Distribución de claves para contenidos multimedia de acceso condicional

Juan Álvaro Muñoz Naranjo
Dpto. de Arquitectura de Computadores
y Electrónica
Universidad de Almería
Email: jmn843@alboran.ual.es

Leocadio González Casado
Dpto. de Arquitectura de Computadores
y Electrónica
Universidad de Almería
Email: leo@ace.ual.es

***Resumen*—Este trabajo ofrece una revisión del estado del arte en materia de distribución de claves criptográficas, centrándose en los casos en los que éstas se emplean como parte de un sistema de acceso condicional (CAS) en escenarios de broadcasting de contenidos multimedia, tales como la IPTV (televisión vía Internet).**

Se revisará la propuesta inicial elaborada por la International Telecommunication Union (ITU), consistente en una jerarquía de claves de tres niveles y, a partir de ésta, se mostrarán algunos de los esquemas más recientes publicados. Son de especial interés los estudios realizados en Japón y China, donde la comercialización de la DVB (Digital Video Broadcast, la televisión digital) y la IPTV llevan algo de ventaja.

Además, se repasarán otras técnicas auxiliares en materia de acceso condicional e IPTV, como son el empleo de metadatos y autenticación de los clientes.

Finalmente se mostrará el trabajo que el autor y su director están realizando para la empresa IPTV Solutions, con la que el Departamento de Arquitectura de Computadores y Electrónica de la Universidad de Almería mantiene un acuerdo de colaboración.

This paper presents an analysis of the state of the art in cryptographic key distribution management, focusing on conditional access systems (CAS) used in IPTV scenarios.

The initial recommendation by the International Telecommunication Union (ITU), consisting of a three level hierarchy, will be examined and, from this point on, some recent works will be revised. Especially interesting are the researchs comming from Japan and China: these countries are a few years ahead in DVB (Digital Video Broadcast, this is, digital television) and IPTV marketing.

Besides, other complementary techniques regarding conditional access and IPTV, like use of metadata and client authentication, will be considered.

Finally we will present the conditional access system which is being developed by the author and his director for IPTV Solutions, a firm which the Departamento de Arquitectura de Computadores y Electrónica de la Universidad de Almería holds a collaboration agreement with.

***Index Terms*—Sistemas de acceso condicional, distribución de claves, IPTV, DVB**

I. Introducción

Internet ha experimentado un enorme éxito como medio de comunicación: una grandísima parte de los hogares del primer mundo dispone ya de conexión. A día de hoy la red es un estándar para la comunicación, el trabajo y el entretenimiento, especialmente para el sector joven de la población. Si atendemos a la parcela del ocio no cabe duda de que muchas de las posibilidades de diversión a nuestra disposición pasan por Internet: descarga y disfrute de contenidos multimedia, videojuegos en red, chats, etc. La tendencia no es otra que unificar todos los dispositivos de entretenimiento que pueden encontrarse en el hogar en un solo centro de ocio conectado a Internet.

El paradigma de la televisión por Internet, conocido como IPTV, surge como una alternativa avanzada al modelo tradicional. Consiste en el mantenimiento de canales que emiten contenido multimedia a través de la red. ¿Por qué adoptar un nuevo modelo de transmisión cuando el actual está tan extendido y es muy fiable? Esencialmente porque el papel del espectador cambia radicalmente: Internet permite la comunicación bidireccional entre emisor y receptor de forma natural, lo que incrementa en gran medida la libertad de elección de este último. De esta forma, el espectador puede elegir el programa que desea ver en un momento determinado, sin importar su fecha de emisión. Un ejemplo de esto es llegar a casa a las 21.00h y ver una película que terminó a las 20.00h. Por otra parte, también encontramos la posibilidad de hacer "tracking", es decir, desplazarnos a lo largo de la duración del programa, hacia delante (saltando partes) o hacia atrás (viendo de nuevo una escena). Este modo de visionado recibe el nombre de "trick-play", y se opone al tradicional "lineal" al que la televisión nos tiene acostumbrados: reproducción continua y hacia delante del flujo que se recibe, sin posibilidad de elección. A esto podemos añadir otras ventajas, como la posibilidad de almacenar el programa en el dispositivo de forma local, participar activamente y en tiempo real a través de Internet (p. ej: expresar nuestra opinión, bien en el propio programa o con otros espectadores), obtener información ampliada sobre los contenidos que se estén viendo (p. ej: información de archivo relacionada con una noticia de un telediario) o acceder a canales no disponibles en nuestra localidad de residencia.

Esta idea está planteada desde hace años (el concepto IPTV data de 1995), pero su puesta en práctica no ha sido posible hasta hace bien poco, debido principalmente a su necesidad de redes con gran ancho de banda y mínima latencia que lleguen

hasta los hogares. Este escollo ha sido ya superado en buena medida gracias a dos motivos principales: el gran aumento de las prestaciones de los elementos de conmutación IP (routers) y la aparición de forma comercial de técnicas que permiten ajustar y garantizar la calidad del servicio. Este último factor es vital para el tráfico de vídeo, muy sensible a cualquier degradación de las prestaciones de red.

Por lo tanto, si la distribución de radio por Internet es ya una realidad, le llega ahora el momento al vídeo. De esta forma, el modelo de negocio IPTV está dando sus primeros pasos en España, con las empresas de telecomunicación más importantes lanzando sus propias plataformas. Este es el caso de Telefónica con Imagenio, Jazztel con Jazztelia, Ono (Ono TV) y Orange (Orange TV). También ha entrado en el mercado recientemente Superbanda. Yacom fue la primera en ofrecer este servicio, antes de ser comprada por Orange. De estos operadores, los tres primeros disponen de red propia en el país, siendo la de Telefónica la más extendida (cinco millones de accesos de datos, cubriendo prácticamente todo el territorio). Por lo tanto pueden permitirse el lujo de montar el sistema de televisión sobre su propia infraestructura.

En el extranjero las cosas siguen una tendencia similar: Fastweb es el principal operador de Italia y uno de los más importantes de Europa. France Telecom lanzó Maligne TV a principios de 2004, alcanzando un gran éxito. En el Reino Unido la empresa Kingston lanzó en 2000 Kingston Interactive TV, pionera mundial, pero cesó la actividad de ésta en 2006 por su imposibilidad de ampliar geográficamente el negocio. En el continente asiático la IPTV lleva algunos años de ventaja, con Yahoo! Broadband y Korea Telecom como las empresas más importantes.

Las tecnologías necesarias para la implantación a gran escala de la IPTV están aún en una fase muy temprana. Es por ello que existe un gran número de líneas de investigación abiertas en distintos campos: redes, coficación de vídeo y audio, calidad del servicio y protección del contenido. La meta es obtener una tecnología de difusión mundial que sustituya a la televisión tradicional.

La protección del contenido es una prioridad fundamental para las plataformas IPTV, dado que las compañías productoras de contenidos están muy sensibilizadas con el problema de la distribución ilegal. Una plataforma IPTV que quiera tener éxito deberá sin duda ofrecer un mecanismo de protección fiable y seguro para garantizar que los contenidos que emite no quedan gratuitamente a disposición del gran público. Los mecanismos de este tipo reciben el nombre de Sistemas de Acceso Condicional (CAS: Conditional Access System). Los introduciremos en la sección IV. Es preciso no confundir el concepto de acceso condicional con el de DRM (Digital Rights Management). El primero, tradicionalmente, se asocia a la gestión de acceso a emisiones multimedia (por ejemplo, el acceso a un canal de IPTV o DVB). El último se orienta más a la protección de reproducción y distribución de contenido almacenado (el ejemplo más corriente es el de la venta de música vía Internet. El sistema DRM correspondiente se encargará de que sólo el comprador pueda escucharla, evitando así su difusión ilegal). En un futuro no muy lejano, como veremos más adelante, ambas tecnologías deberán unirse para dar lugar a una sola, aplicable en ambas situaciones.

Por ahora, en la sección II trataremos los aspectos técnicos del paradigma IPTV. Debido a que se trata de un servicio reciente es todavía poco conocido.

II. Aspectos técnicos de la IPTV

Un sistema IPTV consiste en un flujo (stream) de información multimedia proporcionado por el proveedor a los clientes. Si el contenido es el mismo para todos, el flujo se envía a una dirección broadcast o multicast. En la modalidad bajo demanda (VoD: Video on Demand) el contenido depende de las peticiones del usuario. Por este motivo cada uno de ellos recibe un flujo diferenciado (conexión unicast).

En cualquier caso el stream consiste en información de vídeo y audio comprimida con un códec. Se trata de minimizar el tamaño del archivo a transmitir, maximizando en la medida de lo posible la calidad.

Como es lógico, en plataformas de este tipo el usuario desea ver el contenido en su pantalla en tiempo real, sin interrupciones ni esperas, con una calidad aceptable. Por lo tanto la principal prioridad es siempre llevar el stream hasta el receptor con la menor latencia posible, evitando también el jitter (fluctuaciones en la reproducción a causa de la variación de la latencia). Dado que el sistema se diseña para atender una gran cantidad de peticiones simultáneas son necesarias redes con la suficiente capacidad, problema que hemos mencionado en la sección anterior. Por este motivo la tecnología IPTV no ha podido comercializarse hasta hace poco (Kingston Interactive TV, la primera televisión IP comercial inició su andadura en 2000, y sólo de forma regional).

En los sistemas IPTV comerciales actuales la red que sustenta el sistema normalmente tiene una arquitectura cliente-servidor distribuida: para evitar el cuello de botella que supondría un solo servidor gestionando peticiones de un gran número de usuarios se instalan servidores locales. Del lado del proveedor de contenidos solemos encontrar los siguientes elementos:

- Servidor de vídeo: almacenan el contenido multimedia a distribuir. Existen servidores locales para servir a los usuarios cercanos. Esta necesidad de replicación incrementa el coste del sistema.
- Sistema de distribución de contenidos: mantiene sincronizados los contenidos en todos los servidores locales, de modo que todos los usuarios puedan tener acceso actualizado a los mismos.
- Sistema de gestión de contenidos: proporciona una interfaz al proveedor mediante la cual éste puede subir los contenidos a los servidores de vídeo y gestionarlos posteriormente. Enlaza con el Sistema de distribución de contenidos.

A estos elementos se añaden otros de más bajo nivel, relacionados principalmente con la gestión del transporte de la información por la red.

De esta forma, el proveedor inserta nuevos programas en parrilla haciendo uso del Sistema de gestión de contenidos. Estos programas son distribuidos a los servidores de vídeo por el Sistema de distribución de contenidos. Finalmente, las peticiones de un aparato receptor son atendidas por el servidor local más cercano, para asegurar una latencia suficientemente baja.

Un aspecto adicional muy importante es el del control de acceso. Un proveedor puede decidir cobrar por el acceso a determinada programación, ya sea un evento, un canal o un paquete de canales. En este escenario es evidente que aquellos usuarios que no han pagado no deberían tener la posibilidad de ver dicho contenido. Por ello los sistemas de control de acceso y control de sesión suelen ser una parte importante de una plataforma IPTV. Los fabricantes dedican gran parte de sus esfuerzos a desarrollar soluciones que permitan autentificar de forma fiable a los usuarios, para poder limitar el acceso sólo a aquellos que reúnan los requisitos necesarios (haber realizado el pago). Además, el stream viaja por la red protegido normalmente por algún tipo de cifrado.

Respecto a los protocolos que se emplean para transmitir el stream, normalmente encontramos los siguientes:

- UDP (User Datagram Protocol): la sencillez de este protocolo de nivel de transporte y su baja sobrecarga lo hace apropiado para transmisiones de vídeo bajo demanda. No garantiza la entrega de todos los paquetes, ni su orden, pero en la transmisión multimedia el factor más importante es la latencia.
- RTP (Real-Time Protocol) y RTCP (Real-Time Control Protocol): RTP se diseñó para la transmisión de datos en tiempo real de extremo a extremo, funcionando sobre UDP. Al igual que éste no garantiza la calidad del servicio ni la seguridad en las transmisiones. RTCP es un protocolo de apoyo para monitorizar sesiones que complementa a RTP ocupándose de questiones como el control de la calidad del servicio (QoS, quality of service).
- RTSP (Real-Time Streaming Protocol): es un protocolo a nivel de aplicación para el control de transmisiones de streams, típicamente multimedia. Permite al cliente enviar al servidor peticiones del estilo "play.º "stop". RTSP puede correr sobre distintos protocolos de nivel inferior, como RTP (sirviéndole de complemento) o directamente sobre UDP.
- MPEG Transport Stream: protocolo de comunicación para audio, vídeo y datos. Multiplexa vídeo y audio para conseguir el sincronismo entre estos. Se emplea normalmente en comunicaciones broadcast y vídeo bajo demanda. Normalmente se encapsula en alguno de los protocolos anteriores.

Aunque el servicio IPTV ya funciona en gran parte del primer mundo, la necesidad de una latencia mínima y un gran ancho de banda para abastecer a un gran volumen de clientes sigue siendo un quebradero de cabeza para las empresas que lo comercializan. Aún peor, el problema crece con el número de usuarios. Por este motivo merecen mención especial los esfuerzos por emplear arquitecturas peer-to-peer en la distribución del flujo. Este paradigma ha comenzado a recibir el nombre de P2P-TV, y surge como solución al problema de la escalabilidad del servicio: cada nodo cliente actúa además como servidor para otros nodos cercanos a él. De esta forma la calidad del servicio aumenta con el número de clientes participantes, justo lo contrario de lo que sucede en las arquitecturas cliente-servidor tradicionales. Es preciso señalar que siempre será necesaria la presencia de un servidor principal que inyecte el flujo multimedia en la red (enviándolo a los nodos clientes con más capacidad, p. ej., para que éstos los comiencen a distribuir). El principal argumento contra este enfoque radica en la situación que encontramos cuando participan pocos clientes en la red: en estos escenarios los esquemas peer-to-peer suelen ofrecer un rendimiento bastante pobre. La solución es inmediata: si el número de clientes que demandan contenido es pequeño el propio servidor principal puede atenderlos hasta que el tamaño de la red aumente.

