Capítulo 5

Una infraestructura de autorización basada en certificados

Una vez analizadas las alternativas y las posibilidades de los certificados de credencial, se trata ahora de presentar el conjunto de componentes que dan lugar a la infraestructura de autorización diseñada. En primer lugar, se verá cuál es la estructura general del sistema y cómo está relacionada con la infraestructura de clave pública vista en el capítulo 3. A continuación se introduce el marco de intercambio de información relativa a autorizaciones, tanto su diseño general como el protocolo que implementa las recomendaciones. Posteriormente, se describen tanto las entidades como las especificaciones relativas al sistema de gestión distribuida de credenciales basado en delegación y roles. Por último, el capítulo concluye con la presentación de la metodología que permitirá afrontar la puesta en marcha de un sistema de control de acceso de forma estructurada y haciendo uso de la infraestructura de autorización.

5.1 Visión general del sistema

A la hora de extender la infraestructura de clave pública presentada en el capítulo 3 con el fin de incorporar mecanismos de autorización, era necesario identificar qué elementos compondrían dicha extensión y cuáles serían los nexos con el sistema de partida.

Desde el punto de vista de su funcionalidad, la PKI resulta el mecanismo ideal para la generación y distribución de claves criptográficas entre los usuarios del sistema. La emisión controlada de certificados de identidad y la difusión de los mismos a través de tarjetas inteligentes nos sitúa en el punto de partida a la hora de iniciar la tarea de asignar privilegios a las claves contenidas en dichos certificados. De esta forma, se puede decir que el proceso de gestión de las claves (y de la identidad) se mantiene independiente del proceso de gestión de las autorizaciones, puesto que todas las cuestiones relacionadas con la validez o revocación de claves forman parte de la responsabilidad de la PKI, lo cual permite definir sistemas de gestión de autorizaciones totalmente enfocados a las cuestiones de manejo de privilegios.

El hecho de partir de una infraestructura de gestión de identidades tiene como contrapartida la presencia de identificadores únicos, al menos en el ámbito organizativo en el cual está definido del sistema. Dichos identificadores contenidos en los certificados de identidad X.509 pueden ser un problema en algunos escenarios de control de acceso en los cuales el anonimato resulta un requisito forzoso. En consecuencia, el sistema de gestión de autorizaciones debe proporcionar los mecanismos necesarios para eliminar el enlace que existe entre las claves públicas de los usuarios y su correspondiente identificador único. Es decir, para llegar a un sistema de control de acceso anónimo a partir de una infraestructura de certificación de identidad será necesario desarrollar propuestas que oculten dicha transición.

La figura 5.1 muestra cuál es la conexión entre la infraestructura de clave pública y la infraestructura de autorización basada en certificados que se presenta en este capítulo.

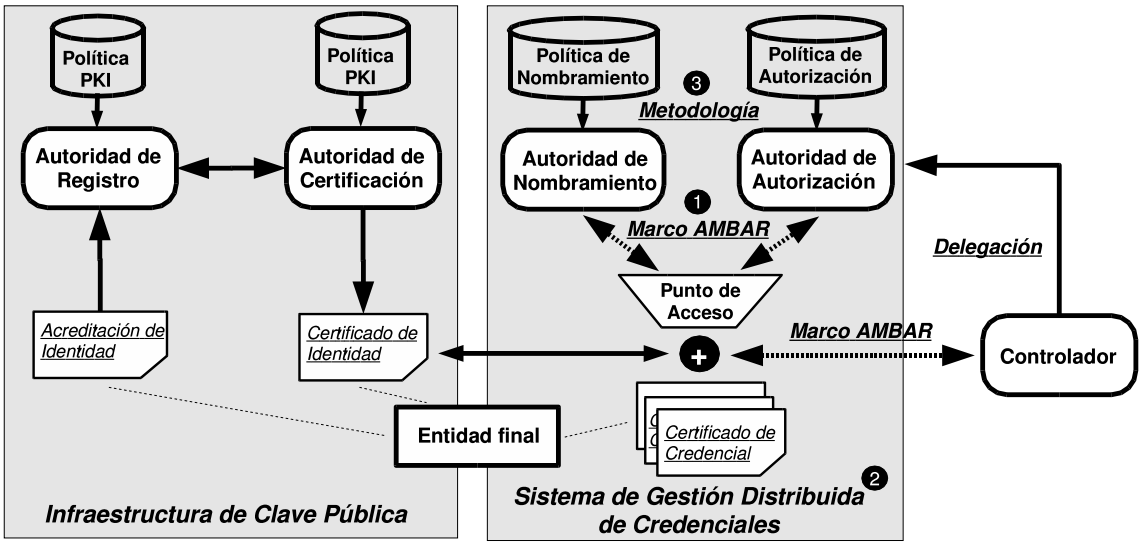

Figura 5.1: Visión general del sistema

Si se comparan las características de una PKI y un sistema de este tipo, es posible encontrar varias similitudes en lo que a estructura y funcionalidad se refiere.

