ESCOLA TÈCNICA SUPERIOR D'ENGINYERIA DE TELECOMUNICACIÓ DE BARCELONA

Seguridad en Redes de Banda Ancha. Contribución al Diseño y Evaluación de un Sistema de Seguridad para la RDSI-BA

TESIS DOCTORAL

Tesis Doctoral presentada en la Universitat Politècnica de Catalunya para la obtención del título de Doctor Ingeniero de Telecomunicación

Autor: **Jordi Forné Muñoz**
Director: **Dr. José Luis Melús Moreno**

CAPÍTULO 5

Gestión de Claves

5.1 Introducción

En capítulo 3 se ha propuesto una arquitectura que permite que diferentes aplicaciones negocien servicios de seguridad para las comunicaciones entre terminales multimedia a través de la RDSI-BA. Factores para garantizar la seguridad del sistema son la autentificación mutua entre terminales y la gestión de claves, así como la integridad de los parámetros de seguridad negociados.

Por otra parte, en el capítulo 4 se mostró la necesidad de un sistema eficiente de gestión de claves para hacer económicamente viable la implantación de una arquitectura de seguridad en una red de comunicaciones.

Por todo ello en este capítulo se propone un protocolo eficiente de gestión de claves para la RDSI-BA. Para garantizar su seguridad se parte del conocido protocolo de autentificación de Needham-Schroeder [NEE78], y se introducen las modificaciones necesarias para permitir la gestión de claves y conseguir la máxima eficiencia sin degradar su seguridad[1].

[1] El protocolo de Needham-Schroeder es ampliamente conocido desde hace bastante tiempo (desde 1978) y hasta la fecha no ha sido públicamente violado.

5.2 Gestión de Claves

En este apartado se comentan aspectos de la gestión de claves que serán útiles para la propuesta de un protocolo para la RDSI-BA. En primer lugar se presentan aspectos generales tanto de la gestión de claves como de la autentificación, factor este último clave en los protocolos de gestión de claves. Por último se señalan aspectos concretos de la gestión de claves a considerar en la RDSI-BA, donde se concluye que la criptografía de clave pública es la mejor opción.

5.2.1 Aspectos Generales de la Gestión de Claves

En el documento ISO 7498-2 [ISO88], que trata sobre la arquitectura de seguridad del modelo de referencia OSI, se define el concepto de gestión de claves como la generación, almacenamiento, distribución, destrucción, archivo y aplicación de las claves de acuerdo con una política de seguridad. Una política de seguridad es un conjunto de reglas que delimitan y controlan las actividades relevantes en cuanto a seguridad de sujetos o entidades. Por tanto, define lo que se entiende como seguridad dentro de un sistema y los procedimientos mediante los que se consigue esta seguridad. El ámbito donde una política de seguridad es aplicable es denominado dominio (pudiendo esta política de seguridad definir las posibles interrelaciones con otros dominios).

Sin duda el problema central en todo sistema de gestión de claves son los procedimientos de distribución de éstas. Esta distribución debe efectuarse previamente a la comunicación. Los requisitos específicos en cuanto a seguridad de esta distribución dependerán de para qué y cómo van a ser utilizadas las claves. Así pues, será necesario garantizar la identidad de su origen, su integridad y, en el caso de claves secretas, su confidencialidad.

Las consideraciones más importantes para el diseño de un sistema de gestión de claves son el tipo de ataques que lo amenazan y la arquitectura del sistema.

Normalmente, es necesario que la distribución de claves se lleve a cabo sobre la misma red de comunicación donde se está transmitiendo la información a proteger. Esta distribución es automática, y la transferencia suele iniciarse con la petición de clave por parte de una entidad a un Centro Distribuidor de Claves o la otra entidad involucrada en la comunicación. La alternativa es una distribución manual (mediante el empleo de correos seguros, por ejemplo), independiente del canal de comunicación. Esta última alternativa implica un alto coste económico y un tiempo relativamente largo para llevarse a cabo, lo que la hace descartable en la mayoría de las situaciones.

La distribución segura de claves sobre canal inseguro requiere protección criptográfica y, por tanto, la presencia de otras claves. En cierto punto se requerirá protección no criptográfica de algunas claves (que llamaremos claves maestras), como se muestra en la Figura 5.1. Entre las técnicas y ejemplos no criptográficos podemos citar seguridad física y confianza.

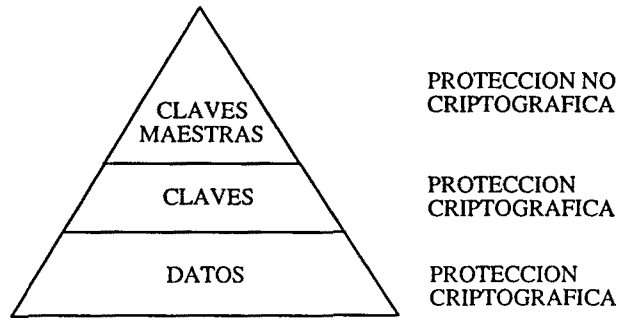

Figura 5.1. Estructura jerárquica de claves

La distribución de claves se lleva siempre a cabo mediante protocolos, es decir, secuencias de pasos de comunicación (transferencia de mensajes) y pasos de computación. Muchas de las propiedades de estos protocolos dependen de la estructura de los mensajes intercambiados y no de los algoritmos criptográficos subyacentes. Por ello las debilidades de estos protocolos provienen normalmente de errores cometidos en los niveles más altos del diseño, en lugar de ser debidos a los operadores criptográficos empleados.

Las claves criptográficas deben ser generadas de forma aleatoria. Para protegerlas será necesaria seguridad física o cifrado, mientras que para evitar que sean modificadas deberá utilizarse seguridad física o autentificación. La autentificación hace uso de parámetros como *time-stamps* y contadores para protegerse también contra la reactuación con antiguas claves.

Los requerimientos genéricos de seguridad para los mensajes intercambiados entre entidades incluyen los siguientes:

- *Confidencialidad de datos:* Las claves secretas y posiblemente otros datos son mantenidos secretos mientras son transmitidos y almacenados
- *Detección de modificación:* En la mayoría de entornos, todos los mensajes deben ser protegidos contra modificación
- *Detección de reactuación:* Para detectar duplicaciones de mensaje no autorizadas

- *Autentificación de entidad:* Para corroborar que una entidad es la que pretende ser
- *Autentificación de origen de datos:* Para corroborar que la fuente de un mensaje es quien pretende ser
- *Prueba de recepción:* Para probar al emisor que el mensaje ha sido recibido por su legítimo receptor
- *Notarización*

En los protocolos de gestión de claves, cada entidad basa sus deducciones no sólo en el protocolo, sino también en la confianza que se tenga en un servidor, a menudo llamado Entidad de Confianza. Sin embargo no existe una especificación formal de lo que se entiende por confianza, y el grado de confianza necesario depende en cierta medida del tipo de servidor. En cualquier caso debe existir cierta "institución" detrás del sistema, que puede ser por ejemplo el propietario, o la autoridad responsable, o el vendedor. La responsabilidad de esta institución debe incluir:

- Refuerzo de la política de seguridad del sistema
- Identificación de entidades registradas
- Garantizar la integridad del sistema

El número de claves a manejar en un sistema puede ser muy grande. Esto hace necesario dividirlas en dominios de responsabilidad de gestión, donde el término dominio describe el área o ámbito del control de gestión. Los dominios de gestión y su interrelación definen la política de seguridad del sistema. Estos dominios pueden ser disjuntos, solapados, o bien subconjuntos de otros dominios.

El concepto de usar claves para proteger otras claves reduce el problema de proteger físicamente un gran número de claves a otro problema en que sólo es necesario proteger un conjunto más pequeño. Además, la introducción de una Jerarquía de Claves reduce significativamente el número de claves que no pueden ser distribuidas automáticamente. Las claves deberían igualmente tener un tiempo de vida limitado basado en tiempo, uso, o cualquier otro criterio.

En cualquier sistema de seguridad, existe un núcleo donde debe implantarse seguridad física. Estos datos protegidos físicamente son normalmente las llamadas claves maestras, que están en el nivel más alto de la jerarquía de claves y por tanto no pueden ser protegidas con otras claves. Dado que la protección física es costosa, debe minimizarse su necesidad usando técnicas más baratas cuando sea posible.

Los criterios más importantes de diseño para un sistema de gestión de claves son los siguientes:

- Minimizar el número y complejidad de los mecanismos basados en confianza involucrados, minimizando especialmente el uso de mecanismos centralizados
- Minimizar la actividad física, es decir, el número de correos debería ser mínimo (inexistente, si es posible). Esto implica que las entidades no deberían viajar lejos para registrarse (para grandes sistemas, esto sugiere una organización jerárquica)
- Minimizar la necesidad de seguridad física, es decir, el número de dispositivos tampón y de canales seguros necesarios
- Alcanzar la máxima flexibilidad
- Alcanzar la máxima robustez (por ejemplo, autosincronización cuando las claves son actualizadas)
- Asegurar que si una entidad es deshonesta, podrá ser descubierta

5.2.2 Autentificación

La autentificación juega un papel fundamental en los protocolos de gestión de claves, puesto que es preciso verificar la autenticidad de las entidades que negocian las claves. Existen tres tipos principales de autentificación: la identificación de una entidad, la autentificación de contenido de un mensaje y la autentificación de su origen. Un proceso complementario es el de verificación o chequeo por parte del receptor del mensaje de las entidades o datos que han sido autentificados.

Aparte de su importancia en los mecanismos de gestión de claves, la autentificación juega un papel fundamental por sí sola en los entornos abiertos, en los que un proceso controlado por un usuario (cliente) debe conectarse remotamente a un servidor. El método de autentificación más ampliamente usado en las redes actuales es la utilización de contraseñas (*passwords*) para llevar a cabo la identificación de usuarios. Sin embargo, este sistema presenta ciertas debilidades que lo hacen desaconsejable en muchas aplicaciones.

En primer lugar los *passwords* son transmitidos en claro cuando viajan por la red. En sistemas UNIX, por ejemplo, la identificación de usuarios se lleva a cabo comparando el contenido del fichero /etc/passwd (donde reside el *password* cifrado de cada usuario) con la contraseña que el usuario introduce por teclado, posteriormente a su cifrado. Cuando un usuario pretende demostrar su identidad ante un servidor remoto (operación muy frecuente si el sistema está conectado a una red), el *password*

introducido en el teclado local viaja en claro por la red hasta llegar al servidor, donde es cifrado y se verifica la identidad. Durante este proceso, es extraordinariamente sencillo escuchar la red y capturar el *password* mediante analizadores de protocolos hardware/software.

Por otra parte, los *passwords* son generalmente fáciles de adivinar, puesto que los usuarios los eligen de tal forma que les sea sencillo recordarlos. Una gran cantidad de *passwords* son nombres propios o palabras que aparecen en el diccionario, por lo que es posible entrar en un sistema simplemente probando *passwords*.

Por último conviene destacar que la autentificación se da únicamente en un sentido, es decir, es el usuario el que se identifica ante el servidor, pero no viceversa. Ello hace posible la utilización de ciertos servidores fraudulentos que permitan obtener cierta información secreta de los usuarios (por ejemplo, su contraseña).

Por todo ello, resulta conveniente la utilización de otras técnicas para verificar la identidad de usuarios ante sistemas, como el reconocimiento de información biométrica (muestras de voz, huellas digitales, firmas manuales, etc.). Sin embargo, estas técnicas presentan problemas de fiabilidad a la vez que requieren un soporte *hardware* costoso, por lo que la solución más adecuada es la utilización de técnicas criptográficas para mecanismos de autentificación. Mediante ellas, el usuario o entidad a autentificar prueba su identidad demostrando su capacidad para cifrar o descifrar ciertos datos con una clave secreta o privada. Para evitar ataques por reactuación en los que un mensaje legítimo es reutilizado con fines deshonestos, es preciso que los datos a cifrar o descifrar (denominados retos) varíen para cada ejecución del protocolo. Existen 3 técnicas para formar estos retos: el uso de marcas temporales (*time-stamps*), el uso de contadores y el uso de números aleatorios estadísticamente irrepetibles (*nonces*).

La técnica basada en *time-stamps* consiste en que la entidad a ser autentificada (A) cifra el contenido de su reloj y lo envía a la entidad que pide autentificación (B). Seguidamente B descifra el mensaje y lo compara con el contenido de su reloj. En este proceso ambas entidades deben tener relojes sincronizados en tiempo real, para que la verificación sea posible. En la práctica el contenido de ambos relojes deberá coincidir dentro de una ventana de tiempo necesaria debido a errores de sincronización y al tiempo de transmisión del mensaje a través de la red. Sin embargo, el enemigo siempre podrá aprovechar la tolerancia de la ventana de tiempo para suplantar una entidad mediante un ataque por reactuación en el que capture y reenvíe la marca temporal cifrada.