En otro orden de cosas, no deben confundirse los conceptos IPTV y DVB (Digital Video Broadcast). El segundo implica una distribución del contenido vía satélite y unidireccional, si bien es cierto que el tramo último hasta el usuario (conocido como última milla) puede realizarse mediante conexiones de cable o antena. La desventaja de esta aproximación radica en que la comunicación es unidireccional, con lo que el usuario no tiene la posibilidad de interactuar (tal y como se menciona en la introducción). Como ejemplo de tecnología DVB tenemos la Televisión Digital Terrestre (TDT), en fase de implantación en España. En cualquier caso, la DVB también puede beneficiarse de los sistemas de acceso condicional para expandir su modelo de negocio. La DVB se ha extendido especialmente en Asia desde los años noventa.

En la gran mayoría de los casos tanto la IPTV como la DVB se visualizan en el aparato de televisión gracias a un dispositivo conocido como *set-top box*. Un set-top box recibe la señal digital del medio, la convierte en analógica y la envía al televisor. Los más modernos pueden encargarse de transacciones complejas, tal como control de acceso, aspecto que nos ocupa, acceso interactivo, visionado en alta definición, etc. Algunos también incluyen disco duro para utilizarlo como grabador, o permiten la conexión de dispositivos externos, tales como videocámaras o impresoras. Esta última característica va a influir decisivamente en los modelos de acceso condicional futuros, tal y como veremos en la sección IV-F.

Los set-top boxes normalmente aceptan smart-cards que se emplean para la parte de autenticación del cliente. Son dispositivos similares a tarjetas de crédito, intercambiables, que el proveedor de servicios entrega al cliente como medio de autenticación. Un dispositivo de este tipo suele incluir un microprocesador y una memoria (una parte volátil y una parte no volátil), en la que almacenar la información de autenticación y el software necesario. Como medida de seguridad algunos sistemas CAS graban una identificación del hardware del set-top box en la smart-card, y obligan a que ésta sólo pueda utilizarse en dicho dispositivo. Esta técnica recibe el

nombre de *pairing*. Su ventaja, obvia, es la protección frente al robo de la tarjeta [17].

III. Conceptos básicos de criptografía

Haciendo una revisión de las definiciones y normas proporcionadas por el TCSEC (Trusted Computer System Evaluation Criteria) (E.E.U.U.) y el ITSEC (Information Technology Security Evaluation Criteria) (Europa) encontramos tres criterios o propiedades fundamentales al hablar de seguridad de la información.

- Confidencialidad: es la más directamente relacionada con la criptografía. Se refiere a la ocultación de la información que se comunica entre las partes. A veces, además, es necesario proteger también la identidad de las propias partes.
- Integridad: esta propiedad permite asegurar que no se ha falseado la información.
- Accesibilidad: determina quién y en qué momento puede acceder a la información.

La criptografía es la disciplina que se ocupa de salvaguardar el secreto de la información.

Todo sistema criptográfico o criptosistema consta de cinco componentes:

- El espacio de mensajes $M = \{m_1, m_2, \ldots\}$ que es el conjunto de todos los posibles textos en claro. A los elementos de este conjunto se les denomina mensajes, entendiendo por esto que son inteligibles. Los mensajes se forman a partir de un alfabeto, formado a su vez por caracteres.
- El espacio de cifrados $C = \{c_1, c_2, \ldots\}$. El alfabeto de los cifrados puede ser el mismo o distinto del de los mensajes.
- El espacio de las claves $K = \{k_1, k_2, \ldots\}$.
- Una familia de transformaciones de cifrado $\{E_k : M \rightarrow C\}$.
- Una familia de transformaciones de descifrado $\{D_k : C \rightarrow M\}$.

D_k ha de ser la inversa de E_k, así $m = D_k(E_k(m))$.

Un criptosistema ha de satisfacer algunos requisitos para ser llevado a la práctica:

- Las transformaciones de cifrado y descifrado deben ser computacionalmente eficientes (y no sólo eficaces) para todas las claves, con el fin de no provocar retardos excesivos que distorsionen el funcionamiento del sistema.
- La seguridad del sistema debe depender exclusivamente del secreto de las claves y no del secreto de las funciones D y E. Aún conociéndose E_k o D_k es necesario que no se pueda inferir la clave k de cifrado o descifrado respectivamente (Principio de Kerchoff simplificado, 1883).

III-A. Criptografía de clave simétrica

Los criptosistemas de clave simétrica son también conocidos como de clave única o secreta. Su nombre se debe a que dicha clave es usada tanto para el cifrado como para el descifrado, con lo que ésta ha de ser guardada en secreto puesto que en ella reside su fortaleza.

Normalmente el cifrado de clave secreta se realiza por bloques. Esto quiere decir que la información se divide en "trozos", que pueden cifrarse independientemente o, para mejorar la seguridad, teniendo en cuenta información del bloque anterior. Actualmente suele emplearse un tamaño de bloque de 64 o 128 bits.

Las características principales de estos cifradores pueden resumirse en los siguientes puntos:

- Cada símbolo o elemento del mensaje se cifra de manera dependiente de los adyacentes.
- Independientemente de la posición relativa del bloque dentro del mensaje, cada bloque se cifra con el mismo algoritmo y la misma clave.
- Si dos mensajes iguales se cifran con la misma clave, los resultados son también iguales.

La gran mayoría de cifradores simétricos siguen el llamado "esquema de Feistel": el mensaje se divide en bloques, cada uno de los cuales se cifra por separado mediante técnicas de sustitución y transposición, además de otras operaciones lineales sencillas de adición y multiplicación, durante un número de ciclos llamados "vueltas". Para descifrar se realiza el proceso inverso, interviniendo en ambas etapas la clave simétrica. Estos algoritmos suelen ser bastante eficientes y generan una salida cifrada de igual longitud que la entrada en claro.

Algunos ejemplos de algoritmos de cifrado con clave simétrica son: DES y su variante Triple DES (más lenta en tiempo de ejecución pero más segura), RC4, Blowfish, IDEA y AES. Si bien se han encontrado vulnerabilidades en la mayoría de ellos, estos criptosistemas suelen ser bastante seguros si se emplean correctamente.

III-A1. El scrambling, un cifrado simétrico ligero: Los algoritmos simétricos que acabamos de ver son considerados "fuertes", en el sentido de que son complejos y difíciles de romper (aún así, tienen sus propias vulnerabilidades, como se ha mencionado). La contrapartida es que no son tan eficientes en tiempo de ejecución como se desearía,lo cual supuso un escollo en los primeros intentos de transmisión multimedia en tiempo real. Estas transmisiones necesitan una fuente que sea capaz de cifrar al mismo ritmo que emite, y un receptor que pueda descifrar el contenido a la velocidad suficiente como para ofrecer una reproducción ininterrumpida. Además, los receptores suelen estar implementados en hardware, por lo que era preferible un esquema de descifrado sencillo.

La solución planteada fue el scrambling. Su significado en castellano es "desordenz en eso consiste. Se trata de encriptar el contenido mediante esquemas sencillos y rápidos en su ejecución, en los que interviene una clave simétrica (que generalmente recibe el nombre de control word, CW). Las operaciones realizadas en el contenido suelen ser XORs, intercambios, desplazamientos, etc, de forma similar a los cifradores de Feistel "fuertes", pero en este caso el proceso global es bastante más sencillo. De esta forma se consigue el objetivo principal de estos algoritmos: la eficiencia.

Como es de esperar, esta mejora en eficiencia conlleva concesiones en el aspecto de la seguridad. Los criptosistemas basados en scrambling son más sencillos de romper. Especialmente efectivos son los ataques estadísticos: si encriptamos dos fragmentos de contenido iguales con la misma clave, obtenemos el mismo resultado cifrado. Los flujos audiovisuales, que contienen gran cantidad de información similar, son un ejemplo claro de aplicación de estos ataques. La solución adoptada tradicionalmente ha sido cambiar la clave de scrambling con una frecuencia muy alta, del orden de 20 a 30 segundos. A su vez, esto provoca una saturación tanto en el emisor del contenido como en la red, que tienen que soportar un intenso tráfico de transmisión de claves.

El algoritmo de scrambling más extendido es CSS (Content Scrambling System), implementado en los DVDs en un intento de protegerlos contra la copia ilegal. Su funcionamiento se vale de operaciones XOR y tablas de búsqueda [9].

CSS fue roto en 1999 por un adolescentenoruego de 15 años, Jon Lech Johansen, y un grupo de personas que aún hoy permanece en el anonimato. La debilidad de CSS está causada por la corta longitud de la clave simétrica, de sólo 40 bits (de los cuales, y por errores en el diseño, sólo 16 son realmente importantes). Esto hace que pueda ser comprometido mediante ataques de texto plano (un bloque de texto en claro se compara con su correspondiente cifrado para extraer información sobre la clave) en un tiempo razonable. El algoritmo se emplea aún hoy, debido al gran coste que supondría un cambio en toda la industria del dvd.

El criptosistema sucesor de CSS es conocido como Cryptomeria [1]. Se emplea principalmente en la televisión digital japonesa.

III-B. Criptografía de clave pública

En este tipo de cifrados (también llamados de clave asimétrica) cada comunicante tiene dos claves, una de ellas pública y la otra privada o secreta. Los algoritmos se diseñan de forma que la publicación de su funcionamiento no disminuya su fortaleza: lo que realmente debe hacer fuerte a un algoritmo son las claves. La peculiaridad de estos algoritmos consiste en que ambas dependen la una de la otra en el sentido de que son inversas, es decir, con una clave desciframos lo que previamente se cifró con la otra. Desencriptar con la misma clave de cifrado no nos proporcionará el mensaje claro.

La clave privada queda siempre en posesión de su dueño, mientras que la clave pública se pone en conocimiento de los demás usuarios. Este último hecho no afecta a la fortaleza del criptosistema, ya que, aunque dependientes, a partir de una es computacionalmente imposible obtener la otra.

El cifrado simétrico puede emplearse con dos finalidades distintas y complementarias:

- Integridad: supongamos que el emisor cifra un mensaje con la clave pública del receptor. Esto quiere decir que sólo el poseedor de la clave privada puede desencriptar la información. De esta forma, el emisor se asegura de que el mensaje sólo será leído por la persona a quien está destinado. El receptor, en cambio, no tiene garantía sobre la identidad de la fuente, ya que la clave pública está a disposición de una gran cantidad de usuarios.
- Autenticación: el segundo método de uso viene a dar una solución al problema presentado en el párrafo anterior: si el emisor cifra la información con su propia clave privada, entonces el receptor puede descifrarla con la clave pública de éste, y asegurarse de que sólo él pudo encriptarla (siempre y cuando la clave privada del emisor no haya sido interceptada por un tercero). El concepto de *firma digital* se basa en esta idea (se describirá en la sección III-C).

Los algoritmos de cifrado con clave pública más comunes son RSA, ElGamal y los algoritmos con Curvas Elípticas (ECC). Estos algoritmos suelen ser lentos en su ejecución.

III-C. Firmas digitales

En las secciones III-A y III-B se ha hablado de cifrados con clave simétrica y asimétrica, respectivamente, y de su velocidad de ejecución. El cifrado simétrico, más rápido, se emplea principalmente para intercambiar la información entre ambas partes comunicantes. Además, el hecho de que el mensaje cifrado que generan tenga la misma longitud que el mensaje original los convierte en idóneos para este propósito. El cifrado asimétrico, más lento, se emplea principalmente en el intercambio de claves simétricas y la generación de firmas digitales.

La finalidad de la firma digital es demostrar la autenticidad del emisor. Éste cifra un resumen de la información en claro con su clave privada (un resumen, o hash, es una serie de "trozos" de información extraídos de la original mediante algún criterio). Esto se conoce como "firma digital" o simplemente "firma". El receptor, después de descifrar el mensaje con la clave simétrica, calcula el resumen del resultado y descifra la firma con la clave pública del emisor. Si el resumen proveniente de la firma y el obtenido por él coinciden entonces tiene la seguridad de que la fuente es quien dice ser (siempre que la clave privada del emisor no haya sido comprometida).

III-D. Funciones resumen o hash

Acabamos de hablar de resúmenes, en relación con las firmas digitales. Una función resumen toma como entrada un mensaje y genera una salida (de tamaño fijo) que depende directamente de éste, lo cual viene a ser un identificador único. Estas funciones se diseñan de tal forma que es altamente improbable que dos mensajes distintos tengan el mismo resumen. Además de su empleo en las firmas también es frecuente enviar por la red un hash de alguna información secreta, para mostrar una prueba de que se posee esta última sin correr el riesgo que supondría exponerla en la red.

Los requerimientos para una función resumen son:

1. Puede aplicarse a un bloque de datos de cualquier tamaño.
2. El resultado tiene un tamaño fijo.
3. Es fácilmente calculable.

4. Dado el resumen, es computacionalmente imposible obtener el mensaje original, es decir, el proceso es unidireccional.
5. Dado el mensaje, es computacionalmente imposible encontrar otro diferente tal que sus resúmenes coincidan (resistencia a colisiones débil).
6. Es computacionalmente imposible encontrar dos mensajes diferentes tales que sus resúmenes sean iguales (resistencia a colisiones fuerte).

Las tres primeras propiedades se requieren para la aplicación práctica de la autenticación. La cuarta es importante si la autenticación involucra un valor secreto. Si esta propiedad no se cumple, un atacante podría obtener el valor secreto a partir del resumen. La quinta propiedad garantiza que el mensaje que se quiere autentificar, aunque pueda ser leido, no puede ser alterado o sustituido. La sexta hace a este tipo de funciones resistentes contra los ataques de cumpleaños basados en la paradoja del cumpleaños. Este tipo de ataques intenta encontrar mensajes que generen el mismo hash que el mensaje original, lo cual puede hacerse mediante fuerza bruta en un tiempo proporcional a la raíz cuadrada de la longitud del resumen [10].

Algunas funciones hash son MD2, MD5, SHA-1, SHA-512 y Tiger.

