadas al protocolo de transporte, MMTP, y utilizando el identificador del paquete “packet_id”, el identificador “packet_sequence_number” y un parámetro “timestamp” representativo del instante de presentación contenidos en un encabezado MMTP, el orden de presentación de las imágenes con relación a la imagen de referencia de una agrupación de imágenes o GOP en curso de codificación, el número de la capa de mejora y la dirección de comienzo de la “slice” en el seno de la imagen,
b) Generar (20) un vector IV de inicialización para el cifrado a partir de la semilla K y actualizar el vector de inicialización con cada cambio de uno de los elementos constitutivos de la semilla K,
c) Especificar un intervalo de aplicación del vector IV de inicialización sobre el flujo de datos y cifrar (22) los elementos cifrables utilizando una secuencia de cifrado Sc procedente de un generador (21) pseudoaleatorio que toma en la entrada una clave kp de cifrado y el vector IV de inicialización generado en la etapa b).

3. Procedimiento según una de las reivindicaciones 1 o 2 **caracterizado porque** el flujo de datos se compone de varias unidades y **porque** cuando el número de elementos a cifrar contenidos en una unidad del flujo de vídeo es superior al tamaño de la secuencia de cifrado generada por el generador de elementos pseudo-aleatoriedad, GPA, el procedimiento genera en un primer tiempo una primera secuencia de cifrado a partir de un primer vector de inicialización, y posteriormente en un segundo tiempo genera una nueva secuencia de cifrado con un modo de operación criptográfico.

4. Procedimiento según una de las reivindicaciones 1 o 2 **caracterizado porque** el algoritmo de cifrado utilizado es el modo AES.

5. Procedimiento según una de las reivindicaciones 1 a 4 **caracterizado porque** el modo de operación de cifrado elegido puede ser el modo contador (CTR), y **porque** se asigna un campo contador de 16 bits inicializado a cero en el proceso de generación del IV.

6. Procedimiento según una de las reivindicaciones 1 a 5 **caracterizado porque** para una aplicación H.264/AVC-SVC o HEVC/SHVC, se introduce un punto de resincronización en cada inicio de la NALU, siendo desencadenado el punto de resincronización por un cambio de semilla.

7. Procedimiento según una de las reivindicaciones 1 a 6 **caracterizado porque** en la inicialización de una comunicación con un protocolo RTP, incluye una etapa de inicialización de los parámetros “timestamp” y SSRC aparte de una etapa de colocación en paquetes de los datos de vídeo.

8. Procedimiento según una de las reivindicaciones 1 a 6 **caracterizado porque** en el curso de la comunicación que utiliza un protocolo RTP, el procedimiento genera una aleatoriedad suplementaria sobre un número de bits de peso reducido del “timestamp”.

9. Procedimiento según una de las reivindicaciones 2 a 6 **caracterizado porque** en la inicialización de una comunicación con un protocolo MMTP, incluye una etapa de inicialización de los parámetros “timestamp”, “packet_sequence_number” y “packet_id” aparte de una etapa de colocación en paquetes de los datos de vídeo.

10. Procedimiento según una de las reivindicaciones 2 a 6 **caracterizado porque** en el curso de la comunicación que utiliza un protocolo MMTP, incluye una etapa de generación de una aleatoriedad suplementaria sobre un número de bits de peso reducido del “timestamp”.

11. Procedimiento según una de las reivindicaciones 8 a 10 **caracterizado porque** el modo de operación es el modo CTR y **porque** se genera también una aleatoriedad sobre al menos una parte del campo reservado al contador.

12. Procedimiento según una de las reivindicaciones anteriores **caracterizado porque** el vector de inicialización se genera sobre 128 bits.