En la técnica basada en contadores ambas entidades deben poseer contadores sincronizados, siendo el reto irrepetible el contenido de este contador. Los contadores

deben ser lo suficientemente largos como para evitar un ataque basado en la espera determinista a la repetición de ese contador. Sin embargo, el mantener sincronizados contadores de gran longitud en máquinas diferentes no es un problema obvio, siendo especialmente conflictivo el caso en que ambas partes comiencen el proceso de autentificación a la vez.

Debido a los problemas de las técnicas anteriores, muchas veces es aconsejable el usar como retos números generados aleatoriamente (*nonces*) que evitan problemas de sincronización. Sin embargo debe tenerse el cuenta que el precio a pagar es un mensaje extra en la red, solución aceptable en la mayoría de los casos.

5.2.3 Aspectos Concretos de Gestión de Claves en la RDSI-BA

Los aspectos de gestión anteriormente tratados son generales, debiéndose considerar en cualquier entorno. Sin embargo, cada red particular tiene sus propias características que determinarán el sistema de gestión de claves finalmente empleado.

En este apartado se considera el caso particular de la red digital de servicios integrados de banda ancha (RDSI-BA). Como aspectos significativos cabe resaltar que se trata de una red pública de área extendida (*WAN*), de alta velocidad y orientada a conexión.

Como red de alta velocidad implica la utilización de cripotosistemas simétricos (en bloque o en flujo) para el cifrado del "grueso" de la información (en inglés, *bulk encryption*), ya que la tecnología actual está muy lejos de conseguir realizaciones de criptosistemas de clave pública a estas velocidades.

Como red orientada a conexión resulta natural la asociación de claves de sesión con conexiones. De esta forma la clave debe negociarse durante el periodo de establecimiento de la conexión o al inicio de ésta[2], y debe tener un período de validez igual a la duración de la conexión. Una vez finalice la conexión, esta clave ya no podrá ser reutilizada. La negociación de la clave de sesión previamente a la comunicación segura hace que la velocidad de cifrado no sea un requisito fundamental para la gestión de claves, lo que permite la utilización de criptosistemas de clave pública.

[2] En el primer caso los mecanismos de establecimiento de conexión (la señalización) establecerán directamente una conexión segura, mientras que en el segundo caso se establecerá una conexión insegura que posteriormente se transformará en una segura. En el capítulo 3 se presenta una arquitectura de seguridad que contempla este segundo caso.

Como red pública de área extendida cabe esperar la presencia de un gran número de entidades pertenecientes a diferentes dominios e instituciones con gran dispersión geográfica. Ello prácticamente obliga a la utilización de criptografía de clave pública para la gestión de claves, a la vez que es necesaria una organización jerárquica de autoridades de confianza (centros de certificación).

El uso de centros de certificación es necesario en este entorno, ya que no es posible que cada entidad almacene de forma auténtica las claves públicas de todas las demás, debido principalmente a dos razones. La primera de ellas es que se trata de una red de ámbito mundial, con lo que la cantidad de información a almacenar sería muy grande. La segunda, y sin duda la más importante, es que de esta forma cada vez que una entidad se dé de alta o de baja en la red deberían actualizarse las agendas de claves públicas de todas las otras entidades, lo que es claramente impensable en una red como la RDSI-BA.

La criptografía de clave pública aporta también ventajas adicionales en una red multidominio orientada a conexión, donde desde un punto de vista económico es interesante minimizar en primer lugar el número de conexiones[3]. Mediante criptografía de clave pública y el uso de certificados, es posible la negociación de la clave de sesión utilizando una única conexión entre ambas entidades involucradas (que puede ser la misma conexión por la que se realice la posterior comunicación). Utilizando criptografía convencional se hace necesario el acceso a centros de distribución de claves, con lo que el número de conexiones es mayor, especialmente en el caso que ambas entidades pertenezcan a dominios diferentes[4].

Además del número de conexiones, la criptografía de clave pública permite reducir también el número de mensajes intercambiados para distribuir una clave de sesión. Mientras [YAH93] demuestra que para un sistema jerárquico de centros de distribución de claves de n niveles son necesarios y suficientes n+4 mensajes con criptosistemas simétricos, con criptosistemas de clave pública únicamente son necesarios y suficientes 3 mensajes[5].

---

[3] Ello es debido a que el establecimiento de una nueva conexión supone una carga de señalización importante.

[4] En este caso se hace necesario acceder a centros de distribución de confianza para ambas entidades.

[5] En ambos casos se ha considerado la utilización de retos y la clave de sesión se considera generada por el centro de distribución del dominio de la entidad destino en el caso de criptografía de clave simétrica y por una de las entidades participantes en el caso de criptografía de clave pública.

5.3 Protocolo Propuesto para la RDSI-BA

En este apartado se propone un protocolo de gestión de claves para la RDSI-BA, que es generalizable a otras redes digitales con gran número de usuarios que soporten servicios orientados a conexión, como es el caso de Internet[6] o de la Red Digital de Servicios Integrados de Banda Estrecha (RDSI-BE). Como criterios de optimización minimizaremos:

- El número de conexiones (se accederá lo mínimo posible a centros de confianza)
- El número de mensajes intercambiados
- La longitud de estos mensajes
- El número y complejidad de operaciones criptográficas involucradas

El protocolo propuesto fue adelantado en [FOR94b], y pretende minimizar los criterios de optimización anteriormente señalados. Se utiliza criptografía de clave pública por ser la mejor opción cuando el número de entidades es alto y por requerir una menor confianza [FUM93]. En este protocolo cada entidad guarda una tabla autentificada de las claves públicas de las L entidades con las cuales es más probable una conexión segura. Para negociar una clave se consulta esta tabla[7]. Si la entidad con la que se desea establecer la comunicación está registrada, se inicia un sencillo protocolo autentificado de 3 mensajes, el cual no utiliza ningún certificado.

5.3.1 Notación

A fin de describir de manera compacta y precisa el protocolo de gestión de claves es necesario tener en cuenta el significado de las siguientes expresiones:

S_A	Clave secreta de *A*
P_A	Clave pública de *A*
N_A	*Nonce* a enviar de *A* a *B* (actúa como reto, y es un número aleatorio que no se repite).

[6] Sí bien el protocolo IP no es orientado a conexión, muchos servicios de Internet utilizan el protocolo TCP, que sí lo es.

[7] Por simplicidad se considera que, o bien ambas entidades tienen registrada la clave de la otra parte, o bien ninguna de ellas la tiene. Los casos mixtos requerirían alguna modificación.

S_{AB}	Clave de sesión compartida por A y B. Obviamente $S_{AB} = S_{BA}$.
T_{exp}	Instante de expiración de un certificado.
A	Identificador de A . Sólo A está autorizado a utilizarlo.
$CERT\{X\}_K$	X ha sido certificado por un Centro de Certificación. Usualmente sólo se certifica, con la clave K, un "resumen", o *message digest*, de X. La clave K es normalmente la clave secreta del centro de certificación.
$A \Rightarrow B \quad \{X_1,..,X_N\}$	A envía un mensaje a B compuesto por la concatenación.

Donde X puede ser cualquiera de las expresiones especificadas, o un conjunto de las mismas.

5.3.2 Protocolo Propuesto

El sistema de gestión de claves que se propone cubre una serie de amenazas contrarrestadas por los siguientes servicios de seguridad:

- *Confidencialidad de claves:* se garantiza la confidencialidad de las claves de sesión negociadas
- *Integridad:* se detectan modificaciones no autorizadas durante la transmisión
- *Detección de reactuación (replay):* se detectan las reproducciones no autorizadas de parte o la totalidad de la transmisión
- *Autentificación de origen:* se garantiza que el origen de una transmisión es quién pretende serlo

El protocolo que se presenta está basado en modificaciones del protocolo de autentificación de Needham-Schroeder [NEE78]. Estas modificaciones se realizan con el propósito de posibilitar la negociación de claves y de cumplir los criterios de optimización previamente enunciados.

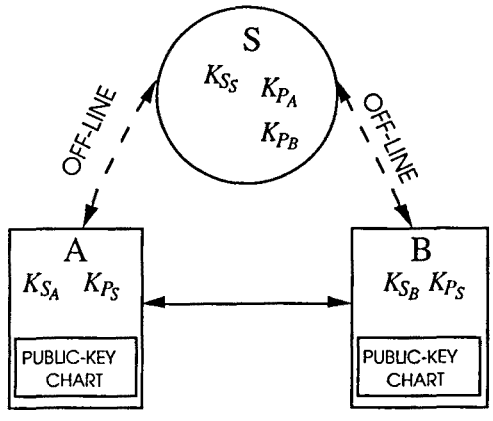

Figura 5.2. Escenario del protocolo propuesto.

La figura 5.2 muestra el escenario para el que se ha desarrollado el protocolo. En este escenario cada terminal tiene una tabla, o agenda, de claves públicas de otros terminales. En principio, esta agenda debería contener las claves públicas de los terminales a los que es más probable conectarse. Evidentemente la construcción de estas tablas no es un problema trivial.

En este contexto, cuando *A* necesita negociar una clave de sesión con *B*, mira primero si la clave de *B* figura en su agenda. Si es así, se lleva a cabo el Protocolo 1, en otro caso se ejecuta el Protocolo 2[8].

El Protocolo_1 sirve para llevar a cabo una negociación de claves y una autentificación entre dos terminales:

Protocolo_1 : Mensaje_1 : A⇒B $\{A,N_A\}_{P_B}$
Mensaje_2 : B⇒A $\{N_A,N_B\}_{P_A}$
Mensaje_3 : A⇒B $\{N_B\}_{P_B}$

Cuando *A* interpreta el *Mensaje_2* puede estar seguro de que N_B proviene de *B* y que *B* conoce N_A. Cuando *B* recibe el *Mensaje_3* puede estar seguro que tanto este como el *Mensaje_1* provienen de *A*, pues en caso contrario sería imposible la ejecución completa del protocolo. El *Mensaje_3* sirve para realizar la autentificación en el sentido contrario y de reconocimiento del *Mensaje_2*. Las reactuaciones (*replays*) son detectadas gracias a que los *nonces* no se repiten nunca y a que sólo las partes autorizadas son capaces de leerlos en claro y así construir mensajes correctos. La

[8]En aras de simplificar la exposición supondremos que sólo se puede dar una de las dos siguientes situaciones: a) Tanto *A* como *B* conocen sus respectivas claves públicas; b) Ni *A* ni *B* conocen sus respectivas claves públicas. Para el caso general consultése [REB96].

integridad de los mensajes se consigue mediante la inclusión de los identificadores de *A* y *B* en los mensajes. Una vez ejecutado este protocolo ambos terminales utilizan la siguiente clave de sesión:

$$S_{AB} = N_A \oplus N_B$$

Observar que ambos terminales participan equitativamente en la generación de la clave. Además, gracias a las propiedades del operador *X-OR* cualquiera de las partes puede estar segura de la buena aleatoriedad (utilizando un buen generador de aleatorios) independientemente de la calidad del generador de la otra parte.

En el Protocolo_1 se transfieren un total de tres mensajes para negociar una clave de sesión. Este número es el mínimo necesario para un protocolo fiable que lleve a cabo la negociación de claves: un mensaje para iniciar la comunicación y dos de reconocimiento, uno por cada parte, son el mínimo posible.

Este protocolo es adecuado para escenarios con pocos terminales donde cada uno de ellos puede guardar en su "agenda" las claves públicas de todos los demás. Sin embargo, en un entorno en el que el número de terminales a considerar pueda ser elevado, o que sea muy frecuente la instalación de nuevos terminales, es inviable que todos conozcan las claves públicas de todos. El Protocolo_2 solventa estos inconvenientes.

Protocolo_2 :

$$\begin{aligned}
&\text{Mensaje_1} : A \Rightarrow B \quad \{CERT\{S, A, P_A, T_{exp}\}_{S_S}\} \\
&\text{Mensaje_2} : B \Rightarrow A \quad \{\{B, N_B\}_{P_A}, CERT\{S, B, P_B, T_{exp}\}_{S_S}\} \\
&\text{Mensaje_3} : A \Rightarrow B \quad \{N_B, N_A\}_{P_B} \\
&\text{Mensaje_4} : B \Rightarrow A \quad \{N_A\}_{P_A}
\end{aligned}$$

En el Protocolo_2 se asume que existe una distribución previa *off-line* de Certificados. Esta distribución la lleva a cabo *S* (ver fig. 2), que es un Centro de Certificación. *S* conoce las claves públicas de todos los usuarios, emite Certificados de claves públicas con un instante de expiración asociado T_{exp} y los renueva una vez han expirado. Cada terminal debe almacenar su Certificado para poder llevar a cabo el Protocolo 2 cuando deba negociar una clave de sesión.