III-E. Distribución de claves

El problema que se plantea es cómo compartir una clave simétrica de forma segura entre las dos partes. Para ello se emplea la criptografía de clave pública, la cual tiene la desventaja de ser más lenta en ejecución y de generar mensajes más largos que la información en claro, pero esto no supone un problema cuando los mensajes son cortos (como una clave, por ejemplo). En III-B hemos comentado su utilidad a la hora de buscar integridad y autenticación. Sin embargo, uno de los mayores problemas que encontramos a la hora de hacer uso de la criptografía asimétrica es el ataque *Man in the middle*, precisamente a la hora de intercambiar claves públicas con otros usuarios. En este tipo de ataque, un usuario malintencionado intercepta las comunicaciones de otros dos. Supongamos que A quiere hacer llegar a B un mensaje cifrado. Para esto, A primero pide a B su clave pública. Si el usuario malintencionado M es capaz de interceptar dicha clave sin que llegue a A, puede enviar a éste su propia clave pública. A creerá que proviene de B, a quien enviará el mensaje cifrado con ella. Si M puede interceptar a su vez dicho mensaje lo descifrará con su clave privada, con lo que tendrá acceso a la información. Para agravar la situación, M puede evitar que B sospeche enviándole el archivo cifrado con la clave pública de éste último. M puede repetir el proceso en sentido inverso y, de esta forma, puede darse la situación de que A y B interambien información cifrada sin tener conocimiento de que están siendo espiados. Por otra parte, M puede modificar a su vez dicha información, con lo que las consecuencias serían aún peores. Para solucionar este problema de seguridad se emplean los certificados digitales y los algoritmos de intercambio de claves. El algoritmo de intercambio más popular es TLS (anteriormente SSL), descrito en III-E2.

III-E1. Certificados digitales: Los certificados son, en esencia, una forma de distribución de la clave pública de su dueño. Un certificado de una persona o entidad, llamémosle A, almacena la clave pública de ésta (junto a otra información adicional) de forma que cuando B desee enviar información a A sólo necesita obtener su certificado y cifrar los datos con la clave pública contenida en él. Si A ha guardado en secreto su clave privada, sólo él podra descifarlos. A la información suele acompañarle en la práctica una firma digital para confirmar que el mensaje no ha sido modificado.

Ahora bien, de nuevo encontramos el problema de la compartición: ¿cómo puede B obtener el certificado de A con garantías de autenticidad? Aquí entran en juego las autoridades certificadoras (CA, Certification Authority): entidades mundialmente reconocidas que garantizan la autenticidad de los certificados que proporcionan. De esta forma, B puede obtener el certificado de A (si A es una empresa, p. ej.) de alguna de estas entidades, y usarlo para comunicarse con él. Ejemplos de CAs son DigiCert Inc., GeoTrust Inc., Thawte y VeriSign Inc.

El uso más común de los certificados se da en la autenticación de las partes y el establecimiento de una clave simétrica. Esta clave será generada por una de las dos partes y viajará hasta la otra protegida por la clave pública del certificado. El receptor empleará entonces su clave privada para obtener la simétrica e iniciar la comunicación. Ampliamos la descripción del proceso en el siguiente apartado.

III-E2. SSL: SSL (Secure Sockets Layer) fue introducido por Netscape en 1994 con la idea de facilitar las comunicaciones cifradas punto a punto entre clientes y servidores. La versión 3.0 sirvió de base al IETF (Internet Engineering Task Force) para publicar la versión 1.0 del protocolo TLS (Transport Layer Security). SSL 3.0 y TLS 1.0 son en gran medida el mismo protocolo. Lo que a continuación se expone es válido para ambos, a no ser que se indique lo contrario.

SSL se ubica en la pila de protocolos OSI en la frontera entre las capas de aplicación y transporte, sobre TCP y bajo otros protocolos como HTTP, TCP, LDAP o IMAP, dando a éstos la posibilidad de trabajar sobre conexiones seguras.

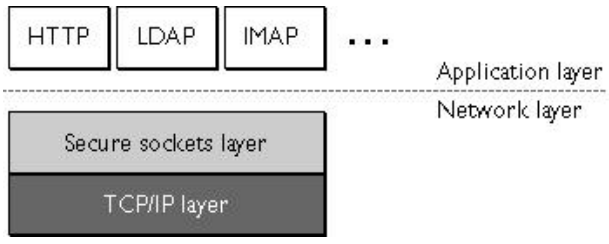

Figura 1. El protocolo SSL en la pila OSI

En esencia, SSL permite autentificar tanto al cliente como al servidor y el establecimiento de una clave simétrica para iniciar una conexión cifrada. También permite a ambas partes acordar qué algoritmos emplear.

- Autenticación del servidor: permite al cliente verificar la identidad del servidor mediante criptografía pública. Se comprueba que el certificado de este último es válido y ha sido emitido por una autoridad certificadora en la que se confía. Esto es importante cuando, por ejemplo, se envía información tal como el número de la tarjeta de crédito al hacer una compra online.
- Autenticación del cliente: si el cliente dispone de un certificado válido, el servidor puede a su vez autentificarlo a él. No es obligatorio. Un ejemplo de esta situación es la gestión a distancia de asuntos relacionados con la Administración Pública, en la que el ciudadano debe probar su identidad.
- Conexión cifrada: después de la autenticación ambas partes acuerdan un algoritmo simétrico y una clave secreta inicial (fase de handshake) para comunicarse de forma segura. En los ejemplos anteriores tanto el número de la tarjeta de crédito como la información bancaria viajarían cifradas por la red.

Una sesión SSL comienza siempre con una fase de *handshake*, en la cual el servidor se autentifica, ambas partes se ponen de acuerdo en qué algoritmos emplear y colaboran en la creación de varias claves simétricas con las que comunicarse. Adicionalmente el cliente puede autentificarse también.

Se da el caso de que las dos partes comunicantes no tienen por qué conocer el mismo conjunto de algoritmos de cifrado, hash y firma. Basta con que ambos conozcan al menos uno igual para cada operación. Por ello se les da la posibilidad de acordar cuál usar: cada uno pone en conocimiento del otro una lista con aquellos que puede emplear, y se elige siempre el más seguro posible.

Puede hallarse más información sobre este protocolo en [3].

III-F. Watermarking

La técnica conocida como *watermarking* ("marca de agua") consiste en ocultar información útil dentro de un elemento multimedia (ya sea una imagen, audio o vídeo). Normalmente la información ocultada se emplea con los siguientes fines [12]:

- Autenticación del contenido multimeda: la fuente de distribución del contenido añade a éste una marca, de forma que el receptor pueda verificar su procedencia.
- Seguimiento de copias ilegales: la fuente añade una marca relativa al receptor del contenido (su localización o identidad, si se conoce). Si posteriormente se detectan copias ilegales puede obtenerse el receptor que inició la cadena.
- Acceso condicional: la copia del contenido contiene una marca con los requisitos necesarios para acceder a él. El hardware/software reproductor debe verificar estos requisitos.

La marca puede ocultarse directamente sobre la información de los pixels o, preferiblemente, en otro dominio (de Fourier, p. ej.). En cualquier caso, éstas deben ser invisibles, o distorsionar el contenido en la menor medida posible, de forma que (1) no se detecte a ojos vista y (2) no perturbe el disfrute del contenido.

El watermarking sobre imágenes se realiza, entre otras técnicas, mediante la modificación del bit menos significativo en partes determinadas del archivo, técnicas sobre la transformada, secuencias M básicas, etc. En vídeo y audio requiere técnicas más complejas. Puede hallarse más información en [20].

IV. Sistemas de Acceso Condicional

La finalidad de un Sistema de Acceso Condicional (Conditional-Access System) es controlar el acceso a un servicio (generalmente una emisión de contenido multimedia bajo broadcast). Sólo los clientes autorizados deberían tener la posibilidad de disfrutar el servicio. La autorización es obtenida generalmente previo pago de suscripción o por un evento determinado.

Este control se consigue, a grandes rasgos, mediante el cifrado del flujo de datos. La información de descifrado sólo se entrega a aquellos usuarios que aportan las credenciales necesarias. Los sistemas de este tipo se usan tanto con emisiones lineales (los receptores reciben el flujo de forma a similar a como se recibe un canal de televisión) como con vídeo bajo demanda. Hasta ahora han sido empleados principalmente en la televisión digital (DVB), ya que la IPTV está aún poco extendida.

IV-A. CAS mediante hardware

Los CAS implementados en hardware para IPTV tienen mucho en común con aquellos empleados en la DVB [16], debido a las similitudes entre ambos paradigmas. Sin embargo, existen algunas diferencias en el sistema de distribución de la señal:

- En la DVB la comunicación es unidireccional. El uso de Internet en la IPTV posibilita el envío de mensajes en ambos sentidos y por tanto la interacción cliente-servidor. Esto amplía en gran medida el modelo de negocio de la IPTV.
- El flujo multimedia es mucho más estable en DVB, ya que la red de distribución es dedicada. El retardo es además muy pequeño. Un flujo basado en paquetes IP sufrirá retardos, llegadas en desorden y cortes.
- El servicio de IPTV comparte la red con muchos otros de diferente naturaleza. Además, debe tener en cuenta la existencia de elementos tales como firewalls, proxies, etc.

Normalmente el cliente recibe un set-top box que se encarga de las tareas relacionadas con el acceso y la decodificación de la señal. Las credenciales de acceso suelen almacenarse en smart-cards proporcionadas por el distribuidor de IPTV. En la práctica este tipo de sistemas no está bien documentado, ya que cada fabricante intenta velar por la seguridad del suyo manteniendo su funcionamiento en secreto. Aún así la mayoría parte de una base bien conocida, basada en una jerarquía de claves, propuesta por la International Telecommunication

Union (ITU) en 1992 [8]. A continuación veremos en qué consiste este esquema básico.

El flujo multimedia se cifra con algún tipo de algoritmo simétrico, tal como un algoritmo de scrambling o un cifrado del tipo Triple DES o AES. Debido a la limitada capacidad de cálculo de los set-top boxes suelen preferirse los algoritmos de ejecución rápida. El cifrado sólo afecta a la información multimedia, no a las cabeceras de los paquetes, de forma que éstos puedan ser procesados por el receptor. Las claves de cifrado/descifrado reciben el nombre de Control Words (CWs). Para distribuir estas claves a los clientes de forma segura el esquema se vale de dos tipos de mensaje enviados por el servidor al cliente, los ECMs y los EMMs:

- ECM (Entitlement Control Message): contienen la CW con la que descifrar el flujo multimedia. Se cifran con una clave de segundo nivel, que suele recibir el nombre de clave de servicio (service key).
- EMM (Entitlement Management Message): este tipo de mensajes permite controlar la capacidad del receptor de desencriptar el flujo. En ellos se envía la service key que permite obtener las Control Words descifrando los ECMs. Están encriptados mediante una clave de tercer nivel, distribuida con la smart-card. Esta clave se llama clave de usuario (user key). Además, el mensaje puede contener otro tipo de información útil, tal como el estado del servicio o datos del pago (en caso de pago-por-visión).

Normalmente, y para evitar ataques basados en estadística contra el algoritmo simétrico que protege el flujo, cada CW tiene un tiempo de vida muy limitado (del orden de 15 a 20 segundos), denominado *criptoperiodo*. Esto implica que con la misma frecuencia el servidor debe enviar un nuevo ECM con la clave correspondiente a cada receptor. Estos paquetes suelen viajar en el mismo flujo, y en principio cualquiera que esté a la escucha en la red puede recibirlo, aunque no descifrarlo. Por motivos de seguridad la service key también se renueva: una clave de este tipo suele tener un tiempo de vida de entre 12 y 24 horas. Esto implica el envío a los clientes de un nuevo EMM con esa frecuencia. Finalmente, cada EMM recibido se descifra con la clave de usuario contenida en su smart-card.

En referencia al tipo de conexión empleada en las transmisiones diremos que el stream multimedia suele viajar bajo UDP. Esto es así debido a la necesidad de aprovechar lo máximo posible el ancho de banda disponible. UDP es un protocolo no orientado a conexión, que no gestiona sesiones ni requiere respuestas de confirmación por parte del receptor. La contrapartida es que no se tiene la seguridad de que todos los paquetes lleguen a su destino, o en orden, pero esto es una prioridad menor cuando de comunicaciones audiovisuales se trata: la pérdida de unos pocos bytes no afecta en gran medida a la reproducción. Los ECMs suelen viajar multiplexados con el stream, por lo que, al no haber garantía de que lleguen a su destino, es necesario enviar cada ECM repetidas veces a lo largo del tiempo. A diferencia de éstos, los EMMs viajan separados del flujo, normalmente en conexiones TCP punto a punto.

Cuando se aplica un CAS a una emisión lineal el contenido se encripta al vuelo, por lo que es preciso emplear un algoritmo simétrico eficiente. Cuando lo que se transmite es vídeo bajo demanda, los proveedores de contenidos exigen, por motivos de seguridad, que estos últimos se almacenen ya encriptados en el servidor. Se suelen reencriptar cada cierto tiempo como medida adicional.

La Figura 2 muestra un diagrama de este esquema básico.

En el diagrama, el lado del servidor consta de los siguientes bloques: un cifrador para el flujo multimedia, un generador de CWs y un módulo de gestión de suscripciones, el cual genera los ECMs y EMMs. El generador de CWs genera una clave y la envía al cifrador, el cual la emplea para encriptar el contenido e inyectarlo en la red. La CW también llega al módulo de gestión de suscripciones que prepara y envía un ECM que la contiene. Este ECM está cifrado con una service key generada por el módulo. Como hemos dicho, se genera un ECM distinto cada 15 segundos aproximadamente. Cada 12 horas (o 24) el módulo de gestión de suscripciones cambia la service key, por lo que debe también preparar un nuevo EMM y enviarlo a cada cliente. La clave de usuario que encripta el EMM es distinta para cada cliente, y coincide con la que éste tiene almacenada en su smart-card. El módulo de gestion de suscripciones debe disponer de una base de datos con las smart-cards distribuidas, que indique también a qué canales o paquetes está suscrito cada cliente. Para restringir el acceso de un cliente a un cierto contenido (porque no esté suscrito) basta con no enviarle el EMM correspondiente.

El cliente, por su parte, dispone de un descifrador para el flujo y una lógica de gestión de ECMs y EMMs (todo implementado en el set-top box), además de la smart-card que contiene la clave de usuario. La lógica de gestión obtiene los ECMs y EMMs de la red y recupera de ellos las sucesivas CWs, que finalmente emplea para desencriptar el flujo multimedia y enviar la señal al televisor.

