Esquemas dinámicos de distribución de claves en redes peer-to-peer multimedia

J.A.M. Naranjo,[1] J.A. López-Ramos[2] y L.G. Casado[3]

Resumen— **Este artículo presenta dos esquemas de distribución de claves para redes peer-to-peer de streaming multimedia. El punto fuerte de ambos esquemas reside en que permiten balancear seguridad y eficiencia a voluntad, lo cual se consigue variando dinámicamente el número de niveles de la jerarquía de claves empleada. Los cambios son realizados por el Servidor de Claves en tiempo real en función del tamaño de la audiencia. Una jerarquía más larga soporta mayores audiencias, pero ofrece una seguridad menor frente a ataques por fuerza bruta y basados en estadística. Lo contrario sucede con jerarquías más cortas.**

Palabras clave— **Distribución de claves, peer-to-peer, streaming multimedia.**

I. Introducción

LOS esquemas de distribución de claves diseñados para redes peer-to-peer (P2P) de streaming multimedia explotan el hecho de que para un canal de TV cada peer puede compartir las claves de cifrado a la vez que reproduce el contenido. Esta ventaja se traduce en una gran reducción de la carga de trabajo y comunicaciones que debe soportar el Servidor de Claves. Sin embargo, lidiar con enormes audiencias sigue siendo uno de los problemas más importantes en este campo: el proceso de refresco inevitablemente marca una cota superior al número de clientes que puede abastecerse. La solución más común para redes multimedia no peer-to-peer es dividir la audiencia en grupos de usuarios [1].

Otro de los problemas que encontramos en este campo es el de manter las claves secretas ante observadores no autorizados durante todo su viaje desde el Servidor de Claves hasta el último peer. Por el momento se han propuesto algunas soluciones, que mostramos a continuación. En [2] la distribución de claves se realiza en forma de árbol. Cada nodo comparte una clave secreta (KEK: *Key Encryption Key*) con sus hijos. Esta clave se emplea para cifrar las comunicaciones entre padre e hijos. La llegada de un nuevo peer implica el establecimiento de nuevas KEKs sucesivamente desde el peer hasta el nodo raíz (el Servidor de Claves). Un proceso similar debe realizarse cuando el peer abandona la red. Todo esto puede conducir a pérdidas en la calidad del servicio (QoS: *Quality-of-Service*) en momentos de llegadas o abandonos masivos, tales como el inicio y el fin de eventos de gran interés.

En [3] se describe otro método de distribución de claves para redes con forma de árbol. En [4] el intercambio de claves entre peers requiere el establecimiento de relaciones de confianza entre ellos, lo que implica autenticación mutua bajo un canal seguro. Esto consume tiempo y ancho de banda.

Este artículo presenta dos esquemas de distribución de claves para redes peer-to-peer de streaming multimedia que tratan los problemas mencionados en los dos primeros párrafos. El primer esquema, *Complete Key Distribution Scheme*, es una extensión a la solución centralizada tradicional [5][6][7] que saca partido de las características de las redes peer-to-peer. El segundo, *Share Based Distribution Scheme*, emplea técnicas *secret sharing* y no requiere procesos de cifrado/descifrado para las comunicaciones entre peers. Ambos esquemas usan una jerarquía de claves que puede variar dinámicamente desde uno hasta tres niveles en el caso de *Complete Key*, y entre uno y dos en el caso de *Share Based*. El objetivo de esta variación dinámica es ofrecer un compromiso entre seguridad y tamaño máximo de audiencia. Cada jerarquía de claves se adapta mejor a un escenario distinto, dependiendo del tamaño de la audiencia. Ambos esquemas, además, evitan la necesidad de establecer claves de cifrado entre peers.

II. Una jerarquía de claves dinámica

Las jerarquías de claves no son algo nuevo: los sistemas de acceso condicional para TV las emplean frecuentemente en su forma estática. Los esquemas propuestos en este artículo introducen la posibilidad de incrementar o decrementar el número de niveles empleados en función del tamaño de la audiencia. Cuanto menor sea ésta última más corta será la jerarquía, y viceversa.