En primer lugar, ambas necesitan una infraestructura formada por entidades emisoras, entidades intermedias (o mediadoras) y usuarios finales. En el caso de la PKI vimos como las autoridades de registro y las autoridades de certificación cooperan para tramitar las solicitudes de certificación presentadas por las entidades finales. En el caso de la infraestructura de autorización, es necesaria la presencia de elementos encargados de emitir los distintos tipos de credencial (autoridades de nombramiento y de autorización), así como la intervención de elementos intermedios capaces de poner en contacto a dichas autoridades con las entidades finales (puntos de acceso). Además, parte de ambas infraestructuras deben ser las especificaciones relacionadas con el formato de las solicitudes de certificación y formato de las políticas de seguridad, así como los medios utilizados para poner en contacto a las distintas entidades que la componen.

Por otro lado, en ambos casos las entidades emisoras deben seguir políticas concretas de certificación a la hora de atender las solicitudes presentadas por los usuarios finales. Vimos que las políticas de PKI permiten determinar si una solicitud o certificado cumplen con lo especificado por las prácticas de certificación. En el caso de la infraestructura de autorización, el uso de políticas está encuadrado en dos entornos distintos. En primer lugar, las políticas de nombramiento y de autorización permitirán determinar si una entidad concreta puede ser asociada a un conjunto de roles o de privilegios. En segundo lugar, el uso de políticas permitirá a los puntos de acceso tener un conocimiento de la estructura del sistema y de los requisitos de seguridad impuestos en el proceso de solicitud de credenciales. Dada la complejidad de ambos tipos de políticas, éstas deberán ser diseñadas siguiendo una metodología concreta que permita manejar de forma estructurada el gran número de usuarios y condiciones del sistema.

Finalmente, la última similitud está relacionada con la necesidad de mecanismos genéricos para el intercambio de los certificados generados, es decir, propuestas que permitan a las entidades participantes de una comunicación transmitir los certificados necesarios para el servicio que se está desarrollando. En el caso concreto de una PKI, dichos mecanismos los constituyen los distintos protocolos de seguridad con soporte para certificados X.509. Respecto a la autorización, es necesario un marco que permita intercambiar información relativa a autorización entre los controladores y las entidades finales.

Todas estas características propias de la infraestructura de autorización se agrupan en los tres bloques básicos presentados en este capítulo, los cuales aparecen numerados en la figura 5.1.

- *Marco AMBAR.* El marco AMBAR (Access Management Based on Authorization Reduction) es el mecanismo mediante el cual se transmite toda la información relacionada con autorización. Por un lado, el marco se emplea en las comunicaciones realizadas entre las entidades finales y los controladores de recursos con el fin de proporcionar un medio mediante el cual intercambiar certificados de identidad, certificados de credencial, políticas de autorización y recursos protegidos. Por otra parte, el marco forma también parte del sistema de gestión distribuida de credenciales ya que se emplea también para transmitir las solicitudes de certificación realizadas a las autoridades por parte de los usuarios finales. Los detalles de AMBAR se expondrán en la sección 5.2.

- *Sistema de gestión distribuida de credenciales.* Este sistema abarca tanto la definición de las entidades necesarias para la gestión de credenciales como la especificación de los elementos de información necesarios para dicho propósito. Está basado completamente en el mecanismo de delegación y en el concepto de rol, lo cual determina la mayor parte de sus características en lo que a estructura y notación se refiere. Tal y como se verá en la sección 5.3, el sistema realiza una distinción clara entre la gestión de la pertenencia a roles y la asignación de privilegios a dichos roles. Esta separación de conceptos puede apreciarse en la figura 5.1, donde el mecanismo de nombramiento (o pertenencia) dispone de sus propias autoridades y políticas independientes de la

autorización. Sin embargo, el acceso a la funcionalidad ofrecida por ambos subsistemas está agrupado en ciertos elementos mediadores denominados puntos de acceso, a través de los cuales es posible solicitar y obtener los certificados de credencial.

- *Metodología de definición de estructuras de gestión.* La puesta en marcha de un sistema de control de acceso basado en roles y delegación requiere una identificación muy concisa de los elementos participantes y de la relación entre ellos. Se trata de identificar todos los recursos que se desea proteger, determinar qué acciones realizadas sobre ellos deben controlarse, descubrir cuáles son los roles fundamentales del sistema, la política de pertenencia a dichos roles, el conjunto de privilegios asociados a los mismos, identificar a las entidades encargadas de emitir los certificados correspondientes y acotar los periodos de validez de los mismos, entre otras tareas. Debido al gran número de elementos involucrados y a la complejidad de las tareas asociadas, es necesario establecer una metodología genérica de construcción de políticas de autorización y de nombramiento, a partir de las cuales pueda abordarse el desarrollo del sistema de una forma estructurada. Dicha metodología de diseño se analizará en la sección 5.4.

Como se verá a lo largo de este capítulo, los distintos componentes se encuentran totalmente relacionados entre sí y, a su vez, con la infraestructura de clave pública ya descrita.

5.2 AMBAR: marco de intercambio de información relativa a autorización

En la sección 4.4.4 se analizaron las distintas alternativas posibles a la hora de obtener o descubrir las credenciales necesarias para tomar las decisiones de autorización. Como ya se comentó, los sistemas de control de acceso pueden emplear enfoques muy distintos en lo que a distribución de credenciales se refiere. Algunos de ellos determinan que la responsabilidad de obtener la información es del controlador de recursos, mientras que otros argumentan que deben ser los solicitantes los encargados de proporcionar la información relativa a sus privilegios. En general, no hay acuerdo acerca de cuál es la mejor alternativa ya que en la mayoría de los casos depende del entorno de aplicación concreto.