Este tipo de sistemas hardware cuenta ya con más de diez años de existencia, por lo que hablamos de una tecnología madura y muy puesta a prueba. Sin embargo, una de sus principales limitaciones es la necesidad de sustituir el hardware cuando se encuentra un fallo de seguridad serio en el sistema, lo cual puede suponer un desembolso de dinero muy importante para el proveedor. De hecho, éste bien podría decidir que es más rentable ignorar el fallo, en lugar de reemplazar todos los descodificadores. Un ejemplo de esto lo tenemos en las distintas proveedoras de televisión por satélite que han operado en España en la última década.

Los dos sistemas CAS comerciales más extendidos para DVB son NDS y Nagravision. Otros fabricantes son Conax, Irdeto, Verimatrix, SECA, Widevine, Philips y Motorola.

IV-A1. Estándares: Además de las sistemas desarrollados por cada fabricante, existen dos intentos de estandarización por parte de foros industriales.

- SimulCrypt: este sistema permite a dos sistemas CAS DVB independientes funcionar sobre un mismo stream multimedia, cada uno con su propio modelo de set-top

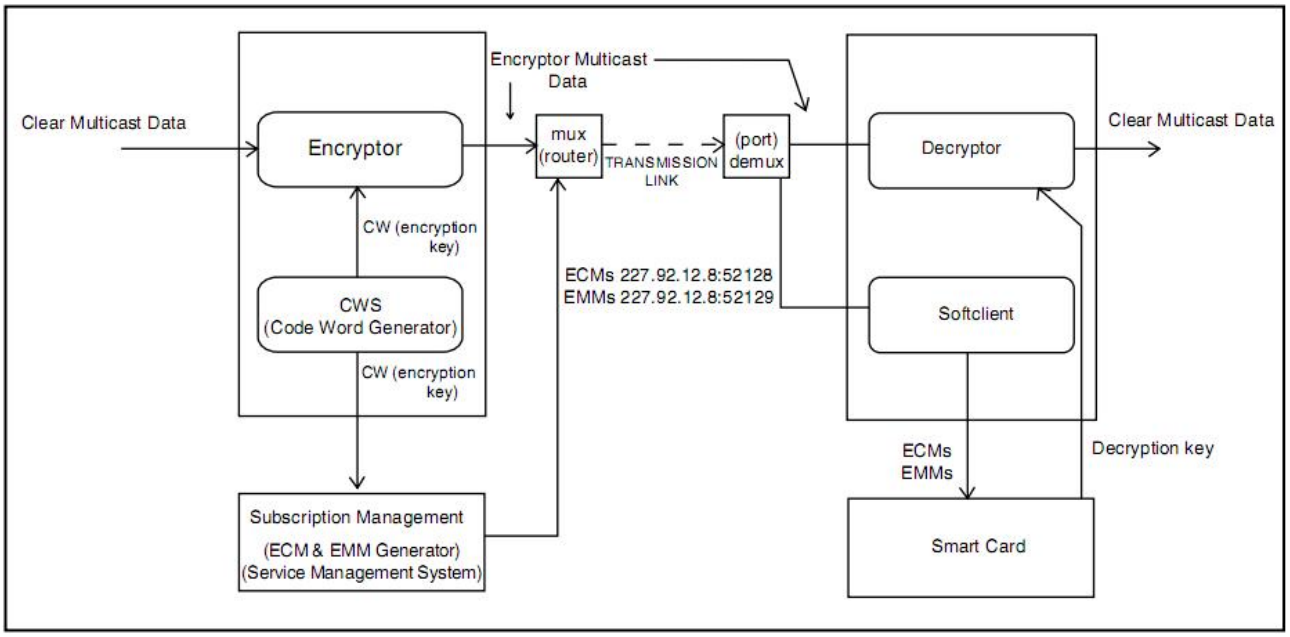

Figura 2. CAS básico mediante hardware

box. La integración se consigue multiplexando los ECMs y EMMs de ambos en el stream, de forma que cada cliente elige aquellos mensajes que puede interpretar, ignorando el resto.

- MultiCrypt: adopta un punto de vista distinto: permite cambiar entre distintos CAS empleando el mismo set-top box, gracias al estándar Common Interface (CI), un adaptador que acepta diferentes smart-cards, en función del proveedor que quiera emplearse. Desde su desarrollo a mediados de los años 90 la Comisión Europea obliga a los fabricantes a incluir el CI en todos los receptores integrados en televisiones.

IV-B. CAS mediante software

Esta aproximación ofrece varias ventajas sobre los esquemas basados en hardware. La principal es la mayor flexibilidad del código, que lo capacita para realizar tareas más complejas que las que pueden llevar a cabo los sistemas basados únicamente en circuitería. Otras ventajas son [17]:

- Fáciles de actualizar en el caso de fallos de seguridad: acabamos de ver que en un sistema hardware es complicado sustituir todos los dispositivos en caso de que se detecte un problema. En cambio, actualizar un programa software es una tarea sencilla.
- Soportan cifrados fuertes: con toda seguridad el software correrá en una máquina más potente que un set-top box tradicional. Esto permite ejecutar algoritmos de cifrado muy resistentes (RC4, AES, Triple Des, etc.), en lugar de algoritmos de scrambling. Además, pueden permitirse emplear algoritmos de clave pública, bastante más complejos.
- No necesitan smart-cards: estos dispositivos se emplean en los sistemas hardware para almacenar claves de usuario que posibiliten la autenticación del cliente. Los CAS software emplean otro tipo de mecanismos, como indica el siguiente punto.
- Usan certificados digitales: como sustituto de las smart-cards. Esto proporciona una seguridad mucho mayor.
- Aprovechan bien la comunicación bidireccional: al estar conectados a Internet, los clientes IPTV tienen la posibilidad de establecer comunicaciones bidireccionales con el servidor. Los clientes software sacan más provecho que los hardware de esta característica, ya que pueden gestionar transacciones más complejas.

En principio, un CAS software consta de tres módulos básicos: un servidor de cifrado, un servidor de claves (KMAS, Key Management and Authorization Server) y un módulo cliente.

El servidor de cifrado se encarga de cifrar, antes de su envío, el contenido multimedia. En la gran mayoría de los casos este cifrado es simétrico. Como acabamos de mencionar, los algoritmos empleados son algunos de los más fuertes: de hecho el preferido es CSA (Common Scrambling Algorithm), un algoritmo propuesto por el European Telecommunications Standards Institute para la transmisión de televisión digital. Actualmente la mayor parte de los sistemas DVB lo emplean. No se le conocen vulnerabilidades en la práctica.

El servidor de claves o KMAS se encarga, principalmente, de:

- generar y almacenar de forma segura las claves con las que el servidor de cifrado encriptará el contenido, asignando a cada una de ellas un tiempo de vida similar al criptoperiodo, y
- distribuir de forma segura dichas claves a aquellos usuarios autorizados que las soliciten.

La segunda tarea es sin duda la más compleja, ya que comprende tanto la autenticación del cliente como la entrega segura de las claves. En ambos casos se hace uso de la criptografía de clave pública y, más concretamente, de certificados y firmas digitales. En primer lugar, los algoritmos de firma digital son robustos y seguros. En segundo lugar, un certificado permite al servidor verificar la identidad del cliente, y viceversa. Adicionalmente, dado que en él se incluye la clave pública del propietario, hace posible el establecimiento de una comunicación segura, cifrada bajo SSL o IPSec, por ejemplo. Como medida adicional de seguridad las autoridades certificadoras guardan listas de revocación: listas con certificados que se han visto comprometidos. El servidor ignorará siempre cualquier intento de comunicación que venga avalado por alguno de estos certificados. Por todo ello, una infraestructura basada en certificados y conexiones cifradas es difícil de romper, siempre y cuando esté bien mantenida.

La Figura 3 muestra el funcionamiento básico de un CAS software. En primer lugar, el cliente busca un servidor de claves y, una vez encontrado, hace una petición (pasos 1 y 2). En respuesta, el servidor desafía al cliente a que demuestre su identidad (paso 3). En el paso 4 el cliente envía su certificado, que es verificado por el servidor. Finalmente éste responde con la/s clave/s, convenientemente protegida/s bajo una conexión cifrada (SSL, p. ej.).

IV-C. La necesidad de eficiencia

El modelo de negocio existente en la DVB tradicional y en la nueva IPTV presenta dos alternativas en cuanto a la contratación. En primer lugar existe la solución obvia de suscripción, o *Pago por canal* (PPC), en la cual el cliente tiene acceso a un canal o grupo de canales bajo pago de una cuota que suele ser mensual. Lo más común es que los canales se agrupen en "paquetes"jerárquicos, de forma que cada paquete añade un cierto número de canales a los del inmediatamente inferior a cambio de un precio más elevado.

La segunda opción de contratación es el *Pago por eventos* o *Pay-per-view*. En este caso el cliente paga por ver un programa concreto en un canal especial. Ejemplos claros y conocidos son un partido de fútbol o una película emitidos a una hora determinada. El espectador emplea el set-top box para enviar la orden de compra al sistema, el cual cargará la factura en su cuenta corriente. Antes de la compra al cliente se le restringe el acceso al canal. Lo mismo sucede cuando el evento termina.

Los dos modelos de pago suelen coexistir en el mismo sistema. Con esto un espectador puede disponer de los canales que más le interesan (digamos, un canal de noticias 24 horas, uno de cine clásico, uno de deportes y uno de documentales) y pagar por programas puntuales de interés que el proveedor convenientemente emitirá bajo pay-per-view en canales dedicados a ello (en España estos canales tradicionalmente reciben el nombre de taquillas. Un partido de fútbol de máxima espectación es un buen ejemplo de evento en una taquilla). El modelo de pago por canal también puede encontrarse por sí solo, no así el de pago por eventos. Un sistema DVB o IPTV únicamente pay-per-view sería muy limitado, caro y ofrecería pocos alicientes al público. En cualquier caso, un usuario no debería tener acceso a los contenidos por los cuales no haya pagado, esto es, (1) cuando el evento termine o (2) cuando deje de abonar la suscripción mensual. Esto se conoce como *privacidad hacia adelante* La *privacidad hacia atrás* consiste en que un usuario recién llegado no pueda tener acceso a la información anterior, lo cual implica refrescar la clave de servicio con cada llegada de un nuevo usuario. Esto también es necesario en el pay-per-view por el hecho de que un cliente no autorizado podría almacenar la señal cifrada durante horas y, con un solo pago posterior, descifrarla. Las conferencias multicast privadas también presentan estas dos restricciones.

Para cumplir con ellas el esquema de distribución de claves propuesto en IV-A precisa renovar la clave de servicio de cada canal cada vez que se da uno de los dos casos. Esto es posible siempre que el número de usuarios se mantenga bajo. De otra forma el envío de EMMs se convierte en un cuello de botella, ya que puede ser muy costoso mantener actualizados de forma contínua a los cientos de miles de clientes que un sistema de este tipo puede tener en la actualidad. Las previsiones de crecimiento y auge de este tipo de sistemas sólo empeoran la situación.

Es por ello que se ha invertido mucho esfuerzo buscado soluciones al problema.

Chiou et al. [11] propusieron en 1989 (antes de la aparición de la recomendanción de la ITU [8]) una solución llamada "secure lock", con el fin de distribuir nuevas claves de forma segura. La computación empleada en el "secure lock"se basa en el Teorema Chino del Resto, el cual requiere un tiempo de computación considerable para número grandes. Esto hace que dicho esquema no sea adecuado para la distribución de vídeo en tiempo real.

Lee [13] propuso en 1996 una solución derivada del esquema de tres niveles: él expande la jerarquía a cuatro niveles de claves, pero el coste del cifrado y la transmisión se vuelve inadmisible para transmisiones en tiempo real, además de no alcanzar demasiada flexibilidad a la hora de procesar incorporaciones y salidas de suscriptores.

En 1999 Tu et al. [21] desarrollan la idea de Lee adaptándola al enfoque por grupos: ahora los suscriptores se dividen en conjuntos, basándose en los paquete de canales que contratan y el momento en que se unen al sistema. Con esto consiguen reducir el coste de refrescar una clave, pero no reducen la cantidad de mensajes EMM que deben enviarse, haciéndolo poco apropiado para la distribución dinámica de claves en tiempo real.

En las siguientes secciones se muestra el estado del arte en este tema, revisando artículos recientes. Además, se tratan otro tipo de cuestiones relacionadas con el acceso condicional, como son el almacenamiento del contenido en el cliente y el empleo de metadatos.

IV-D. Esquemas organizados en grupos

En este tipo de esquemas los canales se dividen en grupos (la distribución puede hacerse atendiendo a los paquetes ofertados o según otro criterio). En líneas generales a cada

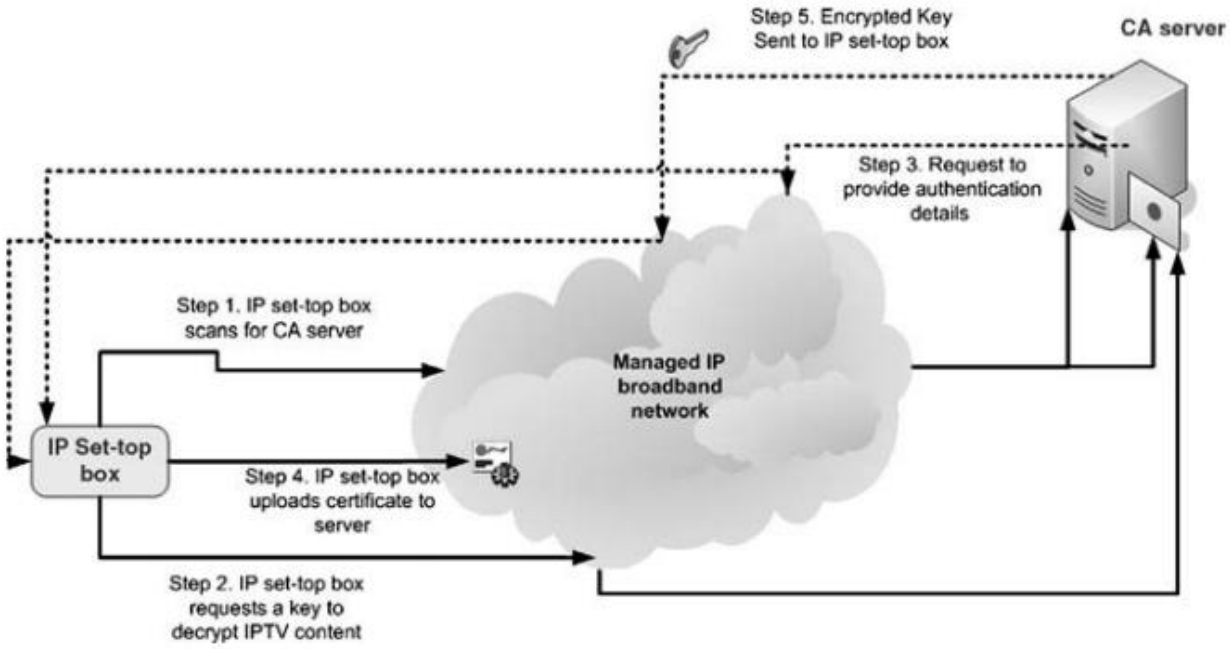

Figura 3. CAS básico mediante software

uno de ellos se le asigna una *clave de grupo* que poseen los usuarios asignados a él, y que es necesario refrescar cuando un cliente se une al grupo o lo abandona. La ventaja de la división en grupos es que acota la magnitud del refresco, reduciendo el número de usuarios asignados a cada clave. Este tipo de esquemas suele basarse en el problema de la factorización de números grandes para garantizar su seguridad, por lo que suelen recordar a RSA.