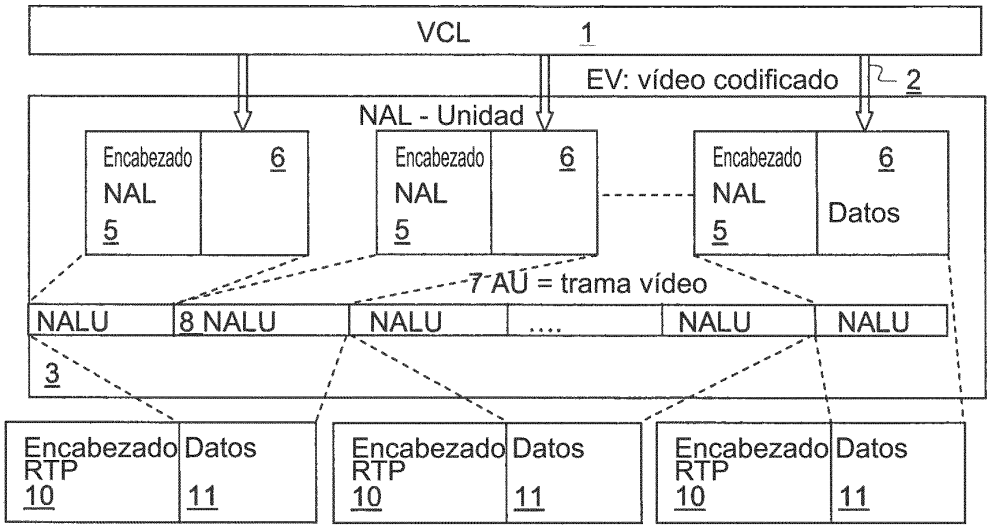
13. Dispositivo generador pseudoaleatorio de elementos de cifrado y de punto de sincronización para un flujo de datos de vídeo que puede descomponerse en varias unidades autónomas **caracterizado porque** incluye al menos los elementos siguientes:

- Un módulo de extracción de elementos no cifrables del flujo de datos (mensaje de vídeo comprimido e informaciones de nivel superior) con el fin de constituir una semilla según una de las reivindicaciones 1 o 2,
- Un módulo de generación de vectores de inicialización, que utiliza una semilla para generar una secuencia de bits, **caracterizado porque** el módulo de generación de IV está adaptado para desencadenar un punto de resincronización cada vez que cambia la semilla utilizada, y **porque** el vector IV de inicialización se recalcula en cada punto de resincronización.
- Un generador pseudoaleatorio GPA adaptado para generar una secuencia de cifrado utilizando un vector de inicialización y una clave de cifrado.

14. Dispositivo según la reivindicación 13 **caracterizado porque** el algoritmo de cifrado es el algoritmo AES.

15. Dispositivo según una de las reivindicaciones 13 o 14 **caracterizado porque** el modo de operación utilizado es el modo CTR, y el protocolo de transporte es el protocolo RTP.

16. Dispositivo según una de las reivindicaciones 13 o 14 **caracterizado porque** el modo de operación utilizado es el modo CTR, y el protocolo de transporte es el protocolo MMTP.