Por otro lado, también se ha comentado la falta de mecanismos genéricos relacionados con el intercambio de información de autorización. Los protocolos de seguridad más comunes, como TLS (Transport Level Security) [59], IKE (Internet Key Exchange) [94], o S/MIME (Secure/Multipurpose Internet Mail Extensions) [168], están preparados para transmitir única y exclusivamente certificados de identidad.

Como consecuencia, uno de los campos de trabajo a los que más esfuerzo se le ha dedicado en esta tesis es la definición de un marco que proporcione los mecanismos necesarios para controlar el acceso a recursos protegidos en escenarios basados en el modelo

cliente-servidor. Este marco, denominado *AMBAR (Access Management Based on Authorization Reduction)*, se ha diseñado siguiendo un enfoque estructurado mediante el cual se han identificado los distintos parámetros relacionados con el intercambio de información relativa a autorización, todo ello con el fin de adaptar el marco a los distintos enfoques ya comentados acerca de distribución de certificados de credencial.

En primer lugar se analizarán las limitaciones de los sistemas existentes y se contrastará la propuesta con otras iniciativas relacionadas. A continuación se enumerarán los requisitos del marco y se describirá su arquitectura. Posteriormente se detallará una implementación concreta del marco basada en un protocolo cliente-servidor denominado protocolo AMBAR. Por último se realizará un análisis de seguridad de dicha implementación del marco.

5.2.1 Análisis de las propuestas actuales

En esta sección se analizará cómo se lleva a cabo normalmente el control de acceso basado en certificados. Este análisis mostrará por qué las propuestas actuales pueden verse mejoradas mediante la utilización del marco que aquí se presenta. Se ha seleccionado un escenario basado en Web, donde el controlador de recursos toma decisiones en función de la información de autorización presentada por los clientes y de su propia política de control de acceso. Se supondrá que dicho controlador delega en autoridades externas el privilegio de determinar qué entidades están autorizadas a acceder a los recursos, determinación que realizarán éstas mediante la emisión de certificados de credencial. Así pues, el acceso será concedido siempre que el controlador disponga de toda la información necesaria para verificar que la solicitud cumple con su política de seguridad.

Uno de los enfoques más tradicionales que se pueden seguir a la hora de implementar este sistema es el mostrado en la figura 5.2. En ella podemos observar como tanto el controlador como el cliente disponen de módulos adicionales encargados de las funciones de control de acceso. Esta funcionalidad puede ser añadida al software del cliente mediante la utilización de *applets* o *ActiveX*. El servidor Web en el cual se encuentra ubicado el controlador puede realizar dicha función mediante el uso de *servlets* o extensiones de servidor específicas. Cuando un usuario solicita el acceso a un recurso, se establece una conexión SSL [9] con el fin de autenticar a los participantes de dicha comunicación y de proteger los datos que se enviarán a continuación. Acto seguido, se genera un mensaje HTTP [77] que especifica el recurso solicitado. Los certificados de credencial deben incluirse en el documento HTML o en alguna cabecera HTTP ya que SSL no proporciona mecanismos para intercambiar este tipo de información. Por tanto, solicitud y credenciales son encapsuladas en el mismo paquete SSL y enviadas al controlador. El paquete es procesado por el módulo SSL y parte de su contenido es entregado al software de control de acceso con el fin de determinar si la solicitud debe ser aprobada. Finalmente, el recurso se entrega al usuario y, opcionalmente, se adjunta un conjunto de certificados destinados a proporcionar una autorización directa que simplifique solicitudes posteriores.

Esta solución puede tener asociados varios inconvenientes. En primer lugar, las credenciales deben incluirse como parte de los datos de la aplicación (HTML). Con el fin de independizar el mecanismo de control de acceso del entorno de aplicación concreto, sería

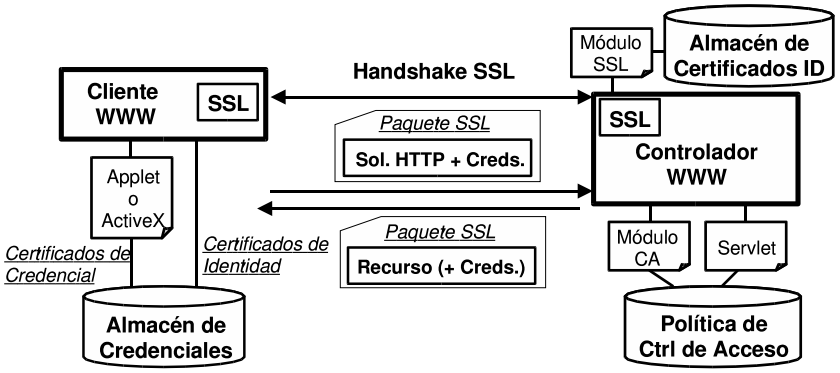

Figura 5.2: Enfoque común de control de acceso basado en certificados

conveniente no mezclar los datos de aplicación con la información relativa a autorización. De hecho, los certificados de identidad intercambiados durante la fase inicial de autenticación SSL se transmiten siguiendo este enfoque. Una forma de no combinar ambos tipos de datos es transmitir en primer lugar la solicitud, después obtener del controlador la política de control de acceso asociada y finalmente enviar los certificados de credencial. El problema está en que el controlador podría considerar que la política de control de acceso contiene información confidencial, la cual no debería ser difundida a usuarios desconocidos. Las credenciales transmitidas junto con la solicitud siguiendo el enfoque original pueden ayudar al servidor a determinar si puede desvelar su política.