En [14] Liu et al. se plantea un esquema de distribución con la escalabilidad como meta principal. Para ello se adopta una jerarquía de cuatro niveles de claves, que llaman CW (Control Word), AK (Authorization Key), RGK (Receiving Group Key) y MPK.

La división del conjunto de canales no tiene por qué ser igual a la ofertada en los paquetes promocionales. De hecho, según los autores el proveedor debería hacer una distribución que favoreciera la eficiencia. El grupo i recibe el nombre G_i. Además, existe un supergrupo, G_H, al cual están subordinados todos los G_i. $ch_{i,j}$ es el canal j-ésimo del grupo i.

AK es única para cada canal: al canal j del grupo i le corresponde $AK_{i,j}$. RGK es única para cada grupo: al canal i le corresponde RGK_i.

El problema de refrescar las claves se traduce en que cada cliente pueda generar periódicamente una nueva AK para cada canal al que está suscrito, en base a información renovada por el servidor. En la generación de AK intervienen varias claves adicionales que mostramos a continuación:

- $HRGK$: clave que recibe el grupo G_H.
- RGK: clave que recibe un grupo. A G_i le corresponde RGK_i. Esta clave es compartida por todos los canales de un mismo grupo.
- SGK (Secret Group Key): es única para cada grupo. G_i recibe SGK_i.
- SCK (Secret Channel Key): es única para cada canal. El canal j del grupo i recibe $SCK_{i,j}$.

El usuario suscrito al canal j del grupo i guarda inicialmente en su smart-card $HRGK$,SGK_i y $SCK_{i,j}$. RGK_i es función de $HRGK$ y SGK_i. La Authorization Key del canal j del grupo i, $AK_{i,j}$, se obtiene apartir de RGK_i y $SCK_{i,j}$. La figura 5 aclara el proceso de generación. Las claves marcadas con un asterisco (*) se almacenan inicialmente en la smart-card. *Los círculos blancos indicas operaciones en las que intervienen las claves indicadas.* Las operaciones no se muestran por claridad.

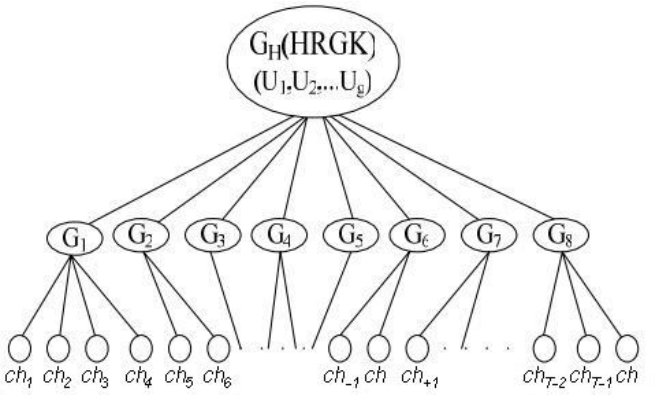

Figura 4. Distribución de canales por grupos en Liu et al.

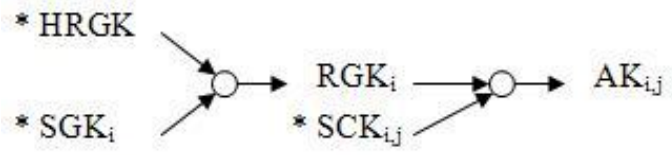

Figura 5. Relación entre las distintas claves en Liu et al.

El proceso de refresco se realiza independientemente para cada grupo, y tiene lugar con cada nueva suscripción o abandono. Consiste en renovar SGK_i y, por ende, RGK_i. Para ello el sistema construye un mensaje único que hace las funciones de EMM. Aquí es donde hace su aparición la última clave de la jerarquía, MPK. Existe una MPK distinta para cada grupo y usuario, esto es, si el usuario 3 está suscrito a algún canal del grupo 1, tendrá una clave $MPK_{1,3}$. El EMM es el resultado de una operación en la que intervienen las MPK: cuando se trata de renovar SGK_i el sistema calcula el mensaje haciendo intervenir en los cálculos SGK_i y las MPK de todos los usuarios del grupo i.

Si se trata de un abandono de suscripción se deja fuera de los cálculos el MPK del usuario que se marcha. El EMM se envía mediante broadcast. El usuario que acaba de abandonar la suscripción obtendrá un resultado inútil si intenta procesar el EMM. Los usuarios receptores que sigan en el grupo realizarán un cálculo inverso para obtener SGK_i. Con ella y $HRGK$ (que no cambia) podrán generar una nueva RGK_i y, finalmente, la nueva $AK_{i,j}$. La Figura 6 esquematiza el proceso. De nuevo los círculos blancos indican operaciones que no se muestran por simplicidad.

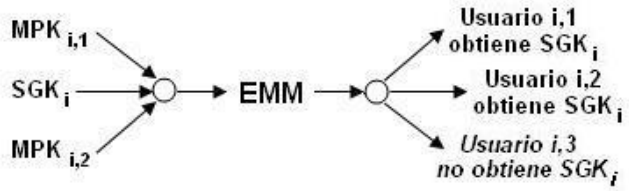

Figura 6. Refresco de SGK para el grupo i. El usuario i,3 ha abandonado.

En caso de no ocurrir eventos de suscripción, los autores proponen renovar la AK diariamente y la SCK semanalmente, para incrementar la seguridad. Como medida adicional se plantea también la posibilidad de emplear pairing para evitar el uso de la tarjeta en otros dispositivos hardware.

Respecto a la eficiencia, tanto para la operación de suscripción como para la de abandono sólo es necesario generar y enviar mediante broadcast un EMM para actualizar el SGK del grupo afectado. Adicionalmente es posible refrescar todos los grupos simultáneamente mediante la actualización de la clave $HRGK$. De nuevo sólo es necesario enviar un mensaje bajo broadcast. Los cálculos correspondientes a los círculos blancos de las figuras, necesarios para derivar las claves, están basados en operaciones aritméticas simples, en las que la carga computacional es baja. Finalmente, dado que la computación requerida para calcular el EMM está en función del número de usuarios pertenecientes al grupo, si se rebasa cierto umbral de rendimiento siempre es posible dividir el grupo en tantas partes como sea conveniente. Los autores remarcan que, dado el bajo coste de refresco, el esquema es especialmente apropiado para el modo pay-per-view, en el que la unión y abandono de usuarios es muy frecuente.

Zhu et al. desarrollan un poco más la idea de distribución en grupos en [24]. Ellos plantean, además de una jerarquía de claves de cuatro niveles, una jerarquía de grupos, en la que los canales son los nodos hoja.

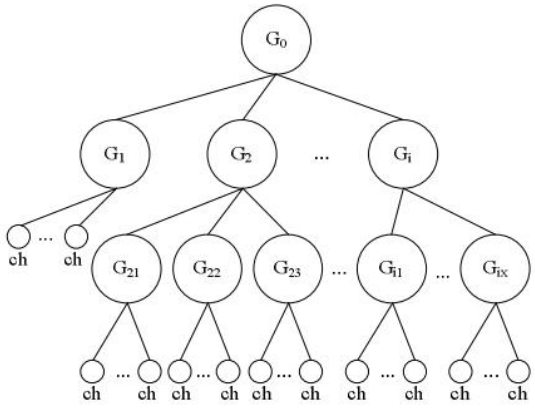

Figura 7. Jerarquía de grupos en Zhu et al.

El nodo raíz es un super grupo, llamado G_0. Los hijos directos son G_1, G_2, etc. Los hijos de G_1 son G_{11}, G_{12}, etc., de forma que el último dígito del nombre indica la posición del grupo o canal dentro del grupo padre, y los dígitos de la izquierda indican su genealogía. Como ejemplo, supongamos que G_2 es un grupo de noticias. Sus hijos podrían estar relacionados con él de la siguiente forma: G_{21} podría ser un grupo de canales de noticias internacionales, G_{22} uno de canales de noticias locales, G_{23} referirse a canales de noticias deportivas, etc. Los clientes pueden suscribirse a grupos completos o, lo que parece más lógico, a canales de distintos grupos individualmente. Los autores remarcan que pueden hacerse tantas divisiones y profundidades como sea necesario.

En cuanto a las claves, reciben los siguientes nombres: CW, AK, GK (Group Key) y DK (Distribution Key). GK se emplea para derivar AK. DK se almacena inicialmente en la smart-card del usuario y se emplea para la distribución de GK y AK. Es preciso resaltar que AK está compuesto por dos sub-claves: AK_{ch} y sk, las cuales se verán a continuación.

La relación entre claves se consigue de la siguiente forma: el grupo G_0 recibe una clave arbitraria GK_0. De forma similar al esquema de RSA, el sistema elige p y q, primos grandes, y calcula $n = pq$. Para un grupo dado, digamos G_j, con un número máximo de hijos u, el sistema elige una serie de primos relativos $m_1, m_2, \ldots, m_u$ que se hacen públicos. Si

G_j es el hijo i-ésimo de G_k entonces

$$GK_j = GK_k^{m_i} mod(n)$$

El cálculo de AK_{ch} es similar: para el i-ésimo canal de G_j tenemos

$$AK_{ch} = GK_l^{m_i} mod(n)$$

Un usuario que posea una clave para un canal o grupo no podrá deducir la del padre o hermano, teniendo en cuenta la complejidad de la factorización de primos largos y que $m_1, m_2, \ldots, m_u$ son coprimos.

Una operación de refresco de claves consiste en distribuir una nueva GK. Para ello, cada cliente posee un primo x_i secreto en su smart-card, que sólo el servidor conoce. El sistema calcula los valores k y X a partir de todos los x de usuarios con acceso. k es público, no así X. A partir de X el sistema deriva una clave secreta sk. El EMM consiste en la nueva GK cifrada con sk. Por su parte, cada cliente puede derivar sk a partir de su propio x_i y k, que es público. Una vez obtenido sk puede descifrar el EMM, obteniendo GK y, a partir de ésta, AK_{ch}. Los autores señalan que sk puede refrescarse cada 10-20 segs.

La clave AK consiste en $< AK_{ch}, sk >$. El sistema cifra CW de forma que ambos valores sean necesarios para descifrarlo.

La operación de suscripción por parte de un usuario requiere que el servidor le entregue AK_{ch}. El usuario puede obtener sk de la forma indicada anteriormente.

Lo novedoso es que una operación de abandono por parte de un usuario no requiere ningún envío: en la siguiente actualización de sk, el sistema recalcula X sin tener en cuenta el primo x del usuario que abandona. Los clientes recalculan su sk y AK_{ch} no necesita ser cambiada. Al modificar sk obtenemos un cambio en AK.

De esta forma, una operación de resfresco completo (en todos los grupos y canales) de AK requiere un número de envíos igual al número de canales. Una operación de suscripción requiere una entrega. Una operación de abandono no requiere ninguna.

Un aspecto interesante de este esquema es que posibilita la gestión dinámica de los grupos: adición, eliminación o cambio de grupos y canales, sin afectar al funcionamiento del resto del sistema:

- Para la eliminación, según los autores, basta con que el servidor cambie la clave (GK en el caso de un grupo y AK_{ch} en el caso de un canal) sin comunicarlo. De esta forma, ninguno de los usuarios adscritos conocerá la nueva clave y perderán el acceso. Los grupos hermanos no se ven afectados.
- Respecto a la adición supongamos que se desea añadir un grupo o canal R bajo un padre P.
  - Si el número de hijos de P es menor que el máximo se le asigna uno de los primos relativos m_i generados. A patir de ahí se puede derivar GK o AK_{ch} (según corresponda) de R.
  - Si no quedan primos relativos generados basta con encontrar otro que lo sea con todos los anteriores y hacerlo público. La clave puede obtenerse de la misma forma que en el caso anterior.
- Un cambio de grupo puede dividirse en eliminación y adición.

Las ventajas de este esquema sobre el de Liu son (1) requiere menos operaciones de cómputo y (2) aunque AK se componga de dos partes, su distribución necesita un menor número de claves auxiliares y es menos compleja.

IV-E. Autenticación del cliente

Los sistemas futuros de IPTV aprovecharán con toda seguridad el canal bidireccional que supone Internet para ofrecer nuevos servicios de interacción entre el cliente y la plataforma. En este escenario es importante verificar la identidad del cliente para evitar usos no autorizados del servicio e identificar a usuarios fraudulentos. La solución más lógica hoy en día parece ser el empleo de un Centro de Autenticación similar a los empleados corrientemente en emisiones de certificados para transacciones vía web.

En [15] encontramos un esquema de este tipo. A primera vista es más sencillo que el de la ITU en cuanto a la jerarquía de claves y la forma de obtención de éstas. Emplea dos niveles de claves, además de un Centro de Autenticación (lo llamaremos en adelante *Authorization Center*) y watermarks. Las dos claves empleadas son CW (Control Word, para el stream) y SK (Service Key).