En el *Mensaje_1* *A* envía su clave pública certificada a *B*, quien a su vez le responde con un mensaje compuesto por su clave pública certificada y la misma información que en el *Mensaje_1* del Protocolo_1. A continuación, una vez los certificados son comprobados el protocolo continúa de forma análoga al Protocolo_1

(con los papeles de *A* y *B* intercambiados). En comparación con el primer protocolo, este requiere de un mensaje más y las longitudes del primer y segundo mensajes son mayores. No obstante se observa que el número de computaciones "costosas" con claves privadas es el mismo (suponiendo que el criptosistema utilizado minimiza la complejidad del proceso de verificación de firmas).

Dado que en un entorno real de red de área extendida el número de terminales es alto no es posible que todos conozcan las claves públicas de todos. De forma que la utilización del Protocolo_2 es inevitable. El punto clave será pues el diseño del algoritmo de mantenimiento de las "agendas" de claves públicas, así como el dimensionado de su capacidad.

Es interesante observar que el Centro de Certificación puede convertirse en un "cuello de botella" del sistema si las dimensiones de la red son muy grandes. Un entorno como este deberá ser dividido en dominios administrados mediante una red jerárquica de Centros de Certificación como recomienda CCITT X.509 [CCI89].

5.3.3 Dimensionado de Tabla de Claves Públicas

Si la clave pública que necesita una entidad se encuentra registrada en su tabla, el coste que representa el obtener el certificado de dicha clave se elimina. Es obvio que cuanto mayor sea la longitud L de la tabla, menor será la probabilidad P de que la entidad con la que se desea establecer la comunicación no esté registrada. Sin embargo, un incremento de L es costoso, debido a la dificultad de mantener una gran tabla autentificada y al coste de búsqueda y actualización asociado.

En este apartado se estudia el dimensionado óptimo de la tabla de claves públicas que minimiza el coste total de la gestión de claves. Se restringe el estudio al de una red con un único centro de certificación, en la que se utiliza el protocolo de gestión de claves descrito.

Sea N el número total de entidades en la red. Supongamos que una entidad se comunica con otra entidad U de entre las N-1 restantes de forma que podemos modelar U como una variable aleatoria discreta cuyo comportamiento aproximaremos al de una continua exponencial de parámetro ν, con un factor de normalización:

$$P\{U=u\}=\frac{1}{1-e^{-\nu(N-1)}}\int_{u-1}^{u}\nu e^{-\nu z}dz \qquad \forall u\in\{1,...,N-1\} \tag{5.1}$$

$$\text{donde, si } e^{-\nu(N-1)} << 1 \quad (\text{N es grande}) \qquad \text{entonces} \quad \frac{1}{1-e^{-\nu(N-1)}} \cong 1$$

Si la tabla contiene registradas las L entidades con las que es más probable la comunicación, obtenemos P (probabilidad de no estar en la tabla) como :

$$P = P\{U \notin \{1,...,L\}\} \cong 1 - \int_0^L \nu e^{-\nu z} dz = e^{-\nu L} \tag{5.2}$$

Usando la notación siguiente :

- CU : indica coste para la entidad CR : indica coste para la red
- C : indica coste para ambos C=CU+CR
- C_{Prot1}, C_{Prot2} : coste asociado al uso de los protocolos 1 y 2, respectivamente
- CU_M : coste en memoria o hardware de la tabla para la entidad
- CU_{TC} : coste en tiempo de cálculo para la gestión de la tabla

Sea la siguiente expresión para CU_{TC} :

$$CU_{TC} = \alpha \log_2 L + (1-P)\beta_1 + P(\beta_2 + \gamma L) \quad \beta_2 > \beta_1 \quad \gamma << \alpha, \beta_1, \beta_2 \quad (L \geq 1) \tag{5.3}$$

donde el término logarítmico representa el coste de la búsqueda binaria (ver [WIR86]) ; los valores β_1 y β_2, costes constantes ; el término lineal representa el coste de reasignación de punteros para mantener una ordenación en la tabla cuando ésta debe actualizarse incluyendo una nueva entidad, ordenación necesaria para permitir la búsqueda binaria, y 1-P y P dan a CU_{TC} el carácter de promedio entre las situaciones de estar o no, respectivamente, una entidad en la tabla. Para CU_M tomemos una dependencia lineal con L según :

$$CU_M = a\,L + b \tag{5.4}$$

El coste total vendrá dado por :

$$C_{total} = CU_{total} + CR_{total} = CU_M + CU_{TC} + (1-P)\,C_{Prot1} + P\,C_{Prot2} =$$

$$= \alpha \log_2 L + (A + \gamma L)\, e^{-\nu L} + aL + B \tag{5.5}$$

$$\text{con} \quad A \equiv (\beta_2 - \beta_1 + C_{Prot2} - C_{Prot1}) \quad \text{y} \quad B \equiv b + \beta_1 + CU_{Prot1}$$

Si en la diferencia de costes entre tener y no tener una entidad registrada, el coste de modificar un puntero, representado por γ, es despreciable de tal forma que $\gamma L << A$, entonces puede aproximarse el coste total. Si además se considera L real en lugar de natural, derivando la expresión aproximada para obtener el valor L que minimiza el coste, resulta :

$$\frac{\alpha}{\ln 2}\frac{1}{L} + a = \nu A e^{-\nu L} \tag{5.6}$$

Puede comprobarse que $a<\nu A$ es condición necesaria para que exista solución (coste *hardware* limitado), y que en general se obtienen dos soluciones para L de las cuales la mayor es un mínimo, es la solución que nos interesa y tiene sentido, y la menor es un máximo. No encontrar solución indica que no conviene usar una tabla.

Puede despejarse L si se cumple la aproximación mostrada :

$$\frac{\alpha}{\ln 2}\frac{1}{L} << a \quad \Rightarrow \quad L \cong \frac{1}{\nu}\ln\frac{\nu A}{a} \tag{5.7}$$

o bien, si no se cumple la aproximación, la L obtenida en (5.7) es una cota superior de la L buscada.

Las expresiones (5.6) y (5.7) permiten hallar el tamaño óptimo L de la tabla que minimiza el coste total para la entidad y para la red, en función de diversos parámetros relacionados con el coste *hardware* de la tabla, la búsqueda en ella y su actualización, los costes para la red y para la entidad según el uso de un protocolo u otro y el comportamiento de las entidades respecto a la elección de la entidad con la que establecen comunicación.

5.3.4 Desarrollo del Protocolo Propuesto para el Proyecto CRIPTO

Para el proyecto CRIPTO del PlanBA (ver apartado 2.6 de esta tesis) se ha desarrollado una versión simplificada del protocolo propuesto para sistemas UNIX. En esta versión no se consideraron criterios de actualización de las tablas de claves públicas registradas ni su dimensionado óptimo.

Los requerimientos específicos para este protocolo son utilizar como operador de clave pública RSA-512 bits, y negociar claves de sesión de 247 bits, que serán utilizadas por un cifrador en flujo.

El software de gestión de claves se utiliza el protocolo de comunicaciones TCP/IP, ya que se considera disponible en todas las estaciones de trabajo que formen parte de los terminales multimedia. Así se han desarrollado aplicaciones en C sobre sistema operativo UNIX de Sun que se comunican mediante TCP/IP. Como identificador de entidad se ha considerado conveniente utilizar la dirección IP de cada terminal (4 bytes), mientras que los retos se generan mediante una función seudoaleatória disponible en UNIX que se inicializa con una semilla función de la fecha y hora[9]. La longitud de cada uno de estos retos (32 bytes) es lo suficientemente grande como para poder generar a partir de ellos la clave de sesión que deberá pasarse al cifrador en flujo (247 bits), a la vez que permite cifrar 2 retos consecutivamente en un mismo bloque RSA (512 bits).

En diciembre de 1995 se realizó una demostración de este software. En ella participaron 3 máquinas (estaciones de trabajo Sun). La primera de ellas jugó el papel de centro de certificación y distribución de claves públicas, mediante un proceso en "background" denominado *democent*. La segunda jugó el papel de terminal B de los protocolos. Sobre ella se ejecutó un proceso servidor en "background" denominado *demoserv*, que estaba a la espera de ser despertado por un proceso cliente de gestión de claves. Este proceso cliente, denominado *democli*, se ejecutó sobre una tercera estación de trabajo que representaba el terminal A.

[CEÑ96] presenta una descripción detallada del software desarrollado para este proyecto.

5.4 Comparación con otros Protocolos

En este apartado se compara el protocolo de gestión de claves propuesto en el apartado anterior con otros protocolos conocidos, en concreto con el estándar X.509 [CCI89] y con el protocolo STS (*Station to Station*) propuesto por Diffie, van Oorschot y Wiener [DIF92].

[9] Este es un claro ejemplo de la utilización de una función de generación de retos sencilla, pero fácilmente previsible por un atacante que conozca el proceso de generación. Sirve para mostrar la importancia de la generación de la clave de sesión por ambas entidades, lo que hace este tipo de ataques mucho más complicados.

5.4.1 X.509

La recomendación del CCITT X.509 (ISO/IEC 9594-8 ó ITU-T Recommendation X.509) forma parte de la serie de recomendaciones X.500 [CCI92j] que definen el directorio, que consiste en un conjunto distribuido de servidores que mantienen una base de datos con información de los usuarios. X.509 es un ejemplo conocido de mecanismos de autentificación basados en clave pública. El protocolo, utilizando la notación del apartado 5.3.1, es el siguiente:

Protocolo X.509

Mensaje_1 : A⇒B $A,\left\{T_A,N_A,B,\text{sgnData},\{S_{AB}\}_{P_B}\right\}_{S_A}$

Mensaje_2 : B⇒A $\left\{T_B,N_B,A,N_A,\text{sgnData},\{S_{BA}\}_{P_A}\right\}_{S_B}$

Mensaje_3 : A⇒B $\{B,N_B\}_{S_A}$ (opcional)

Este protocolo supone que ambos usuarios (A y B) conocen la clave pública del otro, bien sea porque la tienen registrada o porque la han solicitado previamente a la autoridad de certificación (ver capítulo 4). Por ello deberá compararse con el denominado Protocolo 1, que recordemos era el siguiente:

Protocolo_1 : Mensaje_1 : A⇒B $\{A,N_A\}_{P_B}$

Mensaje_2 : B⇒A $\{N_A,N_B\}_{P_A}$

Mensaje_3 : A⇒B $\{N_B\}_{P_B}$

$S_{AB} = N_A \oplus N_B$

El protocolo X.509 permite la utilización de *time-stamps* o de *nonces* para comprobar la "frescura" (en inglés, *freshness*) de los mensajes. El mensaje 3 es opcional, y sólo es necesario cuando se desea utilizar la técnica de los *nonces*. En este caso el número de mensajes del protocolo X.509 será de tres, al igual que en el Protocolo_1. Este número de mensajes es el mínimo posible para una autentificación mutua utilizando esta técnica[10], por lo que ambos protocolos son óptimos en cuanto al número de mensajes.

[10] En efecto, como mínimo será necesario un primer mensaje para lanzar el primer reto, un segundo mensaje para contestar al reto y proponer un nuevo reto y un tercer mensaje para contestar a este tercer reto.

Sin embargo, el protocolo 1 utiliza mensajes de longitud más reducida y el número de operaciones criptográficas involucradas es menor. Ello es en parte consecuencia de aprovechar los *nonces* para generar la clave de sesión.

5.4.2 Protocolo STS

Diffie, van Oorschot y Wiener propusieron en 1992 un protocolo de autentificación y gestión de claves, que denominaron STS [DIF92]. Combina autentificación mutua basada en clave pública con derivación de clave pública según el método de Diffie-Hellman [DIF76]. El protocolo es el siguiente:

Protocolo_STS :

Mensaje_1 : A⇒B $\left\{a^x\right\}$

Mensaje_2 : B⇒A $\left\{a^y, E_K\left\{\left\{a^y, a^x\right\}_{S_B}\right\}\right\}$

Mensaje_3 : A⇒B $E_K\left\{\left\{a^y, a^x\right\}_{S_A}\right\}$

con:

a, p	Parámetros de Diffie-Hellman
a^x	Exponencial de Diffie-Hellman, $a^x \bmod p$ con x secreto conocido sólo por A
a^y	Exponencial de Diffie-Hellman, $a^y \bmod p$ con y secreto conocido sólo por B
$E_k\{\ \}$	Cifrado con la clave $k = a^{x \cdot y} \bmod p$

Este protocolo presupone que ambos usuarios conocen la clave pública del otro. Por lo tanto, comparado con el Protocolo_1 utiliza el mismo número de mensajes (tres), parecido número de operaciones criptográficas, pero está limitado al método de negociación de Diffie-Hellman, mientras que el Protocolo_1 puede utilizar cualquier criptosistema de clave pública.