Por otro lado, esta propuesta carece de una fase de negociación de los parámetros de autorización. Como se vio en la sección 4.3, son varias las especificaciones realizadas en materia de certificados de credencial, y varios los métodos de distribución de los mismos. Los participantes deben poder seleccionar si las credenciales serán proporcionadas por parte del cliente (método denominado *push*) o si bien serán recuperadas por parte del controlador de algún suministrador (método *pull*). Además, el método *push* puede subdividirse a su vez en función de si se realiza una difusión controlada de la política de control de acceso. Mediante la propuesta presentada en la figura 5.2 es difícil realizar una negociación de dichos parámetros que controle todas las solicitudes realizadas por parte del cliente dentro de la misma sesión [91].

Finalmente, dejar constancia de la dificultad existente con este enfoque a la hora de optimizar solicitudes subsecuentes. Es común que colecciones de recursos organizadas por directorios hereden los derechos de acceso conforme nos adentramos en el árbol de documentos. Por tanto, una vez que un usuario ha sido autorizado a acceder a un recurso en la misma sesión no resulta necesario retransmitir las credenciales implicadas en dicha decisión. Sin embargo, la implementación de sesiones, y por tanto la posibilidad de realizar optimizaciones dentro de la misma sesión, es un proceso que debería implementar la propia aplicación del controlador, lo cual complica su diseño.

Podemos encontrar en la literatura algunos sistemas de control de acceso que siguen este enfoque. En [138] se presenta un mecanismo de control de acceso a recursos Web basado en certificados SPKI y en el uso del protocolo HTTP como mecanismo de trans-

porte de información de autorización. En consonancia con lo que se acaba de comentar, dicho sistema carece de fase de negociación, mantenimiento de sesiones u optimización de solicitudes.

Enfoques alternativos

Conscientes de las limitaciones del esquema presentado en la figura 5.2, son varios los autores que han propuesto sistemas alternativos que intentan suplir algunas de las carencias anteriormente comentadas.

En [184] se presenta un mecanismo que emplea certificados digitales para definir y aplicar políticas de control de acceso sobre recursos ampliamente distribuidos. La arquitectura está basada en el modelo *pull*, donde el controlador se encarga de recuperar los certificados asociados a los usuarios con el fin de determinar si se cumplen las condiciones especificadas por los proveedores de los recursos a controlar. El sistema proporciona algunos mecanismos para optimizar solicitudes subsecuentes, como por ejemplo el uso de caches de certificados. Como se verá más adelante, el marco aquí presentado complementa esta propuesta ya que también es capaz de dar soporte a sistemas basados en delegación, los cuales suelen estar basados en el método de distribución *push*.

Otros trabajos están relacionados con el control de la difusión de políticas [179]. Dicha propuesta muestra cómo es posible diseñar sistemas de establecimiento automático de confianza (*automated trust establishment*) que controlen la revelación de información confidencial contenida en las políticas de control de acceso. Como se verá en la sección 5.2.3, este mecanismo puede incorporarse a uno de los módulos del marco AMBAR con el fin de controlar dicha difusión de las políticas.

Un enfoque que hace uso del protocolo SSL es el presentado en [97]. Esta propuesta hace uso de los certificados X.509 intercambiados durante la fase de negociación SSL para asignar a los usuarios solicitantes un rol dentro del sistema. Dicha asignación se realiza siguiendo una política de pertenencia a roles definida por el controlador de los recursos. El sistema proporciona sólo una solución parcial al problema ya que no determina los mecanismos mediante los cuales se asignan los privilegios a los roles, sino que simplemente proporciona una solución destinada al agrupamiento de usuarios en roles. A pesar de lo que se vio en la sección 4.3, los propios autores del sistema no consideran útiles los certificados de credencial a la hora de especificar la pertenencia a grupos ni de especificar los permisos asociados a los mismos.

El grupo de trabajo AAA (Authentication, Authorization and Accounting) del IETF ha propuesto también un marco de autorización [146, 186] destinado a la protección de recursos y servicios dentro del ámbito de Internet. El marco está basado principalmente en el control de acceso a la red, movilidad y calidad de servicio en IPv6. La principal diferencia entre esta propuesta y el marco AMBAR es que este último está más enfocado a aplicaciones y escenarios de alto nivel.

5.2.2 Objetivos generales del marco

El marco que aquí se presenta soluciona gran parte de los inconvenientes que se han mencionado en el apartado anterior. AMBAR da soporte a distintos tipos de certificados de credencial y de identidad, e incorpora un mecanismo de negociación diseñado para adaptar el marco a escenarios de control de acceso con distintas características. Tal y como se verá en apartados posteriores, AMBAR es un marco basado en sesiones que está constituido por varios módulos independientes. La figura 5.3 muestra cómo puede adaptarse dicho marco a escenarios basados en el Web.