Paquete RTP-9-

FIG.1

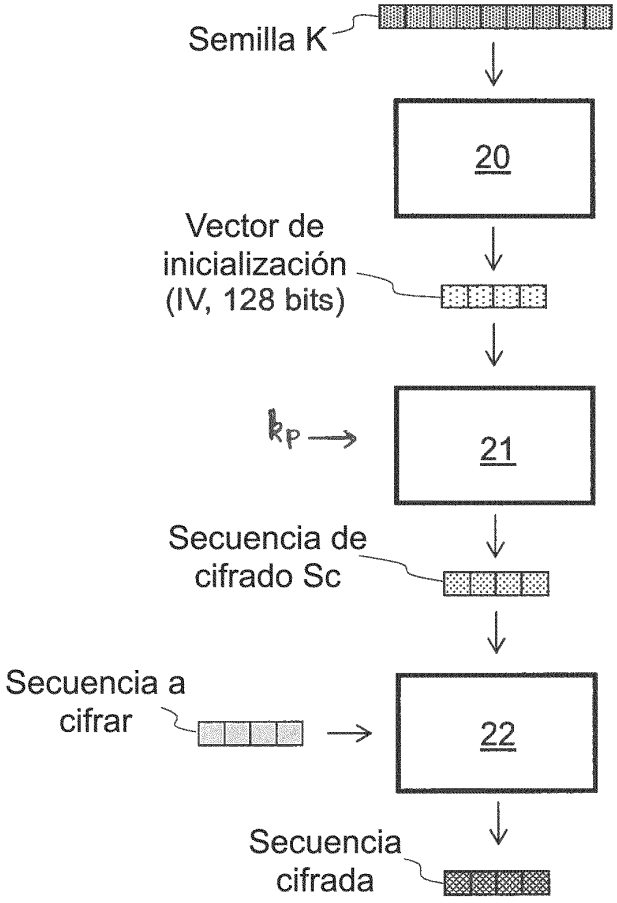
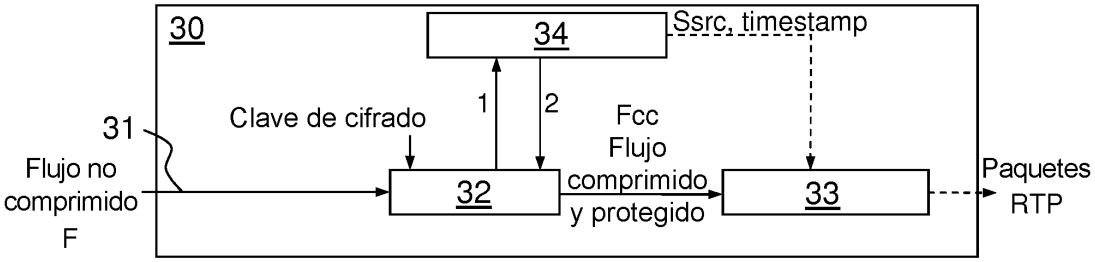
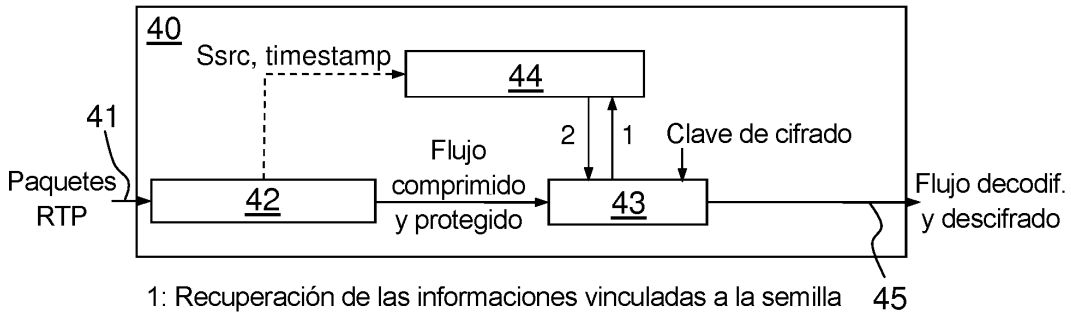


FIG.2



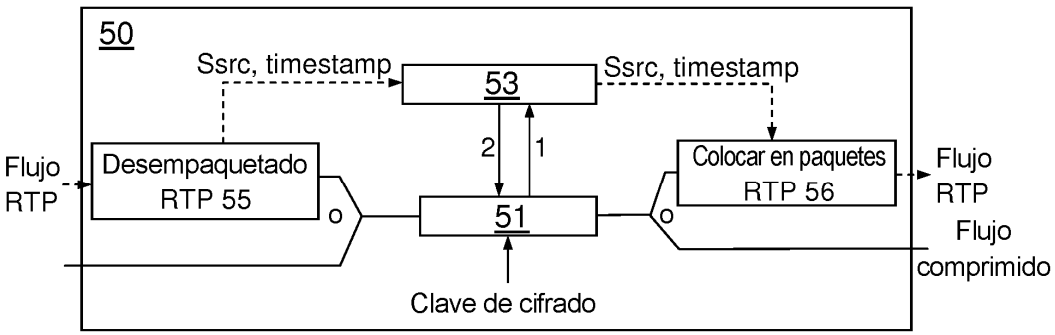
1: Recuperación de las informaciones vinculadas a la semilla
2: IV construido transmitido al algoritmo de cifrado (tipo AES)

FIG.3



1: Recuperación de las informaciones vinculadas a la semilla
2: IV construido transmitido al algoritmo de cifrado (tipo AES)

FIG.4



1: Recuperación de las informaciones vinculadas a la semilla
2: IV construido transmitido al algoritmo de cifrado (tipo AES)