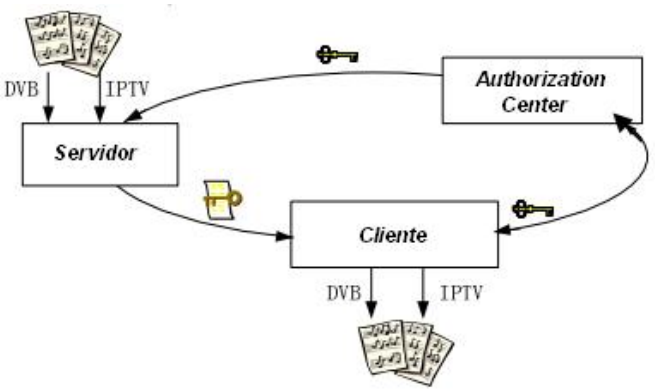

Figura 8. Esquema básico de [15]

El Authorization Center

- Genera la SK (única para todo el sistema) y la actualiza periódicamente.
- Distribuye la SK correspondiente tanto al servidor como a los clientes mediante un canal seguro (SSL, p.ej.).

El servidor de contenidos

- Genera las sucesivas CWs.
- Cifra el contenido con la CW correspondiente y lo distribuye a los clientes. Los autores proponen Common Scramble Algorithm como algoritmo de cifrado.
- Genera ECMs cifrando la CW correspondiente con la SK apropiada.

- Multiplexa los ECMs con el contenido en el flujo.

El cliente

- Se autentifica con el Authorization Center y recibe de él la SK.
- Recibe el flujo y el ECM correspondiente del servidor.
- Descifra el ECM con el SK para obtener CW.
- Descifra el contenido con la CW y lo reproduce.

Los tiempos de vida propuestos son 10 segs. aproximadamente para la CW y una hora para la SK. Implícitamente intervienen más claves, porque el proceso de autenticación con el Authorization Center debe hacerse bajo protocolos seguros (ej. SSL). Esto implica el uso de certificados y firmas digitales.

Un aspecto interesante es el uso de watermarks en el contenido. Con esto los autores persiguen dos cosas (1) control de acceso y (2) identificación del responsable en caso de piratería.

- Control de acceso: el servidor imprime una marca de agua en el contenido antes de cifrarlo. El receptor verifica la marca después del descifrado, y sólo reproduce el contenido si el resultado de la verificación es positivo.
- Identificación del responsable en caso de piratería: se imprime una segunda marca de agua en el contenido que dependa exclusivamente de su receptor. Si éste redistribuye ilegalmente el contenido y el proveedor detecta una copia entonces puede identificar inequívocamente el receptor que generó la filtración y emprender las acciones oportunas contra él. Este tipo de watermark se conoce como *fingerprint*. Los autores señalan que el fingerprint se realiza en el servidor de contenido. Esto implica entonces que debe realizarse un envío (además de proceso de watermarking y encriptado) por cada cliente, lo cual no es de ningún modo práctico o viable.

Encontramos peros al uso que hacen los autores del watermarking por el hecho de que la gestión de la seguridad recae en el cliente. En el primer caso el set-top box o software pueden ser modificados para que reproduzcan el contenido sin tener en cuenta la presencia o ausencia de watermarks en él. En el segundo caso ya hemos señalado la inviabilidad de generar una fingerprint en el servidor. Una posible solución, a nuestro juicio, sería trasladar el fingerprinting al cliente, pero de nuevo éste podría ser modificado para evitar dicho proceso y redistribuir copias sin la marca del usuario.

IV-F. Persistencia del contenido

Los esquemas expuestos hasta ahora son herederos de aquellos empleados en la televisión digital tradicional. Sin embargo, la aparición de receptores con disco duro abre una nueva vía que es necesario explorar: ahora el cliente tiene la posibilidad de almacenar el contenido recibido para su posterior visionado y, posiblemente, copia. Llamaremos a esto *persistencia del contenido*. Este nuevo paradigma se conoce como DBHS: *Digital Broadcasting Based on Home Servers*. Se abren dos vías: por un lado el desarrollo del modelo de negocio pasa por facilitar una buena experiencia de usuario, ofreciendo servicios tales como segmentación por escenas, búsqueda de escenas en base a parámetros, bookmarks y otros tales como información sobre el programa. Por otra parte, los distribuidores de contenido no ven con buenos ojos la posibilidad de que un cliente realice copias indiscriminadas de un programa al que accedió mediante un único pago. Los sistemas de acceso condicional tradicionales expuestos hasta ahora no lidian con estas dos situaciones, por lo que en los últimos años han surgido algunas líneas investigadoras interesadas en desarrollar esquemas más modernos y versátiles.

Nishimoto et al. afirman en [23], no sin razón, que la separación entre Acceso Condicional y DRM está cerrándose progresivamente. Por tanto el futuro parece estar en sistemas de acceso condicional que controlen el acceso al contenido tanto en la recepción de los datos como en su posterior reproducción una vez almacenados.

Los autores plantean las características que debe tener una solución de acceso condicional para DBHS y proponen una. Las características son:

1. Debe proteger el contenido y las claves de descifrado, así como respetar y preservar los derechos de reproducción del contenido.
2. El control de acceso y reproducción debe aplicarse individualmente a cada contenido (es decir, permitir distintas reglas para contenidos diferentes).
3. Debe ser compatible con los sistemas de emisión digital válidos en la actualidad.
4. La reproducción debe admitir comandos tales como rewind, fast forward, reproducción segmentada por escenas, etc.

El sistema planteado por los autores presenta una jerarquía de 4 niveles: los 3 usuales de la ITU (lo que facilita la compatibilidad con el estándar) y uno adicional para la reproducción a posteriori. Las claves reciben los siguientes nombres: Ks (scrambling), Kw (word), Km (master) y Kc (content). Nosotros emplearemos los nombres tradicionales para las dos primeras con el fin de facilitar la lectura: CW (Control Word), Ks (Service Key), Km (Master Key) y Kc (Content Key). Como de costumbre, CW cifra el contenido, Ks cifra CW en ECMs multiplexados en el stream y con Km (única para cada usuario) se encripta Ks para ser entregada al receptor. Las frecuencias de refresco propuestas son de una vez por segundo para la CW y una vez por mes para Ks.

Al recibir el contenido entra en juego el cuarto nivel de la jerarquía: Kc. Existe una clave Kc para cada contenido. Ésta se cifra y distribuye en forma de ECM (lo llamaremos ECM de Kc), junto a la licencia del contenido (permisos de reproducción). Dicho ECM puede multiplexarse también en el stream o enviarse por un canal distinto. No se especifica cómo se protege (con qué se cifra su información), por lo que supondremos que se envía bajo algún protocolo seguro, como SSL. Además, existe una clave adicional, llamada *de grupo*, Km'. Tampoco se proporciona mucha información sobre esta última. Cuando el contenido es almacenado, el ECM de Kc se descifra y se guarda reencriptado (también la licencia) con Km'. Con esto tenemos almacenada y protegida la Kc en el reproductor. Las CWs se guardan también, cifradas con Kc.

A la hora de reproducir, la Kc y la licencia se descifran gracias a Km'. Durante la reproducción se van descifrando (con Kc) los ECMs correspondientes para obtener la CW apropiada en cada instante. Además, se comprueba la licencia para ver si las condiciones son válidas para la reproducción. Si no lo son, ésta se aborta. Si lo son, se envía CW al reproductor. La seguridad del esquema se basa en que la licencia queda encriptada en el disco duro del receptor, con lo que un usuario malintencionado no podrá modificarla. La reproducción del contenido quedará controlada por las indicaciones de la licencia.

Los resultados de rendimiento mostrados parecen suficientes. Además, es compatible con la arquitectura tradicional de emisión de la ITU.

Sin embargo un punto débil del esquema aparece cuando se quieren emitir licencias personalizadas para cada usuario: el ECM de Kc debe ser entonces individual, lo que requiere un envío para cada cliente. Otro problema que encontramos es que la seguridad se confía al cliente. Tal vez sería posible modificar el hardware para que no consulte la licencia, o para que envíe las CWs al reproductor independientemente del resultado de la verificación.

IV-F1. Reproducción del contenido con metadatos: Una de las principales líneas abiertas consiste en el empleo de metadatos junto al contenido. Los servicios comentados en el párrafo anterior (escenas, bookmarks, etc.) así como otros tales como consulta de la parrilla y métodos de pago se ven simplificados gracias a su uso. Es fácil predecir que dicho uso se hará cada vez más frecuente.

El lenguaje XML (eXtended Markup Language) está empezando a ser empleado en estas operaciones, ya que permite estructurar y describir la información de forma simple y eficiente. Además, el hecho de que sea un estándar propicia la compatibilidad entre sistemas distintos. XML permite organizar un documento empleando una estructura en forma de árbol en la que cada segmento de información queda etiquetado e identificado. Además, dada la facilidad con la que XML organiza los documentos, es posible acompañar información descriptiva sobre los datos incluidos. Esto es lo que se conoce como *metadatos*. Su empleo permite, por ejemplo, elegir las jugadas de un partido de fútbol en las que intervenga un jugador determinado, o que hayan sido etiquetadas bajo algún criterio ("polémicas"sería uno muy apropiado). Estas posibilidades y otras parecidas pueden ser muy apreciadas por el usuario, por lo que los sistemas IPTV van a verse abocados inevitablemente a adoptar el modelo de metadatos para ofrecerlas. Los receptores que manejan este tipo de información reciben en la bibliografía el nombre de dispositivos DBRM (Digital Broadcasting Receiver using Metadata). El cliente recibe el stream multimedia por una vía y, bien por la misma bien por otra, la metainformación necesaria para realizar las operaciones de acceso.

El TV-Anytime Forum [4] clasifica los metadatos en cuatro tipos:

- Descripción de contenido (título, género, duración, etc).
- Descripción de instancia: información relativa al proceso de transmisión.
- Segmentación: división del contenido en escenas.
- Creados por el usuario (bookmarks o preferencias).

El ejemplo expuesto anteriormente sobre las jugadas de un partido de fútbol es una buena muestra del uso de metadatos de segmentación. La posibilidad de que el usuario almacene sus preferencias y bookmarks es un punto adicional a favor de este modelo. El espectador podría marcar aquellos pasajes del partido que le han parecido más interesantes para repetirlos posteriormente o, por contra, decidir qué partes desea saltar.

Existen algunas soluciones respecto a la seguridad en estos documentos. La más usada es la *firma digital XML* [7], propuesta por el World Wide Web Consortium (W3C) [5] como una adaptación del proceso de firmado a los documentos XML (aunque no sólo a ellos). Por lo tanto aporta integridad y autenticación.

Una firma XML es un nodo XML al final del documento, etiquetado como <Signature>, que contiene tres elementos hijos: <SignedInfo>, <SignatureValue> y <KeyInfo>.

- <SignedInfo>: contiene un hash en claro de las partes del documento que van a firmarse e indica, entre otras cosas, los algoritmos de hash y firma empleados.
- <SignatureValue>: en él se guarda el resultado de firmar el hash guardado en <SignedInfo>. Normalmente se emplean algoritmos de firma bien conocidos, como DSA, ECDSA o RSA.
- <KeyInfo>: incluye la información criptográfica necesaria para verificar la firma, tal como una clave pública o, preferiblemente, un certificado X.509.

Para verificar la firma, el destinatario descifra el contenido de <SignatureValue> con la información de clave pública de <KeyInfo> y compara el resultado con el hash almacenado en <SignedInfo>. Al consistir la firma en un nodo XML el conjunto metadatos-firma puede ser tratado como un solo documento, lo cual facilita todo el proceso.

Como complemento para proporcionar confidencialidad existe además un cifrado XML [6], también recomendación del W3C.

Es improbable que el flujo multimedia se divida en documentos XML firmados (uno por paquete) por la pérdida de ancho de banda efectivo que esto supondría y el tiempo de cómputo necesario para procesar en tiempo real tal cantidad de información con un algoritmo de clave pública (el tiempo de computo requerido es el principal problema de la firma digital). Además, un atacante probablemente centrará sus esfuerzos en la información confidencial, no en suplantar la película que el cliente está viendo. Es por esto que la firma XML debe emplearse en datos tales como información bancaria del usuario, preferencias, datos personales, etc.

Haremos de nuevo referencia al trabajo reciente de Nishimoto et al. En [22] plantean dos escenarios de uso. En el primero los metadatos son creados y enviados por una fuente que no tiene por qué ser el emisor del contenido, sino terceras partes autorizadas. Además, el propio espectador puede generar los

suyos propios. En el segundo escenario distintos creadores de metadatos cooperan para crear el conjunto final de metadatos. Como ejemplo supongamos que el proveedor de contenidos crea metadatos relativos a la descripción del contenido. El proveedor de la red puede añadir a su vez información sobre cuestiones relativas a la transmisión, y el usuario puede añadir bookmarks. Todo esto conforma un solo conjunto de metadatos.

Dado que puede darse la posibilidad de que los metadatos sean falsificados o generados malintencionadamente por un tercero no autorizado, los autores plantean los siguientes requerimientos para un sistema de acceso condicional de este tipo:

- Las alteraciones de los metadatos deben ser detectadas.
- Los creadores de los metadatos deben estar certificados (identificados).
- El acceso al contenido mediante metadatos debe estar controlado por el emisor (no pueden emplearse metadatos no autorizados por él).

En el mismo artículo se propone un CAS que combina firma digital XML para detectar alteraciones de la información y autenticación del creador con técnicas de control de acceso para evitar el uso no autorizado de metadatos *en reproducciones posteriores a su recepción*.

En primer lugar todos los documentos de metadatos se firman. Esto evita la alteración de éstos por parte de terceros malintencionados, garantizando además su integridad. Además, permite identificar inequívocamente a su creador. Para el empleo de la firma XML el receptor debe disponer de un certificado válido proveniente de cada posible creador de metadatos. Al recibirlos, el DBRM verifica cada firma empleando el certificado correspondiente.

Los controles de acceso tradicionales controlan el acceso en su totalidad. El control de acceso que los autores proponen está adaptado al acceso segmentado mediante metadatos. En los siguientes párrafos describimos el esquema propuesto.

El emisor envía la información de licencia requerida para acceder al contenido (los autores no entran a explicar cómo) junto a los metadatos firmados. La información de licencia incluye una clave de descifrado y una lista con los creadores de metadatos autorizados para el contenido.