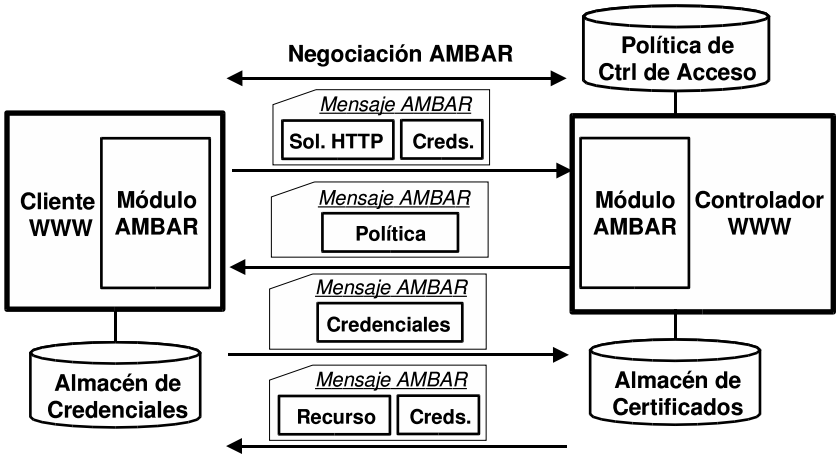

Figura 5.3: Control de acceso basado en AMBAR

Tanto el controlador como el cliente disponen de módulos adicionales AMBAR responsables de las funciones de control de acceso. Cuando un usuario solicita un recurso protegido, se inicia una fase de negociación de sesión con el fin de autenticar a las entidades participantes, negociar qué tipo de credenciales será utilizado, acordar cuál será el método de distribución de las mismas y determinar si la información intercambiada debe ser protegida de posibles ataques tanto pasivos como activos. Una vez que se establece la sesión, las solicitudes HTTP y las credenciales se encapsulan de forma separada dentro de paquetes AMBAR y se envían al controlador. El contenido de dichos paquetes puede ser procesado por un módulo del marco AMBAR o bien ser entregado a otra aplicación. En este ejemplo, el contenido es procesado por el módulo AMBAR, el cual determina la política de autorización relacionada con la solicitud (la selección de políticas puede estar basada en diversos métodos y suele ser dependiente de las credenciales contenidas en el primer mensaje). A continuación, el controlador transmite un paquete que incluye la política de autorización que especifica las credenciales necesarias para obtener el acceso. Finalmente, el cliente transmite dichas credenciales y el controlador suministra el recurso solicitado.

Hay varias ventajas en el hecho de utilizar un enfoque de este tipo. La primera es que las entidades pueden negociar los parámetros relacionados con el control de acceso. Además, las credenciales y los datos de aplicación se envían de forma separada, y es bastante sencillo intercambiar varios mensajes (solicitud, credenciales, políticas) para resolver una solicitud

de acceso. Al estar basado en un enfoque orientado a la sesión, es posible relacionar solicitudes entre sí con el fin de optimizar los cálculos o los envíos necesarios para tomar las decisiones.

En vista de lo analizado, hay tres objetivos principales que deben ser satisfechos por el marco AMBAR. En primer lugar, debe ser independiente del entorno de aplicación, es decir, debe dar soporte a cualquier tipo de política, privilegio o solicitud. En segundo lugar, debe ser capaz de operar con las principales especificaciones en materia de certificados de identidad y de credencial, así como de negociar los parámetros de autorización de cada sesión. Por último, su diseño debe ser extensible, estructurado y estar dividido en módulos con funciones bien definidas.

Con el fin de optimizar aquellos escenarios en los que varios mensajes de solicitud y respuesta se intercambian continuamente entre un cliente y un controlador, debe tratarse de un marco orientado a la sesión. En estos casos, la mayor parte de la información necesaria ya fue enviada con solicitudes anteriores, y algunas de las decisiones de autorización ya obtenidas pueden ser útiles para determinar si una nueva solicitud debe ser aprobada, sin la necesidad de calcular nada de nuevo. Como se verá en posteriores apartados, las caches de certificados y las reducciones de autorización son mecanismos muy indicados para estos propósitos.

5.2.3 Arquitectura del marco

El marco AMBAR está compuesto por diferentes módulos organizados, tal y como muestra la figura 5.4, en dos capas. La capa superior está formada por cinco módulos funcionales distintos: Gestión de Sesiones *(SM, Session Management)*, Gestión de Solicitudes *(RM, Request Management)*, Gestión de Resultados de Autorización *(ARM, Authorization Results Management)*, Gestión de Flujos de Datos *(DSM, Data Stream Management)*, y Gestión de Errores *(EM, Error Management)*. En el nivel inferior se sitúa la capa de Convergencia de Transporte *(TC, Transport Convergence)*. La capa TC encapsula toda la información generada por los módulos superiores de acuerdo con el mecanismo de transporte correspondiente. Opcionalmente, esta capa protege la confidencialidad y la integridad de la información mediante la aplicación de alguno de los mecanismos negociados durante una fase previa. Los siguientes apartados detallan la funcionalidad de cada módulo.

Figura 5.4: Arquitectura AMBAR

Session Management

Este módulo proporciona los mecanismos para negociar las diferentes opciones soportadas por el marco y para establecer los parámetros de la sesión. El módulo genera además todo el material criptográfico que pudiera ser necesario para la capa TC a la hora de proteger la información intercambiada.