Si ambos usuarios no conocen la clave pública del otro, es posible la siguiente modificación del protocolo STS con funciones parecidas al Protocolo_2:

Protocolo_STS modificado :

Mensaje_1 : A⇒B $\left\{a^x, \text{CERT}\{S, A, P_A, T_{exp}\}_{S_S}\right\}$,

Mensaje_2 : B⇒A $\left\{a^y, E_K\left\{\left\{a^y, a^x\right\}_{S_B}\right\}, \text{CERT}\{S, B, P_B, T_{exp}\}_{S_S}\right\}$

Mensaje_3 : A⇒B $E_K\left\{\left\{a^y, a^x\right\}_{S_A}\right\}$

Comparado con el Protocolo_2, el protocolo STS modificado utiliza un mensaje menos y el número de operaciones criptográficas es similar. A cambio, está restringido al método de derivación de claves de Diffie-Hellman y a su seguridad, mientras que el Protocolo_2 es más general y puede utilizar cualquier operador criptográfico de clave pública.

5.5 Autentificación de Parámetros de Seguridad Negociados

En la arquitectura de seguridad propuesta en el capítulo 3 se da una negociación de parámetros de seguridad entre los niveles SEC-APL de los terminales multimedia involucrados en una comunicación segura sobre la RDSI-BA. Un factor clave para la seguridad del sistema es garantizar la autenticidad e integridad de los parámetros negociados. Ello puede lograrse añadiendo un campo con la configuración de los parámetros negociados (*Param_Seg*) en el protocolo de gestión de claves. Así, el protocolo propuesto quedaría de esta forma:

Protocolo_1 :
Mensaje_1 : A⇒B $\{A, N_A, \text{Param\_Seg}\}_{P_B}$
Mensaje_2 : B⇒A $\{N_A, N_B, \text{Param\_Seg}\}_{P_A}$
Mensaje_3 : A⇒B $\{N_B\}_{P_B}$

Prot_2 :
Mensaje_1 : A⇒B $\{\text{CERT}\{S, A, P_A, T_{exp}\}_{S_S}\}$
Mensaje_2 : B⇒A $\{\{B, N_B, \text{Param\_Seg}\}_{P_A}, \text{CERT}\{S, B, P_B, T_{exp}\}_{S_S}\}$
Mensaje_3 : A⇒B $\{N_B, N_A, \text{Param\_Seg}\}_{P_B}$
Mensaje_4 : B⇒A $\{N_A\}_{P_A}$

Donde *Param_Seg* puede ser la salida de una función de *hash* de la configuración de los parámetros negociados.

Otro protocolo que puede adaptarse igualmente a la negociación de parámetros de seguridad de forma muy sencilla es el X.509, simplemente introduciendo *Param_Seg* en el campo sgnData, que proporciona firma digital.

Protocolo X.509 utilizado para autentificar parámetros de seguridad

Mensaje_1 : A⇒B $\left\{A, \left\{T_A, N_A, B, \text{Param\_Seg}, \{S_{AB}\}_{P_B}\right\}_{S_A}\right\}$

Mensaje_2 : B⇒A $\left\{T_B, N_B, A, N_A, \text{Param\_Seg}, \{S_{BA}\}_{P_A}\right\}_{S_B}$

Mensaje_3 : A⇒B $\{B, N_B\}_{S_A}$ (opcional)

5.6 Conclusiones y Aportaciones

En este capítulo se ha propuesto un protocolo a utilizar para la negociación de claves de sesión entre entidades conectadas a la RDSI-BA. Se ha llegado a la conclusión que en este entorno (red multidominio con un gran número de usuarios conectados) la criptografía de clave pública es esencial para la gestión de claves.

Se ha partido de un protocolo de autentificación conocido considerado seguro por la comunidad internacional como el de Needham-Schroeder [NEE78] y se han realizado modificaciones para permitir la gestión de claves y aumentar su eficiencia, sin comprometer su seguridad. En este protocolo ambas entidades (típicamente terminales multimedia) participan en la generación de la clave de sesión, lo que hace innecesario confiar en la bondad del proceso de generación de claves de la otra entidad.

En este entorno no es posible que cada entidad guarde las claves públicas del resto, lo que implica el acceso a autoridades de certificación. Para minimizar el número de conexiones, y consecuentemente el tráfico de gestión de seguridad, se propone que las entidades mantengan certificados con sus claves públicas y los intercambien para la negociación de claves de sesión, reduciéndose de esta forma los accesos al centro de certificación. Además, para reducir el número de mensajes transferidos y la longitud de éstos se propone que las entidades mantengan una tabla con las claves públicas de las entidades con las que es más probable una comunicación segura. Este protocolo fue adelantado en [FOR94b] y se utilizó para la gestión de claves en el proyecto CRIPTO del PlanBA, tal como se señala en [FOR95a].

Se ha abordado el dimensionado óptimo de esta tabla de claves públicas. Para ello se ha modelado la probabilidad que tiene una entidad de conectarse con las restantes de la red según una exponencial decreciente. Ello significa que existirán una serie de usuarios con los que será más probable establecer comunicaciones seguras mientras que con otros lo será mucho menos. A partir de aquí, y tomando como objetivo el minimizar el coste total de la gestión de claves por usuario, se ha hallado una longitud óptima de la tabla de claves públicas. Esta parte está muy relacionada con el estudio de costes del capítulo 4 de esta tesis, y fue adelantada en [FOR96b].

El protocolo propuesto se ha comparado con dos protocolos de características similares y de gran difusión, como son el X.509 [CCI89] y el STS [DIF92]. Respecto al primero utiliza mensajes de menor longitud y un menor número de operaciones criptográficas, siendo en este sentido más eficiente. Respecto al segundo tiene la ventaja de no estar supeditado a ningún sistema de clave pública concreto, pudiendo utilizar cualquiera.

Por último se ha presentado una versión modificada del protocolo propuesto que permite garantizar la autenticidad e integridad de parámetros de seguridad negociados previamente, lo que permite utilizarlo como protocolo de gestión de claves de la arquitectura de seguridad presentada en el capítulo 3 de esta tesis.

CAPÍTULO 6

Conclusiones y Líneas Futuras

6.1 Conclusiones

La red digital de servicios integrados de banda ancha (RDSI-BA) se espera que sea la red pública de comunicaciones del futuro próximo, permitiendo el transporte integrado de datos, voz y vídeo.

Como en todas las redes existentes, en la RDSI-BA son posibles una serie de ataques a la seguridad de las comunicaciones, entre los que destacan acceso no autorizado a recursos, modificación de mensajes o escuchas no autorizadas. En este sentido [LAU96] presenta un estudio de las amenazas existentes en la arquitectura de emulación de red local MTA (ELAN, *Emulated LAN*), algunas de las cuales se dan igualmente en la RDSI-BA. Para contrarrestar estas amenazas es necesario ofrecer servicios de seguridad, que según ISO/IEC 7498-2 [ISO88] son los siguientes: autentificación, confidencialidad, control de acceso, integridad y no repudio.

El proyecto CRIPTO perteneciente al Plan nacional de banda ancha (PlanBA) fue pionero en contemplar la posibilidad de integrar servicios de seguridad en la RDSI-BA. Otros trabajos desarrollados en paralelo y posteriormente [DEN95, STE95, LAU96] reflejan la importancia de este tema.

Existen muchas opciones para la ubicación de cada servicio de seguridad dentro de la arquitectura de red. En este sentido se ha realizado un estudio exhaustivo apuntando las ventajas e inconvenientes que se han considerado más importantes en

cada una de ellas. Se ha contemplado con especial detalle la ubicación de los servicios de confidencialidad e integridad, por implicar el procesado de toda la información y resultar por lo tanto críticos en una red de alta velocidad.

La opción más natural consiste en ofrecer los servicios de seguridad en los terminales extremos. Si se opta por ella, la arquitectura concreta del terminal puede jugar un papel fundamental en la ubicación de los servicios de seguridad. Un ejemplo claro es la arquitectura de seguridad definida por el proyecto CRIPTO, enormemente condicionada por la arquitectura del terminal multimedia TEMA.

Aunque existen diversas opciones para ubicar los servicios de seguridad, a la hora de diseñar una arquitectura de seguridad se debe optar por una en concreto. Para ello es necesario definir previamente el escenario a proteger y sentar una serie de requisitos a exigir al sistema de seguridad.

En este sentido se han sentado una serie de requisitos para el sistema de seguridad, en base a los cuales se ha decidido la ubicación de los servicios de confidencialidad e integridad por encima de la capa AAL del modelo de referencia de protocolos de la RDSI-BA. También se ha decidido la ubicación a nivel de aplicación de los servicios de control de acceso, no repudio y autentificación, así como del mecanismo de gestión de claves.

Seguidamente se ha propuesto un sistema integrado de seguridad capaz de ofrecer servicios de seguridad a aplicaciones multimedia sensibles de todo tipo. Además, el sistema es transparente para las aplicaciones que utilicen otros protocolos de seguridad, permitiendo de esta forma la comunicación segura entre terminales pertenecientes a diferentes redes locales o metropolitanas que se comuniquen a través de la RDSI-BA.

El sistema de seguridad permite que aplicaciones sensibles transformen una conexión insegura en una conexión segura mediante la negociación de unos parámetros de seguridad y de unos identificadores de la asociación a través de una conexión de datos paralela. Ello permite la integración de toda la seguridad en el plano de usuario del MRP de la RDSI-BA, siendo innecesaria la modificación del plano de control (es decir, no debe definirse de nuevo la señalización).

El sistema de seguridad propuesto se compone de dos niveles, el SEC-AAL y el SEC-APL.

El nivel más bajo, el SEC-AAL se ubica encima del nivel de adaptación MTA, y es independiente de los protocolos de nivel superior. Cualquier familia de protocolos de nivel superior podrá ubicarse por encima del nivel SEC-AAL de forma transparente, ya

que se les ofrece la mismo interfaz que el nivel de adaptación. Este nivel ofrece el mecanismo de cifrado y los servicios de confidencialidad e integridad.

El nivel SEC-APL se ubica a nivel de aplicación, y ofrece el mecanismo de gestión de claves y los servicios de control de acceso, no repudio y autentificación. En este nivel se ubica también toda la gestión de la seguridad. Las aplicaciones sensibles solicitan servicios de seguridad especificando ciertos parámetros a través de una interfaz de programación de la aplicación (API). Esta API define una serie de primitivas de seguridad, que son usadas para negociar servicios de seguridad entre las aplicaciones y el nivel SEC-APL.

Para comunicaciones punto a punto los niveles SEC-APL situados en el terminal origen y en el destino intercambian una serie de mensajes para negociar unos parámetros de seguridad compatibles con unos requerimientos de calidad de servicio. Deberá tenerse especial cuidado con la negociación de las claves, siendo necesario el uso de protocolos autentificados de gestión de claves.

Se han definido también una serie de primitivas de comunicaciones que definen los mensajes intercambiados entre ambos niveles SEC-APL durante la fase de negociación de la asociación de seguridad y de liberación de esta asociación.

La evaluación del coste que implica la introducción de servicios de seguridad en una arquitectura de red es de vital importancia para el estudio de la viabilidad económica de la implantación de estos servicios. En efecto, los usuarios considerarán la seguridad como un servicio de valor añadido que ofrece la red y decidirán su contratación en función del coste que les suponga.

Por ello se han presentado algunos elementos de coste necesarios para ofrecer los servicios de seguridad (ver tablas 4.1 y 4.2, así como anexo b). Para el caso concreto de confidencialidad e integridad se ha evaluado el coste de su introducción, mediante la definición de funciones de eficiencia que permiten cuantificar el impacto que estos servicios provocarán tanto a los usuarios finales como a la red. Se ha demostrado cuantitativamente que es preferible la ubicación de estos servicios a niveles de red donde se manejen unidades de información relativamente grandes (las funciones de eficiencia definidas tienden a la unidad cuando ambos servicios se ofrecen sobre unidades de información de longitud grande). En la RDSI-BA en concreto ello hace poco recomendable la ubicación de estos servicios por encima del nivel MTA, siendo preferible la ubicación por encima de AAL, como en realidad se ha propuesto.

Un problema fundamental es evaluar el coste asociado al mecanismo de gestión de claves, que es sin duda el factor clave para la viabilidad de comunicaciones seguras en grandes redes multidominio como la RDSI-BA. En este sentido se ha introducido un

modelo basado en teoría de colas que permite evaluar el coste de un sistema de gestión de claves basado en una estructura jerárquica de autoridades de certificación, y que a la vez permite el correcto dimensionado del número de usuarios por centro de certificación y de la capacidad de cálculo óptima de los centros en función de su coste. Ello ha permitido relacionar parámetros de coste con parámetros de seguridad, lo que constituye un trabajo original en un campo muy poco trabajado.

Finalmente se ha propuesto un protocolo para la gestión de claves entre entidades conectadas a la RDSI-BA basado en criptografía de clave pública. Se ha partido de un protocolo de autentificación conocido considerado seguro por la comunidad internacional como el de Needham-Schroeder [NEE78] y se han realizado modificaciones para permitir la gestión de claves y aumentar su eficiencia, sin comprometer su seguridad. En este protocolo ambas entidades (típicamente terminales multimedia) participan en la generación de la clave de sesión, lo que hace innecesario confiar en la bondad del proceso de generación de claves de la otra entidad.