Añadir una nueva clave en lo más alto de la jerarquía en un momento determinado permite refrescar las que quedan por debajo con mayor frecuencia. Esto se debe al hecho de que la clave más alta debe refrescarse contra el Servidor de Claves, pero las demás pueden renovarse empleando la red peer-to-peer, como se mostrará más adelante. La frecuencia de refresco decrece conforme recorremos la jerarquía (niveles mayores). Existe un inconveniente: un atacante dispondrá de más tiempo para romper la clave más alta, debido a que permanece sin refrescar durante largos períodos de tiempo. Los cambios dinámicos en la jerarquía persiguen eliminar este riesgo cuando una audiencia pequeña lo permita, ya que al acortar la jerarquía conseguimos una clave superior que se refresca con una frecuencia mayor. Las siguientes secciones muestran el funcionamiento de ambos esquemas.

[1]Dpto. de Arquitectura de Computadores y Electrónica, Univ. de Almería, e-mail: jmn843@alboran.ual.es.

[2]Dpto. de Álgebra y Análisis Matemático, Univ. de Almería, e-mail: jlopez@ual.es.

[3]Dpto. de Arquitectura de Computadores y Electrónica, Univ. de Almería, e-mail: leo@ual.es.

III. Complete Key Distribution Scheme

La idea tras este esquema es distribuir la clave que cifra el stream (llamada CW, por *Control Word*, para respetar la nomenclatura más común) en un mensaje (CW-message) empleando para ello la propia red peer-to-peer. Este enfoque se adapta especialmente bien a redes de tipo malla, a la vez que introduce muy poco overhead de comunicaciones y consigue un nivel de seguridad suficiente. La jerarquía de claves puede variar entre uno y tres niveles: los tres modos se muestran a continuación. La Figura 1 muestra el cambio entre modos desde el punto de vista de los peers (el proceso de cambio se inicia en $t = 0$ y termina en $t = f$).

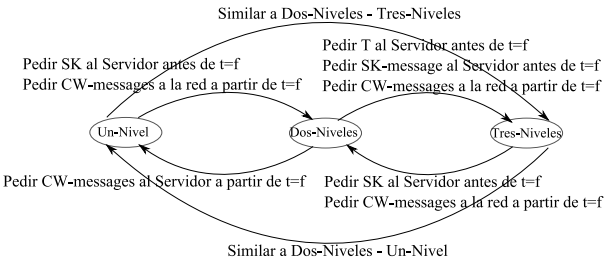

Fig. 1. Cambios de modo en Complete Key.

A. Modo Un-Nivel

Cuando el esquema trabaja en este modo emplea una arquitectura cliente-servidor centralizada para la distribución de CW, la cual se refresca periódicamente con una frecuencia que puede variar desde unos pocos segundos hasta varios minutos. El refresco se realiza bajo TLS, y requiere la autenticación de cada peer. Las conexiones cliente-servidor no explotan los beneficios de una red peer-to-peer. Para incrementar el rendimiento se pueden enviar en una misma comunicación varios refrescos sucesivos de CW: esto reduce el número de comunicaciones necesarias entre cada peer y el Servidor de Claves.

Este modo sólo permite abastecer a una audiencia muy reducida, ya que las frecuentes comunicaciones TLS imponen un overhead en tiempo y computación. La privacidad de las claves es alta, gracias a la elevada tasa de refresco que evita ataques estadísticos contra CW. Además, no es necesario que el peer almacene información criptográfica sensible durante largos períodos de tiempo. El principal riesgo es la distribución volutaria de CW por parte del cliente.

B. Modo Dos-Niveles

En este modo los peers ya no reciben CW bajo conexiones directas con el Servidor de Claves: ahora el Servidor encripta CW con una clave de segundo nivel, que hemos llamado SK (por *Service Key*). El resultado de la encriptación se encapsula en un CW-message que se inyecta en la red peer-to-peer. Los propios peers distribuyen entonces el CW-message. Dicho mensaje puede contener sucesivos refrescos de CW, de forma similar al modo Un-Nivel. El tiempo de vida de SK está en el orden de varias horas: cuando su expiración esté próxima los peers deben solicitar bajo TLS al Servidor de Claves una nueva SK.