Los solicitantes y los controladores de recursos pueden negociar los siguientes parámetros:

- *Cifrador simétrico.* Los participantes pueden seleccionar qué algoritmo de cifrado simétrico (y su longitud de clave) protegerá los datos intercambiados.
- *Modo de operación.* AMBAR proporciona dos modos de operación: modo anónimo (la identidad del solicitante no se revela) y modo identificado (donde tanto el solicitante como el controlador son identificados).
- *Certificados de identidad.* Los participantes pueden seleccionar qué tipo de certificados de identidad serán empleados para propósitos de autenticación (X.509, PGP, SPKI/SDSI, etc).
- *Certificados de credencial.* Los participantes pueden seleccionar qué tipo de certificados serán utilizados para propósitos de autorización (SPKI/SDSI, X.509 AC, KeyNote, etc).
- *Método de distribución.* Es posible negociar si las credenciales serán proporcionadas por parte del solicitante (*push*) o si bien serán obtenidas por parte del controlador desde algún suministrador. El método *push* puede a su vez subdividirse en varias posibilidades dependiendo del criterio seguido para la revelación de la política de autorización y de la entidad responsable de hallar la prueba de autorización.

Todos estos parámetros se negocian en función de las políticas de seguridad específicas definidas por los sistemas finales. El objetivo principal del módulo SM es adaptar el marco a los distintos escenarios de control de acceso y crear sesiones AMBAR.

Request Management

Las decisiones de autorización, procesos de optimización o los algoritmos de control de difusión de políticas son ejecutados como parte del módulo RM. En general, este módulo está encargado de la gestión de todo lo relacionado con solicitudes, credenciales y políticas.

Las solicitudes pueden ser generadas por la aplicación del solicitante o pueden ser derivadas a partir de datos específicos de la aplicación dentro de este módulo. Por ejemplo, una solicitud HTTP puede ser convertida de forma automática a una s-expresión con el fin de simplificar el proceso de autorización, aunque tanto la solicitud HTTP como la s-expresión serían transmitidas al controlador. Las s-expresiones son especialmente útiles porque reflejan sólo aquella información relacionada con el proceso de decisión.

Las credenciales pueden ser recuperadas a partir de almacenes de certificados, de entidades emisoras o de los propios solicitantes. Dichas credenciales deben ser verificadas y validadas, aunque los mecanismos para dichos propósitos son completamente dependientes de la infraestructura disponible ya que, como se ha visto en capítulos anteriores, dichas comprobaciones pueden estar basadas en listas de certificados revocados, mecanismos de verificación en línea, certificados de corta duración, etc. Una consecuencia lógica de todo esto es que las carencias de la infraestructura en la cual se esté aplicando el marco pueden afectar a éste en lo que a seguridad se refiere.

Las políticas son emitidas con el fin de especificar qué credenciales son necesarias para obtener el acceso a los recursos que se están solicitando. El criterio de revelación de dichas políticas depende del controlador en cuestión. En general, hay tres alternativas posibles a la hora de efectuar dicha revelación: los controladores pueden difundir gradualmente las políticas [179]; pueden difundirlas sin ningún tipo de control; o pueden decidir no desvelarlas (más característico del modo *pull*).

Como ya se ha mencionado, un protocolo orientado a la sesión permite realizar algunas optimizaciones. El módulo RM es responsable del cálculo y las optimizaciones de las decisiones de control de acceso. Las optimizaciones están condicionadas por varios parámetros, como el tipo de certificado de credencial que se esté empleando y el método de distribución que se haya negociado.

Una de las formas más sencillas de minimizar el número de envíos es guardar una copia local de los certificados que han sido intercambiados. No obstante, hay que tener en cuenta que la validez de los certificados almacenados de forma local debe ser comprobada de forma periódica, sobre todo teniendo en cuenta que posibles revocaciones pueden alterar su estado.

Otro mecanismo que se emplea para reducir el ancho de banda utilizado es la transmisión de los resúmenes digitales de los elementos de información que ya han sido transmitidos previamente, lo cual es especialmente útil a la hora de retransmitir políticas de autorización extensas. Para ello, el módulo RM puede hacer uso de una tabla de dispersión que relacione los resúmenes digitales recibidos con los elementos previamente transmitidos.

Por otro lado, con el fin de simplificar el cálculo de autorización, es posible hacer uso de decisiones previas para determinar si una nueva solicitud de acceso debe ser aprobada o denegada. La figura 5.5 muestra un ejemplo de esta situación. Una vez que la solicitud de acceso al fichero C ha sido procesada, R no necesita presentar ninguna credencial nueva para acceder a los ficheros A o E ya que la reducción de autorización llevada a cabo por el controlador indica que tiene concedido el acceso a A, C y E. Esta reducción podría incluso ser utilizada durante otras sesiones para evitar la transmisión de credenciales incluidas en dicha reducción. El mecanismo de reducción empleado para realizar estas optimizaciones dentro del módulo será explicado detalladamente en la sección 5.3.5.