El contenido enviado se cifra para restringir el acceso. Una vez que el DBRM lo recibe lo almacena en su disco duro junto a la licencia. El proceso de acceso al contenido se realiza de la siguiente forma:

1. El DBRM identifica al creador de los metadatos con la ayuda del certificado correspondiente y verifica la firma.
2. Un módulo de seguridad del DBRM verifica que el creador de los metadatos está autorizado para el contenido (éstos aparecen en la licencia recibida).
3. Si el paso anterior tiene éxito el módulo de seguridad proporciona al DBRM la clave de descifrado recibida con la licencia. Si no es así, el modulo no emite la clave.
4. El DBRM descifra y reproduce el contenido con la clave.

Los autores remarcan que los metadatos no tienen por qué ser enviados de una sola vez al principio de la emisión. También puede hacerse de forma dividida, como se indicó en el escenario 2. Supongamos la retransmisión de un partido de baloncesto: el emisor puede, por ejemplo, enviar a los receptores un marcador permanentemente actualizado a lo largo de todo el partido. En este escenario cada uno de los documentos parciales debe ser autentificado y verificado en cuanto a la integridad, ya que antes de acceder al contenido es preciso que el DBRM certifique todos los metadatos empleados. Esto implica una gran carga computacional. Por este motivo los autores afirman haber desarrollado una técnica de firma digital para metadatos divididos. Cada documento enviado lleva adjuntas dos firmas: una relativa a su propio contenido parcial y una relativa a toda la metainformación (lo que sería el documento completo único). Al otro extremo, el DBRM une los documentos recibidos y almacena sólo la firma relativa al documento completo. El resumen presente en esta firma debe ser igual al generado por el DBRM.

Los autores son poco claros en la exposición de este esquema de firma divida: según la idea que plantean el emisor debe conocer todos los metadatos dividos que van a enviarse de antemano. Si durante la emisión se emite algún documento no contemplado en la firma global inicial éste no podrá ser utilizado por el receptor. El proveedor de red tampoco podrá adjuntar ningún metadato. Una solución que vemos a este problema sería que el proveedor de red (o el creador correspondiente) sustituya la firma global que recibe por otra generada por él mismo después de añadir sus propios metadatos.

Tampoco vemos sentido a incorporar una doble firma: no parece necesario utilizar una firma parcial (sólo para los datos de cada documento) si la que realmente se va a verificar es la global después de generar un sólo documento. En cualquier caso, en el artículo se recomienda que cualquier verificación de firma se realice durante el proceso de recepción del documento para evitar retardos durante el acceso en tiempo real. Finalmente, los autores se basan en pruebas de rendimiento realizadas para afirmar que este sistema de acceso condicional es viable en PCs con lectores de smart-cards y en set-top boxes con una buena capacidad de cómputo. Probablemente el siguiente paso dado por Nishimoto será combinar las soluciones expuestas en [23] y [22] para crear un sistema de acceso condicional todoterreno y versátil.

V. Colaboración con IPTV Solutions

Como parte del acuerdo de colaboración que el Departamento de Arquitectura de Computadores y Electrónica mantiene con IPTV Solutions estamos trabajando en un sistema de acceso condicional que dé servicio al producto IPTV que la empresa desarrolla. Dicho producto recibe el nombre de eSpectia. Se basa en una arquitectura peer-to-peer para distribuir el flujo multimedia: esto da pie a algunas diferencias con la aproximación multicast tradicional. A continuación se repasan algunas de sus características.

V-A. eSpectia: P2PTV

La aproximación broadcast tradicional requiere grandes inversiones en mantenimiento de redes de distribución propias, o en concepto de alquiler de ajenas. Esto hace que sólo las grandes operadoras puedan competir en el mercado. En respuesta a este escenario surge la idea de emplear una arquitectura P2P para la red de distribución. Dicho paradigma ha empezado a conocerse con el nombre de P2PTV, y se ha convertido en una opción viable en ciertos casos.

El esquema básico de la P2PTV nos presenta a uno o varios servidores de contenido inyectando el flujo multimedia a un número determinado de nodos clientes (aquellos con mayor capacidad de transferencia). A partir de aquí el flujo se difunde por toda la red mediante un protocolo peer-to-peer apropiado, el cual debe estar diseñado para favorecer la difusión del contenido, de forma que todos los nodos obtengan los paquetes necesarios con la mínima latencia posible. Esto implica que cada nodo debe reservar una parte de su ancho de banda disponible para reenviar la información que recibe a otros. Cuando un nodo entra en la red obtiene una lista de aquellos otros a los que puede pedir contenido. Una vez que ha obtenido una cantidad de información suficiente puede comenzar a su vez a atender peticiones de otros solicitantes. El hecho de que exista uno o varios servidores de contenido al inicio de la distribución convierte a la arquitectura en P2P híbrida.

Las mayores ventajas de esta aproximación son:

1. Disminuye la tasa de transferencia de los servidores necesaria para hacer llegar el flujo a todos los clientes.
2. Su rendimiento mejora al aumentar el tamaño de la red.

Ambas características tienen un gran peso juntas, y hacen de los sistemas P2PTV una alternativa idónea en algunos escenarios concretos, como son la distribución de una señal de televisión con acceso lineal (similar al modelo de televisión tradicional) o de eventos en directo. La consecuencia directa de las dos cualidades es el drástico abaratamiento del coste de distribución de los contenidos gracias al ahorro en ancho de banda en los servidores. De tener éxito el modelo P2PTV permitirá a empresas más pequeñas ampliar el mercado de la televisión vía Internet.

Por contra hallamos las siguientes desventajas:

1. El rendimiento empeora cuando hay pocos miembros (audiencia) en la red.
2. Pueden tener una latencia alta, sobre todo al inicio de la conexión.
3. No se adaptan bien al modelo VoD (vídeo bajo demanda).

El primer problema puede superarse fácilmente: en los momentos en los que haya pocos nodos en la red éstos pueden pedir contenido directamente al servidor. Al haber pocos clientes éste estará poco cargado y podrá asumir la tarea sin problemas. Para la segunda cuestión podrían estudiarse soluciones. El último punto hace a la P2PTV poco apropiada para el futuro, donde la VoD será uno de los pilares del modelo de negocio IPTV. En el modelo VoD cada cliente elige ver un contenido determinado (típicamente películas) a una hora determinada, por lo que se hace necesario un amplio repositorio de contenidos disponible las 24 horas. Más aún, cada visionado precisa de una conexión dedicada, ya que es poco probable que dos usuarios vean el mismo programa en el mismo punto de reproducción a la misma hora. En una hipotético servicio VoD bajo P2P un cliente encontraría que ningún nodo dispone del contenido deseado en el instante necesario. La búsqueda de soluciones a este problema sería una línea de investigación interesante.

V-B. Sistema de Protección de Contenidos eSpectia

Hemos denominado Sistema de Protección de Contenidos al módulo de acceso condicional diseñado para eSpectia. El sistema actual es una evolución del trabajo realizado por Nicolás Manuel Piqueras Romero para su Proyecto de Fin de Carrera [18]. La implementación está siendo realizada en software.

Aunque se pensó como complemento de la red peer-to-peer mostrada en la sección anterior el SPC puede dar servicio a cualquier esquema de distribución via IP, ya que se planteó de forma independiente.

Los elementos que intervienen son:

- Servidor de Claves: genera las claves con que se cifrará el contenido multimedia y las entrega tanto al Servidor de Contenido como a los clientes. Se ocupa también de la autenticación de los clientes.
- Servidor de Contenido: no es parte del SPC, sino del sistema de distribución. Cifra el flujo del contenido con las claves generadas por el Servidor de Claves.
- Clientes: se autentican en el servidor y le demandan las claves de cifrado. Reciben el contenido de la red de distribución.

El sistema se ha diseñado teniendo en cuenta la portabilidad. Para ello se combina el uso de cuentas de usuario (login y password) y el concepto de sesión: cuando un usuario quiere acceder al sistema con su cuenta envía al Servidor de Claves un token generado a partir de información de su propio hardware. El Servidor de Claves registra dicho token, asignándole un tiempo de vida (de unas pocas horas). En cada petición de claves el cliente incluye el token como prueba de identidad. La comunicación se cifra bajo SSL para preservar la seguridad tanto del token como de las claves. Si el usuario sigue en el sistema al caducar el tiempo de vida del token éste se prolonga automáticamente. Si el usuario abandona antes pueden adoptarse dos aproximaciones:

- se destruye el token en el servidor. Con esto se pierde la sesión y el usuario puede conectarse al instante desde otro lugar, o bien,
- se mantiene la sesión hasta que expire: el usuario no podrá conectarse desde otro hardware hasta pasado el tiempo de vida.

Si un usuario malintencionado B obtiene los datos de la cuenta de A e intenta acceder mientras A está conectado el sistema rechazará dicha petición al existir una sesión activa. Con esto evitamos que varios usuarios compartan una cuenta

de forma simultánea. La aproximación de impedir el cambio de hardware mientras la sesión no expire puede emplearse para evitar que dos o más usuarios compartan una cuenta a distintas horas, siempre que se ajuste apropiadamente el tiempo de vida.

Cuando hablamos de flujo del contenido nos referimos a la secuencia de paquetes (en este caso UDP) que conforman el contenido a distribuir. Los motivos del uso de UDP se expusieron al inicio de este documento. El Servidor de Contenido cifra el flujo con un algoritmo simétrico fuerte (el PC puede asumir perfectamente el coste del descifrado en tiempo real). Cada paquete se cifra independientemente de los demás y contiene un identificador de clave, indicando inequívocamente cuál se ha empleado en él. Asumimos que el tiempo de vida de estas claves puede ser del orden de minutos, digamos, entre 5 y 20. El cliente va pidiendo periódicamente y por anticipado las claves que va a necesitar al Servidor de Claves, antes de que expire la última que posee.

Las peticiones de claves se realizan bajo SSL, como se ha comentado más arriba. Como respuesta a cada petición el Servidor devuelve varias claves (digamos, suficientes para unos 30 minutos o una hora de reproducción). El cómputo necesario en el servidor para atender una petición consiste en:

1. Procesamiento del protocolo SSL (recepción).
2. Verificación del token en la base de datos del Servidor de Claves.
3. Obtención del juego de claves correspondiente de la base de datos.
4. Procesamiento del protocolo SSL (envío).

Los puntos 2 y 3 se han optimizado usando técnicas de cacheo. Por tanto lo único que consume un tiempo de cómputo reseñable es el procesamiento SSL.

El hecho de que el tiempo de vida de cada clave sea de varios minutos y de que en cada entrega se proporcionen varias de ellas alivia en gran medida la carga que el servidor debe soportar. Aún así, hace surgir un problema: la información multimedia puede ser muy redundante, lo que implica muchos bloques de información similar que resultarán en bloques cifrados iguales. Esta situación es propicia para un ataque basado en estadística.

Existe una solución que permitirá evitar el problema sin incrementar el número de peticiones al servidor: derivar claves mediante resúmentes hash. Se puede obtener un número arbitrario de claves a partir de una inicial, aplicando sucesivamente hash sobre ésta. La idea es usar un nuevo hash como clave cada, digamos, 10 segundos. Como ejemplo, el Servidor de Claves envía k_1 y k_2 al cliente. El Servidor de Contenido comienza a cifrar el flujo con k_1 y, transcurridos los 10 segundos emplea $k_{1,1} = H(k_1)$. Después pasa a $k_{1,2} = H(k_{1,1})$, $k_{1,3} = H(k_{1,2})$ y así sucesivamente. El cambio a k_2 en un momento determinado no implica ninguna operación adicional, ya que el identificador de la clave aparece en el paquete cifrado. El cliente tiene dos opciones a la hora de generar los resúmenes: puede generarlos todos al recibir las claves o según vaya siendo necesario.

El sistema permite tres tipos de canales:

- En abierto: el contenido no se cifra. El identificador de clave de cada paquete recibe el valor 0.
- Suscripción: todo el contenido se cifra. Antes de atender una petición de claves se comprueba que el cliente tiene acceso al canal (ha pagado la suscripción mensual).
- Pay-per-view: emisión alternada de eventos en abierto y cifrados (de pago). También se realiza la comprobación anterior cuando se trata de un evento de pago. Cuando un evento termina el Servidor de Claves descarta todas aquellas claves que pudiera haber entregado ya y genera otras nuevas para el siguiente.

V-C. Líneas futuras de trabajo para eSpectia SPC

El trabajo realizado en el sistema de protección de contenidos no es algo definitivo: este tipo de proyectos son siempre susceptibles de ser mejorados mediante la introducción de nuevas ideas y la puesta a prueba de aquellas ya implementadas. Tenemos en mente algunas cuestiones:

- Implementación del esquema de refresco contínuo de claves mediante hash, introducido en la sección anterior.
- Adopción de un sistema de pago: uso de tarjetas de crédito y plataformas dedicadas, como PayPal.
- Estudio de métodos de acceso alternativos o adicionales al de login y password. Una opción es la autenticación del cliente mediante certificados, pero esto limitaría la portabilidad de la cuenta para aquellos usuarios sin concocimientos informáticos. Sería necesario hacer transparente al usuario los procesos de obtención e instalación del certificado.
- Estudio de métodos para atender grandes cantidades de carga en el Servidor de Claves, en el caso de que exista una red masiva de clientes. La opción más clara es implementar una redundancia de servidores, lo cual requeriría comunicación entre ambos para mantener el sistema en un estado consistente. Esta aproximación, además, aportaría tolerancia a fallos al sistema, el cual es otro tema a tener en cuenta.

VI. Conclusiones y líneas de investigación

A lo largo de este trabajo hemos presentado el paradigma IPTV y las circunstancias y tecnologías previas que han propiciado su desarrollo. Nos hemos centrado principalmente en los sistemas de acceso condicional, necesarios para su desarrollo comercial.

Hemos visto cómo los primeros esquemas propuestos son adaptaciones fieles de aquellos empleados en la televisión digital (DVB), tecnología que la IPTV viene a sustituir. Sin embargo, el uso de Internet como canal de comunicación permite un servicio mucho más flexible y ofrece más posibilidades de interacción, algo a lo que los CAS deben adaptarse si la IPTV ha de tener éxito.