En una red pública no es posible que cada entidad guarde las claves públicas del resto, lo que implica el acceso a autoridades de certificación. Para minimizar el número de conexiones, y consecuentemente el tráfico de gestión de seguridad, se propone que las entidades mantengan certificados con sus claves públicas y los intercambien para la negociación de claves de sesión, reduciéndose de esta forma los accesos al centro de certificación. Además, para reducir el número de mensajes transferidos y la longitud de éstos se propone que las entidades mantengan una tabla con las claves públicas de las entidades con las que es más probable una comunicación segura. Este protocolo se utilizó para la gestión de claves en el proyecto CRIPTO.

Se ha abordado el dimensionado óptimo de esta tabla de claves públicas. Para ello se ha modelado la probabilidad que tiene una entidad de conectarse con las restantes de la red según una exponencial decreciente. De esta forma se ha hallado una longitud óptima de la tabla de claves públicas que minimiza el coste total de la gestión de claves por usuario.

Por último se ha presentado una versión modificada del protocolo que permite garantizar la autenticidad e integridad de parámetros de seguridad negociados previamente, lo que permite utilizarlo como protocolo de gestión de claves para la arquitectura de seguridad propuesta en esta tesis.

6.2 Líneas Futuras

Entre las posibles líneas de continuación de este trabajo destacan las siguientes:

- La generalización de los procedimientos de negociación de parámetros de seguridad para comunicaciones multipunto con la propuesta de sistemas de negociación de claves asociados.
- La filosofía del diseño del sistema de seguridad y del interfaz entre las aplicaciones y el nivel SEC-APL puede exportarse a otras redes. En este sentido se ha desarrollado un modelo similar para *Internet*, donde las primitivas de solicitud de servicios de seguridad son las mismas[1]. Una línea de continuación consistiría en la modificación de las aplicaciones de red existentes (por ejemplo, telnet o ftp) para que hagan uso de este sistema de seguridad.
- El estudio de los requisitos de seguridad de las diferentes aplicaciones y servicios que utilizarán la RDSI-BA (por ejemplo, vídeo a la carta interactivo, interconexión de redes, telenseñanza, teletrabajo, multiconferencia) y de cómo el sistema de seguridad propuesto puede ofrecer estos servicios a estas aplicaciones.
- Un estudio más completo del coste de los servicios de seguridad. En la actualidad este es un campo de investigación completamente abierto. Dentro de esta línea destaca un estudio más general del modelo utilizado para evaluar el coste de la gestión de claves, donde debería considerarse un modelo que contemplase varios niveles de autoridades de certificación e incluyese otros factores de coste, como la dispersión geográfica de los usuarios. Para obtener resultados en estos modelos más complicados la simulación puede jugar un papel fundamental.

[1] Aunque *Internet* no es una red orientada a conexión, muchos de los servicios que ofrece utilizan el protocolo TCP que sí lo es.

ANEXO A

Especificación de Primitivas de Seguridad

En este anexo se presenta una especificación detallada en notación ASN.1 [ISO8824] de las primitivas utilizadas para la negociación de parámetros de seguridad entre las aplicaciones y las entidades SEC-APL, así como las utilizadas para la comunicación entre entidades SEC-APL remotas.

```
Primitivas DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

--Módulo de primitivas

--Tipos básicos

```
UnsignedByte ::= INTEGER (0..255)

SignedByte ::= INTEGER (-128..127)

Unsigned2Bytes ::= INTEGER (0..65535)

Float ::= REAL

IdSocket ::= SEQUENCE
{
      dirIP                              OCTET STRING (SIZE 4),
      puerto                             OCTET STRING (SIZE 2)
}

IdEntidad ::= SEQUENCE
{
      idUsr                              INTEGER (0..65535),
      idSocket                           IdSocket
}

Pathname ::= VisibleString  (SIZE (1..255))
```

```
PassPhrase ::= VisibleString (SIZE (6..255))

IdTipoAlgoritmo ::= UnsignedByte

IdAlgoritmo ::= UnsignedByte

IdLongitudClave ::= UnsignedByte (0..5)

IdTipoClave ::= UnsignedByte (1..4)

ParametroSeguridad ::= SignedByte (-128 | -3 | -2..2)

ParametroCoste1 ::= CHOICE
{
      velocidadCalculos                [0] Float,
      retardoLinealCalculosTC1         [1] Float
}

ParametroCoste0 ::= Float

EspecificacionInformacionCoste ::= SEQUENCE
{
      velocidadCalculos          <     ParametroCoste1 DEFAULT 0,
      retardoConstanteCalculosTC0      ParametroCoste0 DEFAULT maxReal
}

InformacionCoste ::= SEQUENCE
{
      retardoLinealCalculosTC1  <      ParametroCoste1,
      retardoConstanteCalculosTC0      ParametroCoste0
}

LongitudBloque ::= Unsigned2Bytes

IdModoCriptografiaSimetrica ::= UnsignedByte (0..4)

Servicios ::= BIT STRING (SIZE 8)
{
      autentificacion                  (7),
      integridad                       (6),
      confidencialidad                 (5),
      noRepudiacionOrigen              (4),
      noRepudiacionDestino             (3)
}

EspecificacionAlgoritmo ::= SEQUENCE
{
      idTipoAlgoritmo                  [0] IdTipoAlgoritmo OPTIONAL,
      idAlgoritmo                      [1] IdAlgoritmo OPTIONAL,
      idTipoClave                      [2] IdTipoClave OPTIONAL,
      idLongitudClave                  [3] IdLongitudClave OPTIONAL,
      parametroSeguridad               [4] ParametroSeguridad OPTIONAL,
      coste                            EspecificacionInformacionCoste OPTIONAL,
      longitudBloque                   [5] LongitudBloque OPTIONAL,
      idModoCriptografiaSimetrica      [6] IdModoCriptografiaSimetrica
                           OPTIONAL
}

EspecificacionMecGestClavesSim ::= SEQUENCE
{
      idMecGestClavesSim               [0] IdAlgoritmo OPTIONAL,
      nivelSeguridad                   [1] ParametroSeguridad OPTIONAL,
      coste                            ParametroCoste0 DEFAULT maxReal,
      participacionTerceraParte        BOOLEAN OPTIONAL,
      especificacionesAlgoritmos       SET OF EspecificacionAlgoritmo OPTIONAL
}

EspecificacionMecComSegura ::= SEQUENCE
{
```

```
      idMecComSegura                  IdAlgoritmo OPTIONAL,
      serviciosSolicitados            Servicios,
      nivelSeguridad                  ParametroSeguridad OPTIONAL,
      coste                           EspecificacionInformacionCoste,
      habilitacionCompresion          [0] BOOLEAN OPTIONAL,
      participacionTerceraParte       [1] BOOLEAN OPTIONAL,
      especificacionesAlgoritmos      SET OF EspecificacionAlgoritmo OPTIONAL
}

EspecificacionServicio ::= SEQUENCE
{
      nivelSeguridadGlobalSolicitado  ParametroSeguridad OPTIONAL,
      especificacionMecGestClavesSim  [1] EspecificacionMecGestClavesSim
                                      OPTIONAL,
      especificacionMecComSegura      EspecificacionMecComSegura
}

Idprimitiva ::= ENUMERATED
{
--Primitivas de la API entre la aplicación APL y la capa SEC-APL
--Establecimiento y mantenimiento de la asociación
      primitivaSec-Associate-request          (0),
      primitivaSec-Associate-indication       (1),
      primitivaSec-Associate-response         (2),
      primitivaSec-Associate-confirm          (3),
      primitivaL-Sec-Report-indication-1      (4),
      primitivaL-Sec-Report-indication-2      (5),
      primitivaL-Sec-Report-indication-3      (6),
      primitivaR-Sec-Report-indication-1      (7),
      primitivaR-Sec-Report-indication-2      (8),
      primitivaApl-Sec-Report-indication      (9),
--Liberación de la asociación
      primitivaSec-Release-request            (10),
      primitivaSec-Release-indication         (11),
      primitivaSec-Release-response           (12),
      primitivaSec-Release-confirm            (13),
      primitivaSec-Abort-request              (14),
      primitivaSec-Abort-indication           (15),
--Primitivas usadas entre las capas SEC-APL local y remota
--Establecimiento y mantenimiento de la asociación
      primitivaSec-Associate-invoke-1         (16),
      primitivaSec-Associate-invoke-2         (17),
      primitivaSec-Associate-reply-1          (18),
      primitivaSec-Associate-reply-2          (19),
      primitivaSec-Associate-notify           (20),
--Liberación de la asociación
      primitivaSec-Release-invoke             (21),
      primitivaSec-Release-reply              (22),
      primitivaSec-Abort-notify               (23)
}

Configuracion ::= SEQUENCE OF IdAlgoritmo

InfoAcepacionMecComSegura ::= SEQUENCE
{
      aceptacionRetardoLinealCalculosTC1     BOOLEAN,
      aceptacionRetardoConstanteCalculosTC2  BOOLEAN
}

NumConfBuscadas ::= UnsignedByte

InfoNumConfig ::= SEQUENCE
{
      numConfigBuscadas               NumConfBuscadas,
      numConfigNoCumplenTC0niTC1      [0] NumConfBuscadas OPTIONAL,
      numConfigCumplenSolamenteTC0    NumConfBuscadas,
      numConfigCumplenSolamenteTC 1   NumConfBuscadas OPTIONAL
}
```

--Valores básicos

--Constantes

```
maxReal                                REAL ::= {1,10,6}
```

--Valores de los tipos relacionados con las claves

```
claveSimetrica                         IdTipoClave ::= 1
claveAsimetricaRSA                     IdTipoClave ::= 2
claveAsimetricaDSA                     IdTipoClave ::= 3
claveAsimetricaLUC                     IdTipoClave ::= 4
longVariable                           IdLongClave ::= 0
```

--Valores del parámetro de seguridad

```
superior                               ParametroSeguridad ::= 2
alto                                   ParametroSeguridad ::= 1
medio                                  ParametroSeguridad ::= 0
bajo                                   ParametroSeguridad ::= -1
inferior                               ParametroSeguridad ::= -2
noApto                                 ParametroSeguridad ::= -3
diferenteSegunTipoMecanismo            ParametroSeguridad ::= -128
```

--Valores del identificador del modo de encriptación simétrica

```
modoECB                                idModoCriptografiaSimetrica ::=0
modoCBC                                idModoCriptografiaSimetrica ::=1
modoCFB                                idModoCriptografiaSimetrica ::=2
modoOFB                                idModoCriptografiaSimetrica ::=3
modoPCBC                               idModoCriptografiaSimetrica ::=4
```

--Primitivas

--Primitivas de la API entre la aplicación APL y la capa SEC-APL

--Establecimiento y mantenimiento de la asociación

--Primitiva APL A⇒SEC-APL A : SEC-ASSOCIATE.request

```
Sec-Associate-request ::= SEQUENCE
{
```

```
        idPrimitiva-Contenido                   IdPrimitiva,
        idEntidadA                              IdEntidad,
        idEntidadB                              IdEntidad,
        especificacionServicio                  EspecificacionServicio,
        aceptacionConfiguracionSugerida         BOOLEAN OPTIONAL,
        infoFicheroBaseDatos                    Pathname,
        infoFicheroClavesAsimPrivadas           Pathname,
        infoFicheroClavesAsimPublicas           Pathname,
        passPhraseAccesoFicheroClavesPrivadas   PassPhrase OPTIONAL

    }
```

--Primitiva SEC-APL B⇒APL B : SEC-ASSOCIATE.indication

```
Sec-Associate-indication ::= SEQUENCE
{
        idPrimitiva-Contenido                   IdPrimitiva,
        idEntidadA                              IdEntidad,
        idEntidadB                              IdEntidad,
        serviciosSoportados                     Servicios,
        nivelSeguridadMecGestClavesSim          ParametroSeguridad,
        nivelSeguridadMecComSegura              ParametroSeguridad,
        costeMecGestClavesSimSolicitado         ParametroCoste0,
        costeMecComSeguraSolicitado             InformacionCoste
}
```

--Primitiva APL B⇒SEC-APL B : SEC-ASSOCIATE.response

```
Sec-Asociate-response ::= SEQUENCE
{
        idPrimitiva-Contenido                   IdPrimitiva,
        notficacionAceptacionComunicacion       BOOLEAN,
        infoFicheroBaseDatos                    Pathname,
        infoFicheroClavesAsimPrivadas           Pathname,
        infoFicheroClavesAsimPublicas           Pathname,
        passPhraseAccesoFicheroClavesPrivadas   PassPhrase OPTIONAL
}
```

--Primitiva SEC-APL A⇒APL A : SEC-ASSOCIATE.confirm

```
Sec-Associate-confirm
{
        idPrimitiva-Contenido                   IdPrimitiva
}
```

--Primitiva SEC-APL A⇒APL A : L-SEC-REPORT.indication