Authorization Results Management

El módulo ARM proporcionar los mecanismos necesarios para generar las notificaciones acerca de las decisiones de control de autorización, y puede usarse además para gestionar los

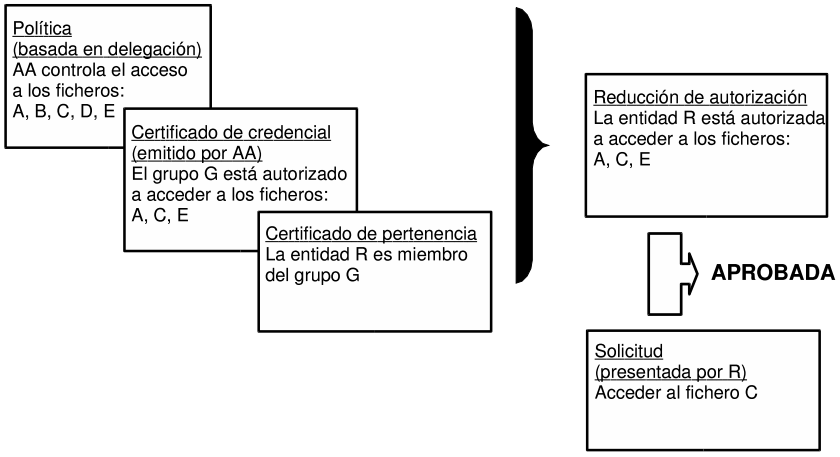

Figura 5.5: Ejemplo de optimización de solicitud de acceso

recursos solicitados. Se generarán notificaciones negativas cuando el acceso sea denegado. En el caso de que sea aprobado, dos son las posibles respuestas: una notificación positiva si el solicitante no desea obtener el recurso, sino sólo llevar a cabo alguna acción sobre él; o bien el recurso en sí. El módulo además habilita o deshabilita el módulo DSM siempre que se apruebe una solicitud relacionada con el inicio o la finalización de un flujo de datos.

Error Management

Las aplicaciones hacen uso del módulo EM para notificar situaciones de error o de alerta, como por ejemplo la verificación incorrecta de un mensaje, la recepción de un mensaje fuera de orden, la imposibilidad de negociación debida a preferencias incompatibles, la validación negativa de un certificado, etc. La información difundida acerca de estas situaciones contienen un nivel de gravedad y una descripción del error que se ha producido.

Data Stream Management

El modelo descrito basado en intercambios de solicitudes y respuestas no es apropiado si se desea utilizar el marco AMBAR para controlar comunicaciones bidireccionales basadas en flujos de datos. El módulo DSM, inicialmente deshabilitado, controla la transmisión de dichos flujos tras la aprobación de una solicitud de activación del mismo. Una vez que éste es activado, las aplicaciones son capaces de intercambiar datos libremente, que además puede ser protegidos por la capa TC si así se negoció durante la sesión (la sección 5.2.4 se detalla este proceso).

Transport Convergence

La capa TC codifica toda la información generada por los módulos superiores. Dicha codificación puede estar basada en XDR [143], XML o cualquier otro formato que resulte

apropiado. Además, la implementación del módulo TC depende del mecanismo de transporte concreto que se esté empleando para transmitir la información, el cual puede ser una conexión SSL, un socket TCP, etc. En el siguiente apartado se presenta una implementación concreta del marco AMBAR que hace uso de sockets TCP. En dicha implementación la capa TC ofrece servicios de confidencialidad y autenticación de los mensajes intercambiados.

5.2.4 El protocolo AMBAR como implementación del marco

Una vez que se ha analizado el diseño del marco AMBAR, en esta sección se expondrá cómo puede intercambiarse información relativa a autorización mediante el protocolo AMBAR [47]. El marco puede considerarse como las directrices a seguir a la hora de implementar un sistema de control de acceso concreto.

Parte del diseño del protocolo está basado en SSL. SSL es sin duda una aportación muy valiosa al campo de las comunicaciones confidenciales y su seguridad ha sido ampliamente estudiada durante los últimos años [145, 187]. AMBAR toma como punto de partida una modificación de la funcionalidad ofrecida por SSL y por tanto algunos mensajes se han visto simplificados o han sido eliminados. Dicha modificación se han realizado siguiendo algunas de las prácticas de diseño de protocolos criptográficos expuestas en [2, 13], las cuales hacen referencia al uso correcto de cargas aleatorias, la inclusión de información que pueda ser asociada a los participantes o la codificación de un mensaje dentro de una secuencia, entre otros factores. Es importante recalcar que el objetivo principal de esta implementación del marco no es la definición de un nuevo protocolo criptográfico, sino la especificación de un protocolo que cumpla con los requisitos expuestos en la sección 5.2.2.

La descripción del protocolo está estructurada atendiendo a los distintos módulos del marco, poniendo un énfasis especial en lo que a negociación de sesiones y gestión de solicitudes se refiere. Una especificación completa de los mensajes que componen el protocolo se encuentra en el apéndice B.

Notación empleada

La especificación de los mensajes está formada por 4 campos distintos. El primero de ellos hace referencia al orden del mensaje dentro de la fase en la cual se encuentra encuadrado. El segundo indica el nombre del mensaje. El tercer campo muestra si el mensaje es enviado desde el cliente al servidor o viceversa. Por último, el cuarto campo especifica los contenidos del mensaje.

Dichos contenidos se encuentran también expresados siguiendo una notación concreta. A continuación se detalla como debe interpretarse dicha notación:

- *item1+item2+item3*: Concatenación de varios elementos de información.
- k_X: Clave pública de X.