El tamaño de la audiencia que puede abastecerse es mayor que en el modo anterior, ya que los peers no necesitan comunicarse con el Servidor de Claves durante varias horas (excepto un pequeño grupo de nodos que se comunica constantemente con el Servidor: éstos son los que reciben en primera instancia los CW-messages para iniciar su distribución por la red). Además, las comunicaciones entre peers se realizan bajo canales en claro (los CW-messages ya están encriptados), por lo que no sufren ningún tipo de overhead. Respecto a la privacidad de las claves, el riesgo de que SK se vea comprometida es bajo, dado que los peers la reciben bajo TLS del Servidor de Claves.

C. Modo Tres-Niveles

En este caso se emplea una clave de nivel tres para distribuir SK. La nueva clave recibe el nombre de T (por *authorization Token*). El Servidor de claves encripta SK y la encapsula en un mensaje llamado SK-message, el cual es inyectado en la red peer-to-peer. T se renueva cada varios días (o incluso semanas) mediante TLS contra el Servidor de Claves.

El largo tiempo de vida de T permite aumentar enormemente el período de tiempo durante el cual los clientes pueden descifrar el contenido sin comunicarse con el Servidor de Claves. Ello posibilita abarcar grandes audiencias con este modo. La privacidad de las claves, al contrario, disminuye por la misma causa: un atacante dispondrá de mucho tiempo para descifrar T y así poder reproducir el contenido. El tiempo de vida de T puede ajustarse a voluntad para dificultar la labor del atacante (con la desventaja de perder tamaño de audiencia).

D. Cambios entre modos en Complete Key

Cada cambio se inicia a decisión del Servidor de Claves cuando el tamaño de la audiencia excede o cae por debajo de un umbral dado (el tamaño de la audiencia puede aproximarse mediante *heartbeats* enviados por cada peer). Los cambios no tienen por qué limitarse a modos consecutivos: si el tamaño de la audiencia varía (o se espera que lo haga) de forma ingente puede saltarse el modo Dos-Niveles.

La información necesaria para realizar un cambio de modo se incluye en los CW-messages. Dicha información consta de tres campos: el modo en el que el sistema corre actualmente (campo *Current Mode*), cuántos CW-messages quedan por emitir antes del cambio (campo *Left*) y qué modo se empleará entonces (campo *Next Mode*). Por lo tanto, un CW-message contiene:

- Una o más CWs sucesivas (no encriptadas en el modo Un-Nivel, encriptadas con SK en Dos-Niveles y Tres-Niveles).
- Campo *Current Mode*.
- Campo *Left*.
- Campo *Next Mode*.

Es necesaria la inclusión de *Current Mode* para que los peers que se unen conozcan en qué modo está trabajando el sistema.

E. Consideraciones sobre seguridad y tamaño de audiencia en Complete Key

El aumento de la longitud de la jerarquía de claves favorece también el aumento del tamaño máximo de audiencia alcanzable. Para evitar cuellos de botella al refrescar la clave más alta, las peticiones al Servidor de Claves deberían realizarse en un instante de tiempo aleatorio previo a la expiración de la clave. En ese caso es muy recomendable que las peticiones sigan una distribución uniforme, para repartir las peticiones en un intervalo de tiempo previo. Esto hace la carga de trabajo más soportable para el Servidor de Claves y favorece la eficiencia global del sistema.

La seguridad se basa en cifrado simétrico para la distribución de claves dentro de la red peer-to-peer, y en comunicaciones TLS contra el Servidor de Claves. Si las claves se generan de forma criptográficamente segura entonces será difícil romperlas en un tiempo lo suficientemente breve como para poder aprovecharlas.

A pesar de todo esto, y como ya se mencionó en la Sección II, las claves con una vida muy larga son más débiles. El uso de una jerarquía dinámica proporciona más seguridad a la distribución de claves si los cambios se planean cuidadosamente. Como medida adicional el Servidor de Claves puede firmar cada CW-message y SK-message para asegurar su autenticidad.

IV. Share-Based Key Distribution Scheme

Esta aproximación es similar a *Complete Key*, pero difiere en el hecho de que emplea técnicas de *secret sharing* para realizar la distribución de claves.