```
L-Sec-Report-indication ::= SEQUENCE
{
        idPrimitivaContenido                    IdPrimitiva,
        contenido                               ANY DEFINED BY idPrimitivaContenido
}

Contenido-L-Sec-Report-indication-1 ::= SEQUENCE
{
        especificacionSintacticamenteIncorrecta     BOOLEAN,
        ningunMecGestClavesSimDispLocal              [0] BOOLEAN OPTIONAL,
        ningunMecComSeguraAptoLocal                  BOOLEAN,
        configuracionMecGestClavesSimSugerida        [1] Configuracion OPTIONAL,
        configuracionMecComSeguraSugerida            [2] Configuracion OPTIONAL
}

Contenido-L-Sec-Report-indication-2 ::= SEQUENCE
{
        numConfigMecGestClavesSimLocal               [0] InfoNumConfig OPTIONAL,
```

```
        numConfigMecComSeguraLocal                  InfoNumConfig,
        configuracionMecGestClavesSimSugerida       [1] Configuracion OPTIONAL,
        configuracionMecComSeguraSugerida           [2] Configuracion OPTIONAL
}

Contenido-L-Sec-Report-indication-3 ::= SEQUENCE
{
        --indica deteccion de violación de la seguridad
}
```

--Primitiva SEC-APL A⇒APL A : R-SEC-REPORT.indication

```
R-Sec-Report-indication ::= SEQUENCE
{
        idPrimitiva-Contenido                   IdPrimitiva,
        contenido                               ANY DEFINED BY idPrimitivaContenido
}

Contenido-R-Sec-Report-indication-1 ::= SEQUENCE
{
        numConfigMecGestClavesSimRemoto         [0] InfoNumConfig OPTIONAL,
        numConfigMecComSeguraRemoto             InfoNumConfig,
        configuracionMecGestClavesSimSugerida   [1] Configuracion OPTIONAL,
        configuracionMecComSeguraSugerida       [2] Configuracion OPTIONAL
}

Contenido-R-Sec-Report-indication-2 ::= SEQUENCE
{
        --indica detección de violación de la seguridad
}
```

--Primitiva APL B⇒SEC-APL B : APL-SEC-REPORT.request

```
Apl-Sec-Report-indication ::= SEQUENCE
{
        idPrimitiva-Contenido                   IdPrimitiva
}
```

--Primitiva SEC-APL A⇒APL A : APL-SEC-REPORT.indication

```
Apl-Sec-Report-indication ::= SEQUENCE
{
        idPrimitiva-Contenido                   IdPrimitiva
}
```

--Liberación de la asociación

--Primitiva APL A⇒SEC-APL A o APL B⇒SEC-APL B : SEC-RELEASE.request

```
Sec-Release-request ::= SEQUENCE
{
        idPrimitiva-Contenido                   IdPrimitiva
}
```

--Primitiva SEC-APL B⇒APL B o SEC-APL A⇒APL A : SEC-RELEASE.indication

```
Sec-Release-indication ::= SEQUENCE
{
        idPrimitiva-Contenido                   IdPrimitiva
}
```

--Primitiva APL B⇒SEC-APL B o APL A⇒SEC-APL A : SEC-RELEASE.response

```
Sec-Release-response ::= SEQUENCE
{
      idPrimitiva-Contenido                  IdPrimitiva
}
```

--Primitiva SEC-APL A⇒APL A o SEC-APL B⇒APL B : SEC-RELEASE.confirm

```
Sec-Associate-confirm ::= SEQUENCE
{
      idPrimitiva-Contenido                  IdPrimitiva
}
```

--Primitiva APL A⇒ SEC-APL A o APL B⇒ SEC-APL B : SEC-ABORT.request

```
L-Sec-Abort-indication ::= SEQUENCE
{
      idPrimitiva-Contenido                  IdPrimitiva
}
```

--Primitiva SEC-APL B⇒APL B o SEC-APL A⇒APL A : SEC-ABORT.indication

```
Sec-Abort-indication ::= SEQUENCE
{
      idPrimitiva-Contenido                  IdPrimitiva
}
```

--Primitivas usadas entre las capas SEC-APL local y remota

--Establecimiento y mantenimiento de la asociación

--Primitiva SEC-APL A⇒SEC-APL B : Sec-Associate.invoke

```
Sec-Associate-invoke ::= SEQUENCE
{
      idPrimitivaContenido                   IdPrimitiva,
      contenido                              ANY DEFINED BY idPrimitivaContenido
}

Contenido-Sec-Associate-invoke-1 ::= SEQUENCE
{
      idEntidadA                             IdEntidad,
      idEntidadB                             IdEntidad,
      serviciosSoportados                    Servicios,
      nivelSeguridadMecGestClavesSim         ParametroSeguridad,
      nivelSeguridadMecComSegura             ParametroSeguridad,
      costeMecGestClavesSimSolicitado        ParametroCoste0,
      costeMecComSeguraSolicitado            InformacionCoste,
      --datos para reducir la información de disponibilidad remota necesaria
      conjuntoMecComSeguraAptosLocal         SET OF IdAlgoritmo,
      conjuntoMecGestClavesSimDispLocal      SET OF IdAlgoritmo,
      idAlgoritmosEspecificados              SET OF SEQUENCE
      {
            idTipoAlgoritmo                  IdTipoAlgoritmo,
            idAlgoritmo                      IdAlgoritmo
      }
}

Contenido-Sec-Associate-invoke-2 ::= SEQUENCE
{
```

```
        listaConfiguracionesMecComSegura        [0] SEQUENCE OF Configuracion
                                                OPTIONAL,
        listaConfiguracionesMecGestClavesSim    [1] SEQUENCE OF Configuracion
                                                OPTIONAL
}

--Primitiva SEC-APL B⇒SEC-APL A : Sec-Associate.reply

Sec-Associate-reply ::= SEQUENCE
{
        idPrimitivaContenido                    IdPrimitiva,
        contenido                               ANY DEFINED BY idPrimitivaContenido
}

Contenido-Sec-Associate-reply-1 ::= SEQUENCE
{
        informacionDisponibilidadRemoto         SET OF SEQUENCE
        {
                idTipoAlgoritmo                 IdTipoAlgoritmo,
                idAlgoritmo                     IdAlgoritmo
        },
        informacionClavesAsimDispRemoto SET OF SEQUENCE
        {
                idTipoClave                     IdTipoClave,
                idLongitudClave                 IdLongitudClave
        }
}

Contenido-Sec-Associate-reply-2 ::= SEQUENCE
{
        listaAceptConfigMecComSegura    [0] SEQUENCE OF
                                        InfoAceptacionMecComSegura OPTIONAL,
        listaAceptConfigMecGestClavesSim[1] SEQUENCE OF BOOLEAN OPTIONAL
}

--Primitiva SEC-APL B⇒SEC-APL A o SEC-APL A⇒SEC-APL B : Sec-Associate.notify

Sec-Associate-notify ::= SEQUENCE
{
        idPrimitiva-Contenido           IdPrimitiva,
        codigoNotificacion              NotificacionSec-Associate-notify
}

NotificacionSec-Associate-notify ::= ENUMERATED
{
        aplBRechazaEstablecimientoCom           (0),
        deteccionViolacionSeguridad             (1)
}

--Liberación de la asociación


--Primitiva SEC-APL A⇒SEC-APL B o SEC-APL B⇒SEC-APL A : Sec-Release.invoke

Sec-Release-invoke ::= SEQUENCE
{
        idPrimitiva-Contenido                   IdPrimitiva
}

--Primitiva SEC-APL B⇒SEC-APL A o SEC-APL A⇒SEC-APL B : Sec-Release.reply

Sec-Release-reply ::= SEQUENCE
{
        idPrimitiva-Contenido                   IdPrimitiva
}
```

```
--Primitiva SEC-APL B⇒SEC-APL A o SEC-APL A⇒SEC-APL B : Sec-Abort.notify

      Sec-Abort-notify ::= SEQUENCE
      {
            idPrimitiva-Contenido                  IdPrimitiva
      }