FIG.5

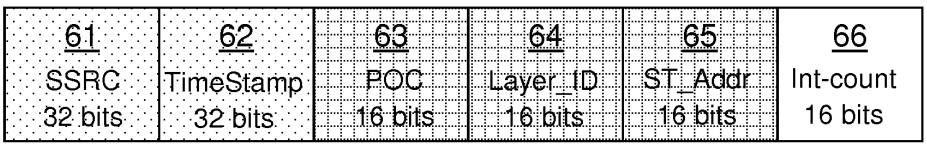


FIG.6

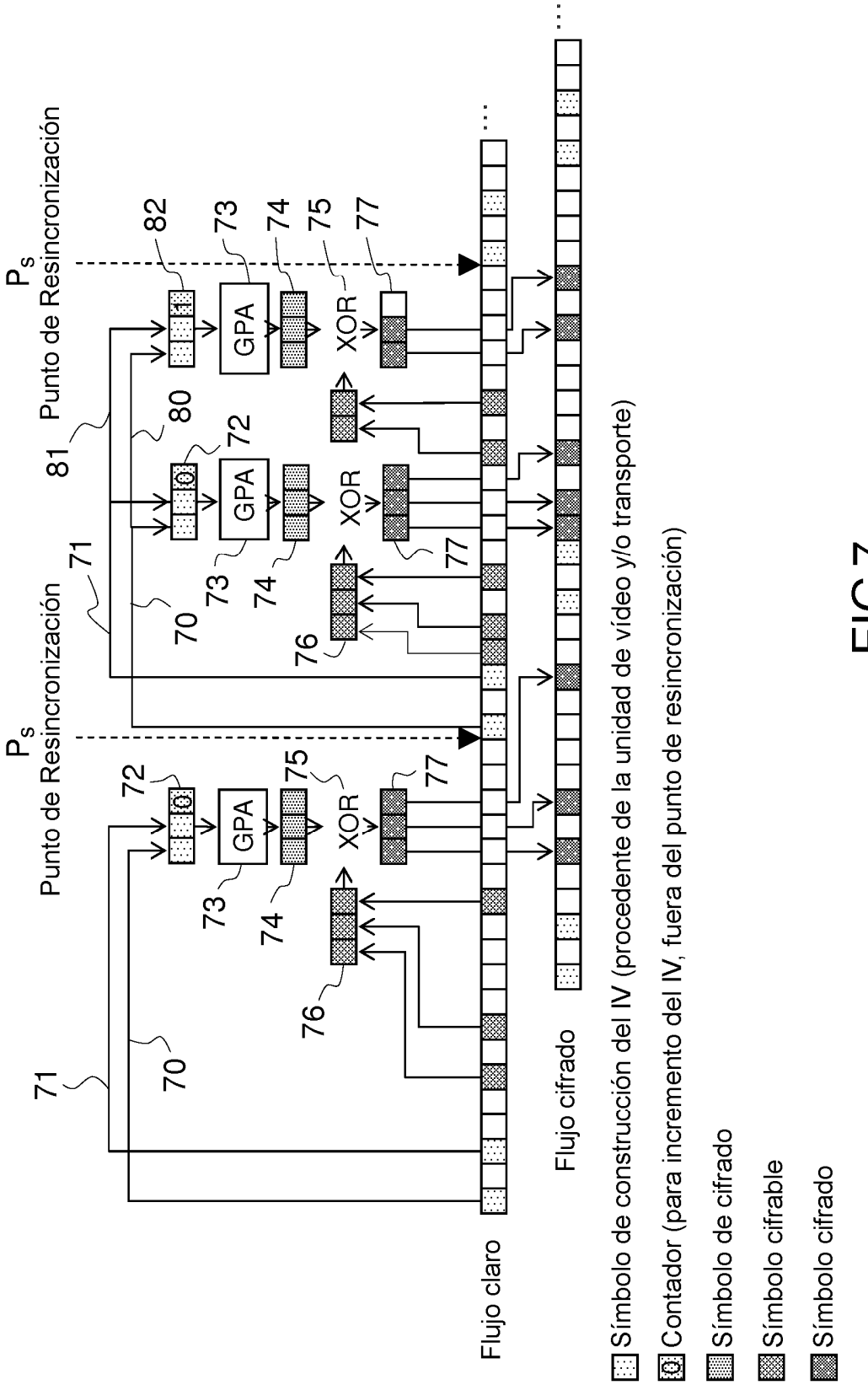

Ps
Punto de Resincronización
Ps
Punto de Resincronización
71
70
72
0
GPA
73
74
75
XOR
76
77
81
80
82
1
Flujo claro
Flujo cifrado
Símbolo de construcción del IV (procedente de la unidad de vídeo y/o transporte)
Contador (para incremento del IV, fuera del punto de resincronización)
Símbolo de cifrado
Símbolo cifrable
Símbolo cifrado

FIG.7