Las técnicas *secret sharing* pueden definirse de la siguiente forma:

Distribuir un secreto S en n participaciones o shares, de forma que se necesiten $t \leq n$ shares para reconstruirlo, siendo esto imposible con un número de shares menor que t (denominado umbral).

El *Método Umbral de Shamir* es quizás el más conocido entre los distintos métodos existentes. Puede encontrarse una descripción detallada en [8][9]. Se basa en la interpolación de puntos en el plano para conseguir un polinomio único. Puede calcularse muy rápidamente gracias a la Transformada Rápida de Fourier [13][14], y es seguro en el sentido de que "su seguridad no se apoya en asunciones no probadas (como la dificultad de resolver ciertos problemas numéricos)" [8]. Existen otros esquemas equivalentes, como puede verse en [10][11][12].

Por todo lo explicado este esquema se adapta mejor a topologías multi-tree. Además, presenta dos características ventajosas: (1) trabaja en dos modos en lugar de tres, pero alcanza tamaños de audiencia similares a los de *Complete Key*, y (2) el proceso de distribución de claves requiere muy poco esfuerzo computacional. Esto lo hace apropiado para pequeños dispositivos tales como set-top boxes, smartphones y PDAs.

A. Modo Un-Nivel

El stream se encripta con CW, la cual es refrescada con una frecuencia de segundos o incluso minutos. El refresco se realiza como sigue: el Servidor de Claves genera un número determinado de *shares* y las encapsula en mensajes distintos, sin encriptar. Las shares reciben el nombre de SSs (*Secondary Shares*), y los mensajes, SS-messages. Estos últimos se inyectan en la red peer-to-peer. Adicionalmente, cada peer se autentica contra el Servidor de Claves y recibe vía TLS un fragmento de información denominado MS (*Master Share*), que debe permanecer secreto. CW puede calcularse tal y como se muestra en la Figura 2. F es una función de una vía, no definida en este artículo. El tiempo de vida de MS es de varias horas. La privacidad de CW se consigue manteniendo secreta MS y refrescándola para evitar ataques por fuerza bruta o basados en estadística.

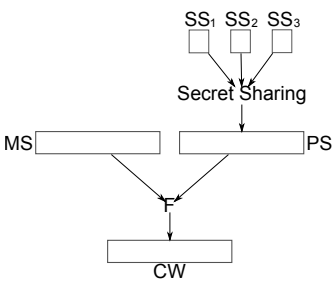

Fig. 2. Construcción de CW en Share-Based.

B. Modo Dos-Niveles

CW se distribuye de forma similar a como se hace en el modo Un-Nivel. MS, en cambio, se distribuye a través de la red peer-to-peer: el Servidor de Claves encripta MS con una nueva clave, T (*authorization Token*, de nuevo). El resultado se inyecta en la red, contenido en un nuevo tipo de mensaje, MS-message. Los peers distribuyen el mensaje según sea necesario. El tiempo de vida de MS dura varias horas. T se renueva bajo TLS contra el Servidor de Claves tras varios días (o incluso semanas).

La larga vida de T permite alcanzar grandes audiencias. La seguridad, por otra parte, es menor debido a la misma razón. Para encontrar un compromiso la vida de T puede acortarse según sea conveniente, a costa de perder tamaño de audiencia.

C. Cambios entre modos en Complete Key

Los cambios se llevan a cabo por las mismas razones y de la misma forma que en la Sección III-D. La Figura 3 muestra los detalles desde el punto de vista de los peers. El proceso comienza en $t = 0$ y finaliza en $t = f$.

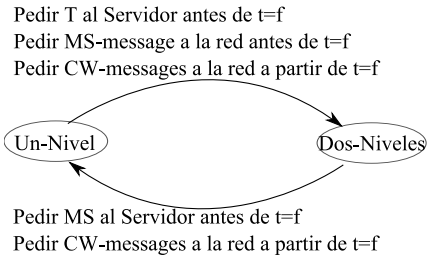

Fig. 3. Cambios de modo en Share-Based.

D. *Consideraciones sobre seguridad y tamaño de audiencia en Share-Based*

En relación con la eficiencia, el proceso de refresco requiere muy pocos recursos computacionales, tal y como se menciona en la Sección IV. Más aún, el hecho de que los SS-messages no estén encriptados ayuda en ese aspecto.