END
```

ANEXO B

Coste de los Servicios de Seguridad

En este anexo se enumeran los elementos de coste que supone la implantación de servicios de seguridad en una red de comunicaciones, tanto para cada entidad como para la red.

En primer lugar la tabla 1 considera el coste asociado a la negociación de parámetros de seguridad, necesario para arquitecturas de seguridad como la presentada en esta tesis. La tabla 2 considera individualmente el coste de ofrecer cada uno de los servicios de seguridad definidos por [ISO88]. La tabla 3 presenta el coste asociado con la gestión de claves, que aparece en cada uno de los servicios anteriores. Se ha considerado criptosistema de clave pública para claves maestras y entorno de autoridades de certificación X.509. Podrían añadirse detalles para otras configuraciones.

Este anexo pretende completar el estudio presentado en el capítulo 4, con tablas más exhaustivas en las que se consideran más elementos. Sin embargo se debe reconocer que sólo se cubren algunas de las opciones posibles y de los costes que se consideran más importante. Los elementos que configuran el coste final para un sistema en concreto deben ser analizados con mucho mayor detalle.

NEGOCIACIÓN DEL SERVICIO DE SEGURIDAD	COSTE PARA LA ENTIDAD	COSTE PARA LA RED
	Retardo • Negociación de los servicios de seguridad deseados y de sus parámetros • Ejecución de algoritmos de decisión para decidir configuraciones de seguridad en función de requerimientos de aplicaciones	Tráfico y conexiones • Tráfico para la comunicación y el acuerdo de los servicios de seguridad a utilizar y de sus parámetros

Tabla 1. Coste de la negociación del servicio de seguridad.

SERVICIO DE SEGURIDAD	COSTE PARA LA ENTIDAD	COSTE PARA LA RED
Autentificación	**Autentificación de entidad**	
	Memoria y retardo • Protocolo de autentificación de entidad (que eventualmente incluye la distribución de claves de sesión). Uso de la información propia de gestión de claves. Ver tabla 3	Elementos • Servidores de autentificación (terceras partes fiables) Elementos, tráfico y conexiones • Protocolo de autentificación de entidad (que eventualmente incluye una distribución de claves de sesión). Uso de la información propia de gestión de claves. Ver tabla 3 Tráfico y conexiones • Conexión y tráfico de la entidad con una tercera parte
	Autentificación de origen de los datos	
	Memoria y retardo • Uso de la información propia de la gestión de claves. Puede requerirse la distribución de claves de sesión (autentificación basada en el cifrado, o en un checksum criptográfico) . Ver tabla 3 Retardo • Generación de códigos de autentificación de mensaje (MACs), firmas digitales (ver no repudiación de origen) y otros cálculos - Cifrado y descifrado - Cálculo de checksum criptográfico - Cálculo de funciones hash - Verificación y comparación - Otros cálculos para la generación de mensajes (unión de partes)	Elementos, tráfico y conexiones • Uso de la información propia de la gestión de claves. Puede requerirse la distribución de claves de sesión. Ver tabla 3 Tráfico y conexiones • Incremento de tráfico debido al incremento de la longitud y el número de los mensajes

Tabla 2. Coste de los servicios de seguridad .

SERVICIO DE SEGURIDAD	COSTE PARA LA ENTIDAD	COSTE PARA LA RED
Confidencialidad	**Confidencialidad de los datos**	
	Memoria y retardo • Distribución de claves de sesión. Uso de la información propia de la gestión de claves. Ver tabla 3 Retardo • Cifrado y descifrado de los mensajes Hardware • Si se necesita gran velocidad de cifrado y descifrado, conviene que estas funciones las desempeñe un elemento hardware	Elementos, tráfico y conexiones • Distribución de claves de sesión. Uso de la información propia de la gestión de claves. Ver tabla 3 • Las técnicas de cifrado no incrementan la longitud de los mensajes (excepto si es necesario introducir padding para algoritmos en bloque), sin embargo, es necesario negociar qué algoritmo, claves, etc. se utilizarán. Ver negociación de servicios (tabla 1)
	Confidencialidad del flujo de tráfico	
	Retardo • Las técnicas de data padding (usadas conjuntamente con el cifrado) incrementan la longitud de los mensajes hasta una longitud fija, necesitándose un mayor tiempo de transmisión • Técnicas de envío de datos en partes separadas independientes	Tráfico y conexiones • Incremento de tráfico debido a las técnicas de data padding • Las técnicas de envío de datos en partes separadas independientes pueden requerir conexiones adicionales

SERVICIO DE SEGURIDAD	COSTE PARA LA ENTIDAD	COSTE PARA LA RED
Control de acceso	Memoria • Almacenamiento de tickets de acceso a los servicios • Almacenamiento de claves para comunicación con los distintos elementos y otros datos Retardo • Protocolo con el servidor de autentificación, con el servidor de entrega de tickets y con el servidor objeto del control de acceso – Espera debida al intercambio de mensajes – Cifrado y descifrado – Operaciones relacionadas con timestamps – Verificación y comparación – Otros cálculos para la generación de los mensajes (unión de partes)	Elementos • Servidores de autentificación • Servidores de entrega de tickets • Adaptación de los servidores objeto del control de acceso Tráfico y conexiones • Tráfico y conexiones adicionales con los distintos elementos, tanto de la entidad como del servidor • Actualización de información sobre permisos de acceso • Revocación de control de acceso

Tabla 2. Coste de los servicios de seguridad

SERVICIO DE SEGURIDAD	COSTE PARA LA ENTIDAD	COSTE PARA LA RED
Integridad	Memoria y retardo • Uso de la información propia de la gestión de claves. Ver tabla 3, claves maestras Retardo • Puede requerirse la distribución de claves de sesión (integridad basada en el cifrado, o en un checksum criptográfico). Ver tabla 3, claves de sesión • Generación de valores de comprobación de la integridad (ICVs), firmas digitales (ver no repudiación de origen) y otros cálculos • Soluciones similares a la autentificación del origen de los datos	Elementos, tráfico y conexiones • Uso de la información propia de la gestión de claves. Ver tabla 3, claves maestras Tráfico y conexiones • Puede requerirse la distribución de claves de sesión. Ver tabla 3, claves de sesión • Incremento de tráfico debido al incremento de la longitud y el número de los mensajes • Soluciones similares a la autentificación del origen de los datos
No repudiación	Memoria y retardo • Uso de la información propia de la gestión de claves. Ver tabla 3 Memoria • Almacenamiento de firmas digitales o de reconocimientos de entrega (si no lo almacena una tercera parte) según se trate de no repudiación de origen o de entrega respectivamente y según se sea emisor o receptor Retardo • Generación de firmas digitales (si no las genera una tercera parte) – Cifrado y descifrado – Cálculo de funciones hash – Verificación y comparación – Operaciones relacionadas con timestamps – Otros cálculos propios del algoritmo • Resolución de disputas	Elementos • Árbitros (terceras partes fiables) Elementos, tráfico y conexiones • Uso de la información propia de la gestión de claves. Ver tabla 3 Tráfico y conexiones • Conexión on-line o in-line y tráfico de la entidad con una tercera parte (para firmas digitales arbitradas, para resolución de disputas) • Incremento de tráfico debido al incremento de la longitud y el número de los mensajes

Tabla 2. Coste de los servicios de seguridad.

GESTIÓN DE CLAVES	COSTE PARA LA ENTIDAD	COSTE PARA LA RED
Claves maestras (públicas en sistemas asimétricos)	Memoria • Almacenamiento de la propia pareja de claves privada y pública • Tabla de claves maestras - Claves públicas de un número determinado de entidades con las que es más probable comunicarse junto al identificador de la entidad - Clave pública del centro de autentificación - Información asociada a la clave (identificador de usuario, tiempo de expiración, nivel de seguridad) - Datos para la gestión de la tabla (datos para facilitar la búsqueda, actualizar la tabla) - Información para la autentificación e integridad de la tabla • Lista de revocaciones - Lista de claves públicas no válidas e información asociada - Información para la autentificación e integridad de la lista • Registro de nonces para detectar repeticiones Retardo • Gestión de la tabla de claves maestras - Búsqueda - Actualización - Cálculos (algunos relacionados con la autentificación de la tabla) • Comprobaciones en la lista de revocaciones y actualización • Comprobaciones en el registro de nonces y actualización • Obtención de certificados para claves públicas que no se encuentran en la tabla (participación off-line de una tercera parte) - Comunicación con una o varias autoridades de certificación - Verificación de los certificados	Elementos • Autoridades de certificación o centros de distribución de certificados (terceras partes fiables) Conexiones y tráfico • Conexión y tráfico de la entidad con una o varias terceras partes para la obtención de certificados • Conexiones y tráfico para la comunicación de revocaciones

Tabla 3. Coste de la gestión de claves

GESTIÓN DE CLAVES	COSTE PARA LA ENTIDAD	COSTE PARA LA RED
Claves de sesión (secretas en sistemas simétricos)	Memoria • Almacenamiento de la clave de sesión Retardo • Protocolo de autentificación de entidad que incluye la distribución de claves de sesión (posible participación on-line de una tercera parte o centro de distribución de claves) - Espera debida al intercambio de mensajes - Cifrado y descifrado - Generación de números aleatorios (nonces y claves de sesión) - Operaciones relacionadas con timestamps - Verificación y comparación - Otros cálculos para la generación de los mensajes (unión de partes, transformaciones aplicadas a los nonces)	Tráfico y conexiones • Protocolo de autentificación de entidad que incluye la distribución de claves de sesión - Tráfico debido al intercambio de mensajes

Tabla 3. Coste de la gestión de claves

Referencias Bibliográficas

[ATM93] ATM: User-Network Interface Specification, Version 3.0, The ATM Forum, PTR Prentice Hall, Englewood Cliffs, N. J. 1993.

[BAN89] M. Burrows, M. Abadi, R. Needham, "A Logic of Authentication", Proc. of the 12th ACM Symposium on Operating Systems Principles, Litchfield Park, Arizona, 1989. Published as ACM Operating Systems Review, 23 no. 4. 1989.

[BEK82] H. Beker, F. Piper, *Cipher Systems: the Protection of Communications.* London, Northwood Books,1982.

[BEL91] M. Beller, L. Chang, Y. Yacobi."Privacy and Authentication on a Portable Communications System". Proceedings of the GLOBECOM, 1991.

[BIR93] R. Bird et al., "Systematic Design of a Family of Attack-Resistant Authentication Protocols". *IEEE Journal on Selected Areas in Communications.* Volume 11. Number 5. June 1993.

[BLU84] M. Blum, S. Micali. "How to generate cryptographically strong sequences of pseudo-random bits". SIAM J. Comput. 13 (1984), pp. 850-864.

[BLU86] L. Blum, M. Blum, M. Shub. "A simple unpredictable pseudo-random number generator". SIAM J. Comput. 15 (1986), pp. 364-383.

[BRU84] J.O. Bruer. "On pseudo random sequences as crypto generators" Proceedings of Int. Zurich Seminar on Digital Communicationes, Switzerland, 1984.

[BRA88] G. Brassard. "Modern Cryptology" Tutorial Lecture Notes in Computer Science. Springer-Verlag. 1988.

[BRI93] E. Brickell, D. Denning, S. Kent, D. Maher, *SKIPJACK Review Interim Report: The SKIPJACK Algorithm.* July 28,1993 report to the public; available from Georgetown University, Office of Public Affairs, Washington; DC. 1993.

[CCI89] CCITT, Recommendation X.509*: The Directory - Authentication Framework.* Geneva, 1989. (ISO-9594-8).

[CCI92a] BROADBAND ASPECTS OF ISDN, Recommendation I.121, CCITT & ITU, Geneva, 1992.

[CCI92b] VOCABULARY OF TERMS FOR BROADBAND ASPECTS OF ISDN, Recommendation I.113, CCITT & ITU, Geneva, 1992.

[CCI92c] B-ISDN SERVICE ASPECTS, Recommendation I.211, CCITT & ITU, Geneva, 1992.

[CCI92d] B-ISDN PROTOCOL REFERENCE MODEL AND ITS APPLICATION, Recommendation I.321, CCITT & ITU, Geneva, 1992.

[CCI92e] B-ISDN USER-NETWORK INTERFACE, Recommendation I.413, CCITT & ITU, Geneva, 1992.

[CCI92f] B-ISDN USER-NETWORK INTERFACE PHYSICAL LAYER SPECIFICATION, Recommendation I.432, CCITT & ITU, Geneva, 1992.

[CCI92g] B-ISDN ATM ADAPTATION LAYER (AAL) FUNCTIONAL DESCRIPTION, Recommendation I.362, CCITT & ITU, Geneva, 1992.

[CCI92h] B-ISDN ATM ADAPTATION LAYER (AAL) SPECIFICATION, Recommendation I.363, CCITT & ITU, Geneva, 1992.

[CCI92i] B-ISDN ATM LAYER SPECIFICATION, Recommendation I.361, CCITT & ITU, Geneva, 1992.

[CCI92j] *The Directory: Overview of concepts, Models and Services.* Recommendation X.500, CCITT & ITU, 1992.

[CEÑ96] S. Ceña, "Seguridad en Comunicaciones TCP/IP. Desarrollo de un Protocolo Autentificado de Gestión de Claves", Proyecto Final de Carrera, UPC, ETSETB, Director del Proyecto: J. Forné, Junio 1996.

[CHO94] S. Chockhani, "Towards a National Public Key Infrastructure". IEEE Communications Magazine. Vol. 32. No. 9. September 1994.

[CRU95] E. Cruselles, M. Soriano, J. Forné, J.L. Melús, "Secure Communications in Broadband Networks". Proc. Third International Conference on Telecommunication Systems. Modelling and Analysis. Nashville, Tennesse. USA. Marzo 1995.

[CRUZ93] L. de la Cruz, Ll. Cedó, J. Forné. "Implementación de un Sistema de Seguridad en una Red Local Ethernet". URSI-93. Valencia. 1993. pp 33-37.

[CRUZ94] L. de la Cruz "Diseño y realización del módulo de cifrado y del protocolo de seguridad para un bridge cifrador sobre redes Ethernet", Proyecto Final de Carrera, UPC, ETSETB, Director del Proyecto: J. Forné, Junio 1994.

[DAV84] Davies and Price. *Security for Computer Networks* John Wiley & Sons. 1984.