En primer lugar hemos descrito el esquema inicial para DVB propuesto por la International Telecommunication Union en 1992. Posteriormente se han expuesto esquemas evolucionados a partir de las soluciones de Lee y Tu. Estas evoluciones exploran la idea de distribuir los canales en grupos, de forma que

el refresco de claves se haga de forma separada, reduciendo en gran medida la dimensión de dichas operaciones.

A partir de ahí se han revisado propuestas más propias de la IPTV que de la DVB. Por un lado se ha tratado el problema de la autenticación del cliente, necesaria si se van a ofrecer servicios más allá del simple visionado del flujo multimedia. Aquí juegan un papel muy importante las técnicas tradicionales de criptografía de clave pública: uso de certificados para la verificación de firmas digitales y presencia de Centros de Autorización que los emitan.

También hemos entrado en el campo de la persistencia del contenido, algo que ofrecen los nuevos receptores hardware. Si asumimos que el cliente va a tener la posibilidad de grabar el contenido y redistribuirlo entonces las reglas del juego cambian: es necesario proteger la información tanto durante el acceso como durante sus posteriores reproducciones. Para esto se emplean, a día de hoy, dos alternativas. La primera es el uso de licencias: documentos que especifican los derechos de acceso del poseedor del contenido. La segunda es el watermarking: el reproductor verifica la presencia de una marca de agua (watermark) para reproducir la información multimedia.

Finalmente hemos visto cómo el empleo de metadatos (normalmente expresados en formato XML) permite el acceso al contenido de formas más avanzadas que la lineal tradicional. En este caso los CAS también tienen algo que decir, en el sentido de autentificar y proteger la integridad de los metadatos, así como de regular su uso.

La conclusión final de este documento es que los sistemas de acceso condicional para IPTV pueden y deben alejarse aún de aquellos esquemas iniciales para DVB. Debido a la aparición de nuevas formas de acceso al contenido los esquemas futuros se acercarán más al modelo DRM para regular el acceso: empleo de licencias, certificados, metadatos, etc. La distancia entre CAS y DRM se irá acortando poco a poco, hasta la obtención de esquemas de acceso globales que cubrirán todos los posibles escenarios. Además, su complejidad computacional aumentará, gracias al incremento de la capacidad de los dispositivos receptores (bien hardware específico, bien software corriendo en PCs o, preferiblemente, centros multimedia).

Creemos además que las soluciones más fiables pasarán inevitablemente por delegar la seguridad en el servidor, no confiando en el cliente que, hardware o software, puede ser modificado con intenciones fraudulentas. Esto implica no entregar al dispositivo cliente información que pueda servirle para la reproducción si no tiene las credenciales apropiadas. Algunos de los esquemas revisados nos presentan al cliente tomando decisiones del tipo "Si se cumple la condición reproducir. Si no, abortar". En nuestra opinión el cliente debería encontrarse ante la siguiente situación: "Si dispongo de las credenciales apropiadas obtendré la información necesaria para reproducir el contenido. Si no, no se me facilitará".

Finalmente se ha mostrado un adelanto del sistema de acceso condicional en desarrollo surgido de la colaboración entre el Departamento de Arquitectura de Computadores y Electrónica de la Universidad de Almería y la empresa IPTV Solutions.

VI-A. Líneas futuras de investigación

A la hora de plantear la dirección de una investigación futura es necesario reparar en que la aparición de la comunicación bidireccional cambia las reglas del juego: el esquema propuesto por la ITU en 1992 y sus derivados se diseñaron para un escenario muy concreto, como era el de la DVB, y quizás hoy son demasiado complejos. En cambio nos parece una buena idea profundizar en el fenómeno del acercamiento entre CAS y DRM: creemos que esta va a ser la tendencia predominante en el futuro. Ya hemos remarcado que ambas aproximaciones deberán unirse para dar lugar a un esquema de control de acceso multimedia todoterreno. Las siguientes ideas, por tanto, nos parecen interesantes:

- Ahondar en el estudio del empleo de metadatos. Éstos aportan un valor añadido a los contenidos. Además, pueden emplearse en otras situaciones, como son los procesos de acceso al sistema e interacción con él: alta de usuario, login, preferencias del usuario, servicios, etc.
- Preparar el terreno para la llegada masiva de servicios con los que el espectador puede interactuar. Éstos también podrán ser de pago, por lo que el acceso a ellos deberá ser controlado.
- Con el desarrollo de los *smart devices* (dispositivos inteligentes tales como agendas electrónicas, móviles de última generación, centros multimedia, etc.) la distribución del contenido debe adaptarse a todos los posibles receptores. Es por tanto necesario buscar esquemas de control de acceso multidispositivo, que necesiten poca capacidad de cómputo y empleen preferentemente estándares tales como XML.
- El punto anterior fuerza al empleo de herramientas de autentificación estandarizadas: por ello vemos clave el empleo de firmas digitales, certificados, Autoridades Certificadoras, etc.

Otro aspecto importante es desarrollo de soluciones que no confíen la seguridad al cliente, como se mencionó anteriormente. Por otra parte, si las redes peer-to-peer van a seguir disfrutando de buena salud en el futuro e incluso encontrado aplicaciones comerciales (tales como la P2PTV) también necesitarán esquemas de control de acceso específicos.

Finalmente, el empleo de certificados digitales para la identificación personal es cada día más corriente. Un claro ejemplo de esto es la reciente introducción en España del Documento Nacional de Identidad electrónico (DNIe). Cualquier servicio telemático que lo acepte, tanto de la Administración Pública como del sector privado, necesitará disponer de un control de acceso compatible con él (ISO 7816, [2]). Además, creemos que el desarrollo de un esquema de control de acceso IPTV compatible sería una buena idea.

DEFINICIONES

Authorization Center

Sistema que genera claves y certificados en un sistema para facilitar la comunicación segura y/o autenticada entre miembres de éste.

CA

Certification Authority. Entidad global de confianza que proporciona certificados empleados en comunicaciones web seguras y/o autenticadas. Véase la sección III-E1.

CAS

Conditional Access System. Sistema de restricción de acceso a contenidos (generalmente multimedia). Este tipo de sistemas sólo permite el acceso si el solicitante verifica uno o varios requisitos. Véase la sección IV.

CSA

Common Scrambling Algorithm. Algoritmo de cifrado simétrico ampliamente utilizado en la *DVB*. Véase la sección IV-B.

Centro de Autenticación

Véase *Authorization Center*.

DBHS

Digital Broadcasting Based on Home Servers. Paradigma *IPTV* en el que los receptores disponen de dispositivos de almacenamiento masivo, lo que les permite grabar el contenido recibido. Véase la sección IV-F.

DBRM

Digital Broadcasting Receiver using Metadata. Paradigma *IPTV* en el que el contenido distribuido se describe mediante metadatos. Véase la sección IV-F1.

DRM

Conjunto de soluciones tecnológicas cuyo fin es restringir el uso de obras sujetas a derechos en plataformas digitales. Se relaciona con la protección anticopia y el acceso a arhivos multimedia. Véase la sección I.

DVB

Digital Video Broadcasting. Televisión digital, en oposición a la televisión analógica tradicional. Véase la sección II.

ECM

Entitlement Control Message. Véase la sección IV-A.

EMM

Entitlement Management Message. Véase la sección IV-A.

Fingerprint

Tipo de *watermark* que un usuario deja en el contenido que reproduce. Se emplea para rastrear el origen de la distribución de un contenido en el caso de que ésta sea ilegal. Véase la sección IV-E.

Firma Digital

Método de autenticación de comunicantes y documentos. Véase la sección III-C.

Hash

Función que genera una salida de longitud fija para cualquier conjunto de datos de entrada. La probabilidad de que dos conjuntos de datos obtengan la misma salida es muy baja. Se emplea en criptografía. Véase la sección III-D.

IPTV

Internet Protocol Television. Paradigma de televisión por el cual la señal se distribuye a través de Internet. Puede ofrecer también otro tipo de servicios. Véanse las secciones I y II

ITU

International Telecommunication Union. Organismo de las Naciones Unidas encargado de regular las telecomunicaciones, a nivel internacional, entre las distintas administraciones y empresas operadoras. [19]

KMAS

Key Management and Authorization Server. Servidor que proporciona claves de cifrado a las partes que intervienen en un *CAS*. Véase la sección IV-B.

P2P

Red de ordenadores sin clientes ni servidores fijos. En lugar de ello, cada nodo se comporta como cliente o servidor en un determinado momento, o simultáneamente. Su uso más extendido es el intercambio de archivos.

P2PTV

Rama de la *IPTV* que emplea una red *P2P* para distribuir el flujo multimedia entre los clientes. Véase la sección V-A.

Pairing

Técnica por la cual un receptor (generalmente *DVB*) sólo acepta la smart-card de un usuario determinado. Véase la sección II.

Pay-per-view

Modo de visionado por el cual el espectador debe realizar un pago antes de tener acceso a un programa concreto. Una vez terminado el acceso queda bloqueado hasta nuevo pago. Véase la sección IV-C.

Reproducción lineal

Modo de visionado similar al de la televisión tradicional. Sólo puede verse el contenido que esté siendo emitido en el momento, sin posibilidad de avanzar o retroceder en la reproducción, o repetir escenas. Véase la sección I.

Resumen

Véase *hash*.

Scrambling

Método de cifrado ligero, eficiente pero poco seguro. Véase la sección III-A1.

Smart-card

Dispositivo en forma de tarjeta con circuitería capaz de procesar información. Véase la sección II.

Set-top box

Dispositivo encargado de recibir una señal multimedia del canal de distribución y prepararla para el dispositivo reproductor (generalmente un televisor). Se emplea en *DVB* e *IPTV*. Véase la sección II.

SSL

Protocolo que proporciona comunicaciones cifradas y autenticadas. Véase la sección III-E2

Suscripción

Modo de visionado por el cual el espectador debe realizar un pago para obtener acceso a un canal determinado durante un período de tiempo extenso (generalmente un mes). Véase la sección IV-C.

TDT

Televisión Digital Terrestre. Es una modalidad de *DVB* empleada en Europa, Australia, África y algunos países de Suramérica.

Trick-play

Modo de reproducción en el que puede saltarse hacia atrás o hacia delante y repeteir escenas. Véase la sección I.

TV-Anytime Forum

Asociación de organizaciones cuyo objetivo es desarrollar especificaciones que permitan el almacenamiento digital de servicios audiovisuales y de otro tipo, en plataformas de usuario, basándose en el mercado de masas. [4]

UDP

Protocolo de comunicación IP integrado en la capa de transporte del modelo OSI. Véase la sección II.

VoD

Video on Demand. Modalidad de visionado que permite al espectador elegir desde su terminal el contenido que desea ver a cualquier hora.

Watermark

Técnica que consiste en almacenar información adicional en un archivo multimedia. La huella dejada suele ser inapreciable. Véase la sección III-F.

XML

eXtended Markup Language. Lenguaje de etiquetado cuya función principal es describir datos, no mostrarlos (como HTML) o procesarlos (como los lenguajes de programación). Véase la sección IV-F1.

REFERENCIAS

[1] C2 block cipher specification. http://www.4centity.com/data/tech/spec/C2_100.pdf.

[2] Estándar ISO 7816. http://www.info-ab.uclm.es/labelec/Solar/Otros/ISO_7816/index.htm.

[3] Introduction to SSL. http://docs.sun.com/source/816-6156-10/contents.htm.

[4] Tv-Anytime Forum. http://www.tv-anytime.org.

[5] World Wide Web Consortium. http://www.w3.org.

[6] XML Encryption syntax and processing. http://www.w3.org/TR/2002/REC-xmlenc-core-20021210.

[7] XML Signature syntax and processing. http://www.w3.org/TR/2008/REC-xmldsig-core-20080610.

[8] ITU-R Rec. BT.810. Conditional-Access broadcasting systems. 1992.

[9] M. Becker and A. Desoky. A study of the DVD Content Scrambling System (CSS) algorithm. 2004.

[10] M. Bellare and T. Kohno. Hash function balance and its impact on birthday attacks. *Advances in cryptology - EUROCRYPT 2004*, 2004.

[11] G. H. Chiou and W. T. Chen. Secure broadcasting using the secure lock. *IEEE Transactions on Software Engineering*, 15(8):929–934, 1989.

[12] A. M. Eskicioglu. Multimedia security in group communications: recent progress in key management, authentication, and watermarking. *Multimedia Systems*, 9(3):239–248, 2003.

[13] W. Lee. Key distribution and management for conditional access system on DBS. *Proc. of international conference on cryptology and information security*, pages 82–86, 1996.

[14] B. Liu, W. Zhang, and T. Jiang. A scalable key distribution scheme for conditional access system in digital pay-tv system. *IEEE Transactions on Consumer Electronics*, 50(2), 2004.

[15] J. Liu, C. Yang, and J. Tian. A novel conditional access architecture for TV service protection. *International Conference on Computational Intelligence and Security Workshops*, 2007.

[16] D. Minoli. *IP Multicast with applications to IPTV and Mobile DVB-H*. Wiley-Interscience, 2008.

[17] G. O'Driscoll. *Next generation IPTV services and technologies*. Wiley-Interscience, 2008.

[18] N. M. Piqueras Romero. *Sistema de acceso condicionado para canales multimedia en entornos de streaming en vivo. Proyecto Fin de Carrera*. Universidad de Almería, 2008.

[19] I. Teleommunication Union. http://www.itu.int.

[20] G. A. Tsihrintzis and L. C. J. (Eds.). *Multimeda Services in Intelligent Environments*. Springer, 2008.

[21] F. Tu, C. S. Laih, and H. H. Tung. On key distribution management for conditional access system on pay-tv system. *IEEE Transactions on Consumer Electronics*, 45:151–158, 1999.

[22] Y.Nishimoto, A. Baba, T. Kimura, H. Imaizumi, and Y. Fujita. Advanced conditional access system for digital broadcasting receivers using metadata, 2007.

[23] Y.Nishimoto, A. Baba, T. Kurioka, and S.Namba. A digital rights management system for digital broadcasting based on home servers. *IEEE Transactions on Broadcasting*, 52(2), 2006.

[24] M. Zhu, M. Zhang, X. Chen, D. Zhang, and Z. Huang. Hierarchical key distribuition scheme for conditional access system in DTV broadcasting. *International Conference on Computational Intelligence and Security*, 2:1532–1535, 2006.