Hay un punto importante a considerar: las técnicas de secret sharing son suficientemente seguras, pero F debería elegirse con cuidado. Si se emplea una función hash su salida debería ser suficientemente larga como para soportar ataques por fuerza bruta mediante un período de tiempo similar a (al menos) la vida de SK. Finalmente, si el Servidor de Claves firma cada SS-message y MS-message entonces la autenticidad de estos últimos queda asegurada.

Las consideraciones relacionadas con el tamaño máximo de la audiencia son similares a las expuestas en la Sección III-E.

V. Conclusiones

Este artículo presenta dos nuevos esquemas de distribución de claves para redes peer-to-peer de streaming multimedia. Ambos esquemas comparten una cualidad: la posibilidad de variar dinámicamente el número de niveles de la jerarquía de claves. Ello permite ofrecer la privacidad más alta posible en cada momento, dependiendo del tamaño de la audiencia.

El primer esquema, *Complete Key Distribution Scheme*, es más apropiado para redes con topología de malla. El segundo, *Share-Based Key Distribution Scheme*, se adapta mejor a multi-trees con gran audiencia que dependan de hardware con poca capacidad computacional. Esta última característica lo hace especialmente apropiado para set-top boxes, smartphones y PDAs.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Ciencia e Innovación (TIN2008-01117), la Junta de Andalucía (P06-TIC-1426, P08-TIC-3518, FQM 0211 and TEC2006-12211-CO2-02) y el Fondo Europeo de Desarrollo Regional (FEDER).

Referencias

[1] Y.-L. Huang, S. Shieh, F.-S. Ho and J.-C. Wang. "Efficient key distribution schemes for secure media delivery in pay-tv systems." *Multimedia, IEEE Transactions on.* vol. 6, no. 5, pp. 760-769, Oct. 2004.

[2] X. Liu, H. Yin, C. Lin and Y. Deng. "Efficient key management and distribution for peer-to-peer live streaming system." 2007.

[3] R. Song, L. Korba and G. Yee. "A scalable group key management protocol." *Communications Letters, IEEE.* vol. 12, no. 7, pp. 541-543, July 2008.

[4] F. Qiu, C. Lin and H. Yin. "EKM: an efficient key management scheme for large-scale peer-to-peer media streaming." *Proceedings of PCM 2006.* pp. 395-404, 2006.

[5] ITU-R Rec. BT.810. Conditional-Access Broadcasting Systems. *International Telecommunication Union.* 1992.

[6] W. Lee. "Key distribution and management for conditional access system on DBS." *Proceedings of International Conference on Cryptology and Information Security.* pp. 82-86. 1996.

[7] F.K. Tu and C.S. Laih and H.H. Tung. "On key distribution management for conditional access system on pay-TV system." *IEEE Transactions on Consumer Electronics."* vol. 45. pp. 151-158. 1999.

[8] A. J. Menezes, P. C. V. Oorschto, S. A. Vanstone and R. L. Rivest. "Handbook of applied cryptography." 1997.

[9] B. Schneier. *Practical Cryptography.* Wiley & sons, 2003.

[10] E. F. Brickell and D. M. Davenport. "On the classification of ideal secret sharing schemes." *CRYPTO '89: Proceedings on advances in cryptology.* pp. 278-285, Springer-Verlag New York, 1989.

[11] L. Chen, D. Gollmann, C. J. Mitchell and P. R. Wild. "Secret sharing with reusable polynomials." *ACISP '97: Proceedings of the Second Australasian Conference on information security and privacy.* pp. 183-193, Springer-Verlag, 1997.

[12] M. Numao. "Periodical multi-secret threshold cryptosystems." *ASIACRYPT '99: Proceedings of the International Conference on the theory and applications of cryptology and information security.* pp. 363-377, Springer-Verlag, 1999.

[13] R. E. Blahut. "A universal Reed-Solomon decoder." *IBM Journal of Research and Development,* vol. 28, no. 2, pp. 150-158, 1984.

[14] R. J. McEliece and D. V. Sarwate. "On sharing secrets and Reed-Solomon codes." *Communications of the ACM,* vol. 24, no. 9, 1981.