[DEN95] R. H. Deng, L. Gong, A. A. Lazar, "Securing data transfer in Asynchronous Transfer Mode networks", Proceedings of Globecom'95, Singapore, pp 1198-1202, November 1995.

[DEP93] M. de Prycker, "Asynchronous Transfer Mode. Solution for Broadband ISDN", Second Edition, Ellis Horwood, Chichester, England, 1993.

[DES89] Y.G. Desmedt. "Cryptanalysis of conventional and public key cryptosystems" Proc. SPRCI´89. Roma, Nov. 1989.

[DIF76] W. Diffie and M. E. Hellman, "New directions in cryptography" *IEEE Transactions on Information Theory*, Vol. IT-22, No. 6, November 1976, pp. 644-654.

[DIF82] W. Diffie, "Conventional versus public key cryptosystems," in G. J. Simmons, Ed., Secure Communications and Asymetric Cryptosystems, pp. 41-72. Boulder, CO: Westview Press, 1982.

[DIF92] W. Diffie, P. C. van Oorschot, M. J. Wiener, "Authentication and Authenticated Key Exchanges". *Designs, Codes and Cryptography.* Kluwer Academic Publishers, The Netherlands. 1992.

[ELG85] T. ElGamal, ."A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logaritms". *IEEE Transactions on Information Theory*, July 1985.

[FORD94] W. Ford. *Computer Communications Security. Principles, Standard Protocols and Techniques.* Prentice Hall. 1994.

[FOR91] J. Forné, F. Recacha, X. Simón, M. Soriano. "Desarrollo de un Sistema de Seguridad para la Red UPCNet basado en Bridges Cifradores" Palma de Mallorca. I Reunión Española Sobre Criptología. Sept. 1991. pp.41-45.

[FOR93] J. Forné, M. Soriano, J.L. Melús, F. Recacha. "Hardware Implementation of a Secure Bridge in Ethernet Environment" *Proceedings of the IEEE Globecom'93.* Houston (USA). Noviembre 1993. pp. 177-181.

[FOR94a] J. Forné, J. L. Melús, F. Recacha, M. Soriano. "The Cripto Project: Security in Broadband Communications". Brighton (Inglaterra). Poster presentado en ESORICS-94 (European Symposium on Research in Computer Security). Nov. 1994.

[FOR94b] J. Forné, F. Recacha. "Gestión de Claves en un Terminal Multimedia para RDSI-BA". III Reunión Española sobre Criptología. Barcelona, Noviembre 1994.

[FOR95a] J. Forné, F. Recacha, M. Soriano, J. L. Melús "The Cripto Project Architecture: A Spanish Experience in Broadband Networks Security", *Proceedings ICC'95 (IEEE International Conference on Communications).* Seattle (USA), Jun. 95.

[FOR95b] J. Forné, M. Soriano, F. Recacha, J. L. Melús, "Seguridad en redes de banda ancha", revista MUNDO ELECTRONICO. Num. 260. Oct. 95.

[FOR95c] J. Forné, M. Soriano, J. L. Melús, "Criptografía y seguridad en redes de comunicación", revista NOVATICA, Ago. 95

[FOR96a] J. Forné, J. L. Melús,. An Integrated Solution for Secure Communications over B-ISDN. *Communications and Multimedia Security II.* Chapter 9. CHAPMANN-HALL. 1996.

[FOR96b] J. Forné, J. L. Melús, D. Rebollo, "Gestión Eficiente de Claves en Grandes Redes" Actas de la IV Reunión Española sobre Criptología. Valladolid. Sept. 96.

[FOR96c] J. Forné, J. L. Melús, D. Rebollo, "Securing Multimedia Applications over B-ISDN". *Proceedings of the PROMS'96 (3rd International Workshop on Protocols for Multimedia Systems).* Madrid. Oct. 96.

[FUM93] W. Fumy and P. Landrock, "Principles of Key Management". *IEEE Journal on Selected Areas in Communications.* Volume 11. Number 5. June 1993.

[GAN94] R. Ganesan, "Securing the Information Superhighway", IEEE Communications Magazine, September 1994, vol. 32, n. 9, pp. 28-30.

[GEF73] P.R. Geffe. "How to protect data with ciphers that are really hard to break", Electronics, Enero 1973.

[GOL84] S. Goldwasser, S. Micali. "Probabilistic encryption and how to play mental poker keeping secret all partial information" J. Comput. Sys. Sci. Vol 28, nº2, Abril 84.

[GON90] L. Gong, R. Needham, R. Yahalom, "Reasoning about Belief in Cryptographic Protocols", Proc. 1990 Symp. on Research in Security and Privacy. 1990. pp 234-248.

[GON93] L. Gong, "Lower Bounds on Messages and Rounds for Network Authentication Protocols", Virginia (USA).. 1st ACM Conference on Computer and Communications Security. Noviembre 1993. pp 26-37.

[GUI94] D. de la Guía, F. Montoya, E. Valderrama, Ll. Porta "ASIC_CRIPTO: un circuito integrado para el módulo de seguridad del PLANBA". III Reunión Española sobre Criptología. Barcelona, Noviembre 1994.

[HAN91] R. Händel, M. N. Huber, "Integrated Broadband Networks. An Introduction to ATM-Based Networks", Addison-Wesley, 1991.

[HEN90] J. Henshall, S. Shaw, OSI Explained. Ellis Howard Series in Computer Communications and Networking. ELLIS HOWARD, 1990.

[IEE89] IEEE 802.10a. Standard for Interoperable LAN Security (SILS): Part A - The Model. Unapproved Draft. 1989.

[IEE91] IEEE 802.10b. Standard for Interoperable LAN Security (SILS): Part B - Secure Data Exchange. Unapproved Draft. 1991.

[IEE92] IEEE 802.10c. Standard for Interoperable LAN Security (SILS): Part C - Key Nanagemenent Proposal. Unapproved Draft. 1992.

[ISO84] ISO 7498. Information Processing Systems - Open Systems Interconnection - Basic Reference Model. 1984.

[ISO88] ISO 7498-2, Information Processing Systems - Open Systems Interconnection. Reference Model - Part 2: Security Architecture. 1988.

[ISO8824] ISO/IEC 8824: Information Technology - Open Systems Interconnection - Abstract Syntax Notation One (ASN.1). (Also ITU Recommendation X.680, X.681, X.682 and X.683).

[KIR90] K. E. Kirkpatrick."P802.10-90/16: SILS Overview", Documentation of a 802.10 plenary. MITRE Corporation (1990).

[KLE75] L. Kleinrock, *Queueing Systems, Volume I : Theory*, John Willey & Sons, Inc. 1975.

[LAI90] X. Lai, J. Massey "A proposal for a New Block Encryption Standard". *Proceedings EUROCRYPT'90*; Springer-Verlag. 1990.

[LAI91] X. Lai, J. Massey "Markov Ciphers and Differential Cryptoanalysis". *Proceedings EUROCRYPT'91*; Springer-Verlag. 1991.

[LAI92a] X. Lai. *On the Design and Security of Block Ciphers*. Konstanz, Germany: Hartung-Gorre, 1992.

[LAI92b] X. Lai, R.A. Ruepple, J. Woollven "A fast cryptographic checksum algorithm based on stream ciphers". Lecture Notes in Computer Science 718. Advances in Cryptology-Proc. Auscrypt'92. Springer-Verlag , Queensland, Dic. 1992. pp. 339 - 348.

[LAM89] Paul A. Lambert. "Actual Considerations for LAN Security Protocols", Lecture Notes in Computer Science LANSEC'89, 1989. pp. 5-11.

[LAM92] Paul. A. Lambert.*"Security for Universal Personal Communications"*. IEEE, 1992.

[LAU96] M. Laurent, Security Flows Analysis of the ATM Emulated LAN Architecture. *Communications and Multimedia Security II.* Chapter 4. CHAPMANN-HALL. 1996.

[LEE93] B. G. Lee, M. Kang, J. Lee, "Broadband Telecommunications Technology", Artech House, Boston, 1993.

[LEV93] S. Levy, "Crypto Rebels". *Wired*, May-June 1993.

[MAC95] Macq, B. M. and Quisquater, J. J., "Cryptology for digital TV broadcasting", *Proc. of the IEEE*, vol. 83, no. 6, pp. 944-957. 1995

[MAR93] Martínez del Cerro, F. J., Fernández-Amigo Barranco, J. "Servicios multimedia: TEMA/PLANBA". *Comunicaciones de Telefónica I+D.* Vol. 4, N° 2, julio-diciembre 1993.

[MIL86] V. S. Miller, "Use of Elliptic Curves in Cryptography", Proceedings of CRIPTO'85, LNCS, Springer-Verlag, 1986. pp. 417-426.

[MIN89] S. E. Minzer, "Broadband ISDN and Asynchronous Transfer Mode (ATM)", IEEE Communications Magazine, vol. 27, No. 9, September 1989, pp. 17-24.

[NBS77] National Bureau of Standards "Data Encryption Standard". Federal Information Processing Standards. Publication 46. January 1977.

[NEE78] Needham, R. M. and Schoroeder, M. D., "Using encryption for authentication in large networks of computers", Communication of the ACM, Vol. 21. No. 12, December 1978, pp. 993-999.

[NEE87] Needham, R. M. and Schoroeder, M. D., "Authentication revisited", ACM Operating Systems Review, Vol. 21. No. 1, 1987.

[NES90] D. M. Nesset, "A Critique of the Burrows, Abadi and Needham Logic", ACM Operating Systems Review, 24, 2, pp. 35-38, 1990.

[NEU95] B. C. Neuman, "Security, Payment, and Privacy for Network Commerce", IEEE Journal on Selected Areas in Communications, Vol. 13, No. 8, October 1995.

[PAL94] E. Pallarés, "Desenvolupament del mòdul de gestió de claus i d'administració per a un bridge segur", Proyecto Final de Carrera, UPC, ETSETB, Director del Proyecto: J. Forné, Junio 1994.

[PIP82] F. Piper. "Stream Cipher". Proc. Workshop on Cryptography, Springer-Verlag. Lecture Notes in Computer Science, No 149, New York. 1982.

[PLE77] V. S. Pless. "Encryption schemes for computer confidentiallity", IEEE Trans. Comput. vol C-26, pp. 1133-1136, Nov. 1977.

[REB96] D. Rebollo, "Seguridad en Internet. Desarrollo de una plataforma de negociación de servicios de seguridad para sistemas UNIX", Proyecto Final de Carrera, UPC, ETSETB, Director del Proyecto: J. Forné. 1996.

[REC93] F. Recacha., J.L. Melús, X. Simón, M. Soriano, J. Forné. "Secure Data Transmission in Extended Ethernet Environments". IEEE Journal on Selected Areas in Communications. Vol.11. No.5 Junio 1993. Pp.794-803.

[REC95] F. Recacha, J. Forné, J.L. Melús, "A solution to secure inter-networking with Metropolitan Area Networks". Minutes of the EEC DG XIII/B (InfoSec) WorkShop on Security Dependability and Safety of Communications Systems and Services, Brussels (June 1995).

[REC96] F. Recacha, Seguridad en Redes Locales de Datos. Contribución a la Seguridad de Redes 802.3 / Ethernet Extendidas. Tesis Doctoral. U.P.C. 1996.

[RFC1319] B. S. Kalisky, "The MD2 Message Digest Algorithm", RFC 1319, Apr. 1992.

[RFC1320] R. L. Rivest, "The MD4 Message Digest Algorithm", RFC 1320, Apr. 1992.

[RFC1321] R. L. Rivest, "The MD5 Message Digest Algorithm", RFC 1321, Apr. 1992.

[RFC1421] J. Linn, "Privacy Enhancement for Internet Electronic Mail: Part I: Message Encipherment and Authentication Procedures", RFC 1421, Feb. 1993.

[RFC1422] S. T. Kent, "Privacy Enhancement for Internet Electronic Mail: Part II: Certificate-Based Key Management", RFC 1422, Feb. 1993.

[RFC1423] D. Balenson, "Privacy Enhancement for Internet Electronic Mail: Part III: Algorithms, Modes, and Identifiers", RFC 1423, Feb. 1993.

[RFC1424] B. S. Kalisky, "Privacy Enhancement for Internet Electronic Mail: Part IV: Key Certificates and Related Services", RFC 1424, Apr. 1992.

[RIV77] Rivest, Shamir, Adleman "A method for obtaining digital signatures and public-key cryptosystems" MIT Laboratory for Computer Science. Technical Memo LCS/TM82 (April 1977).

[RUE86] R. A. Rueppel. Analysis and Design of Stream Ciphers. Springer-Verlag (1986).

[SAL90] Salomaa A. Public-Key Cryptography. Springer-Verlag.1990.

[SCH96] B. Schneier, Applied Cryptography, John Wiley & Sons, 1996.

[SCHW94] M. Schwartz, "Broadband Integrated Networks", Classnotes (EE E6762), Department of Electrical Engineering, Columbia University, New York, 1994.

[SES92] SES/Workbench. Reference manual. 1992.

[SHA49] C. E. Shannon "Communication theory of secrecy systems". Bell Syst. Tech. J. vol 28, pp. 656-715, Oct. 1949.

[SIM92] G. Simmons. Contemporary Cryptology. IEEE Press 1992.

[SIR94] P. Siron, B. D'Ausbourg. "A Secure Medium Access Control Protocol: Security versus Performances". ESORICS'94. Brighton (UK). Nov. 1994.

[SMI93a] P. Smith, "LUC Public-Key Encryption: A Secure Altenative to RSA". Dr. Dobb's Journal. January 1993.

[SMI93b] P. Smith, M. Lennon "LUC: A New Public Key System". Proceedings Ninth International Conference on Information Security, IFIP/Sec. 1993.

[SOR93a] M. Soriano, J. Forné, J.L. Melús, F. Recacha. "Implementation of a Security System in a Local Area Network Environment" Minneapolis (USA). 18th Annual Conference on Local Computer Networks. Septiembre 1993. pp. 230-237.

[SOR93b] M. Soriano, J. Forné, F. Recacha, J.L. Melús. "A Particular Solution to Provide Secure Communications in an Ethernet Environment". Virginia (USA). 1st ACM Conference on Computer and Communications Security. Noviembre 1993. pp 17-25.

[SOR96a] M. Soriano, J. Forné, J.L. Melús. "Linear Complexity Stability in Stream Ciphering for High Speed Networks." 4th. International Conference on Telecommunications Systems. Nashville, Tennessee, Marzo 1996.

[STA95a] W. Stallings. Network and Internetwork Security. Principles and Practice . Prentice-Hall. 1995.

[STA95b] W. Stallings. Protect your Privacy: the PGP User's Guide. Prentice-Hall. 1995.

[STA96] W. Stallings. "IPv6: The New Internet Protocol". IEEE Communications Magazine. July 1996. Pp. 96-108.

[STE95] D. Stevenson, N. Hillery, G. Byrd, "Secure Communications in ATM Networks." Communications of the ACM. Vol. 38, No. 2. Feb 1995.

[TAY93] R. Taylor. "An Integrity Check Value Algorithm for Stream Ciphers". Lecture Notes in Computer Science, No 773. Advances in Cryptology-Proc. Crypto '93. Springer-Verlag , 1994.

[TSU93] G. Tsudik, E. Van Herreweghen, "On Simple and Secure Key Distribution", Virginia (USA). 1st ACM Conference on Computer and Communications Security. Noviembre 1993. pp. 49-57.

[VOY83] V. L. Voydock, S. T. Kent, "Security Mechanisms in High-Level Network Protocols" ACM Computing Surveys, Vol. 15, No. 2, June 1983, pp. 135-171.

[WIR86] N. Wirth, Algorithms and Data Structures. Prentice Hall, Inc. 1986.

[YAH93] R. Yahalom,"Optimality of Multi-Domain Protocols", Virginia (USA). 1st ACM Conference on Computer and Communications Security. Noviembre 1993. pp. 38-48.

[ZHE93] Y. Zheng, J. Seberry. "Immunizing public key cryptosystems against chosen ciphertext attacks". IEEE Journal on Selected Areas in Communications. Volume 11. Number 5. June 1993. pp. 715-724.

[ZOR94] V. Zorkadis, "Security versus Performance Requirements in Data Communication Systems". Proc. ESORICS 94, Springer-Verlag. Lecture Notes in Computer Science, No. 875, Brighton, United Kingdom, Nov. 1994. pp. 19-30.