- $\{M\}_{k_X^{-1}}$: Mensaje M cifrado con la clave privada de X (en ocasiones equivalente a la firma digital de M).
- *SHA1(M) o MD5(M)*: Resumen digital de M calculado mediante las funciones SHA-1 [42] o MD5 [171].
- $\{M\}^{k_{MAC}}$: Código de autenticación de M calculado mediante la clave k_{MAC}.
- $\{M\}_{k_S}$: Mensaje M cifrado con la clave k_S.

Módulo SM

Los mensajes del módulo SM se usan para negociar las diferentes opciones soportadas por el marco y para establecer los parámetros de la sesión. Parte de los datos intercambiados se emplean para generar el material criptográfico que utiliza la capa TC para proteger la información (siempre que la opción de confidencialidad se haya negociado).

Cada mensaje SM contiene un campo que define el tipo concreto de mensaje. Los siguientes apartados detallan los mensajes que se intercambian durante la negociación tanto de una sesión identificada como de una sesión anónima.

Sesión identificada

1 **ClientInit** $C \Rightarrow S$ $Ver_c, N_c, Assert_c, Category_c, Suite_c, Identity_c, Distribution_c$

El mensaje *ClientInit* inicia la negociación AMBAR y contiene las preferencias del cliente. Ver_c identifica la versión de AMBAR del cliente, N_c es una carga aleatoria de 64 bytes que será empleada posteriormente para calcular el material criptográfico, $Assert_c$ indica las preferencias del cliente en lo que a certificados de credencial se refiere, $Category_c$ expresa el modo de operación propuesto (anónimo o identificado), $Identity_c$ indica las preferencias en lo que respecta a certificados de identidad y $Distribution_c$ contiene el modo de distribución de credenciales propuesto. $Suite_c$ es un campo que contiene una lista de los algoritmos de cifrado simétrico que soporta el cliente a la hora de proteger los datos de la capa TC. Cuando dicha lista contiene sólo el elemento *null* la capa TC no ofrece ningún tipo de servicio de protección de la información transmitida, y por tanto la fase SM se reduce al intercambio de los mensajes *ClientInit* y *ServerInit*.

2 **ServerInit** $S \Rightarrow C$ $Ver_s, N_s, Assert_s, SessionID, Category_s, Suite_s, Identity_s, Distribution_s$

El mensaje *ServerInit* es la respuesta del servidor a *ClientInit*. Las diferencias más significativas entre ambos mensajes son la presencia del identificador de sesión *SessionID* y la selección por parte del servidor de uno de los algoritmos simétricos propuestos. La elección de los parámetros por parte del servidor se realiza siempre teniendo en cuenta lo especificado por el cliente. Si alguna de las preferencias del cliente son incompatibles con las del servidor se produce el intercambio de mensajes EM para notificarlo.

3 **PKValue** $S \Rightarrow C$ $\{S, k_s\}_{k_{CA_1}^{-1}}$

El mensaje *PKValue* enviado por el servidor contiene información relativa a su identidad, y está compuesto normalmente por un certificado digital que incluye datos acerca de la clave pública del servidor k_s, su identificador asociado S y la entidad emisora CA_1.

4 **PKValue** $C \Rightarrow S$ $\{C, k_c\}_{k_{CA_2}^{-1}}$

Cuando el mensaje *PKValue* lo envía el cliente éste contiene un certificado digital con la identidad del cliente C, su clave pública k_c, y su entidad emisora CA_2 (CA_1 y CA_2 podrían hacer referencia a la misma autoridad, pero no es obligatorio).

5 **ActivateCrypto** $C \Rightarrow S$ $\{PreMasterSecret\}_{k_s}, \{SHA1(N_c + MasterSecret + N_s)\}_{k_c^{-1}}$

El mensaje *ActivateCrypto* se emplea para establecer el material criptográfico que protegerá los siguientes mensajes del protocolo y para verificar la identidad del cliente. En primer lugar, está compuesto de un valor de 64 bytes, denominado $PreMasterSecret$, que se transmite cifrado mediante la clave pública del servidor obtenida con el mensaje *PKValue*. Este $PreMasterSecret$ y las cargas aleatorias intercambiadas con los primeros dos mensajes dan lugar al $MasterSecret$. En segundo lugar, el mensaje contiene una cadena de bytes que representan la firma digital de la concatenación del $MasterSecret$ a dichas cargas aleatorias. Una vez que el servidor recibe el mensaje, éste descifra el $PreMasterSecret$, calcula a partir de él el valor del $MasterSecret$ y verifica la firma del cliente. De esta forma, el servidor puede averiguar si el cliente controla la clave privada asociada a la clave pública que fue transmitida en el cuarto mensaje y si ambos participantes han llegado al mismo $MasterSecret$. El cálculo del $MasterSecret$ se muestra en el apéndice B.

6 **InitSession** $C \Rightarrow S$ $\{SHA(MasterSecret + Issuer + SM\_Messages)\}_{k_{SYMM_s}}^{k_{MAC}}$

7 **InitSession** $S \Rightarrow C$ $\{SHA(MasterSecret + Issuer + SM\_Messages)\}_{k_{SYMM_c}}^{k_{MAC}}$

El último mensaje de la fase SM es *InitSession*. Sirve para indicar que la fase de negociación ha concluido y que se ha activado la protección criptográfica de los mensajes. Se trata del primer mensaje AMBAR protegido por la capa TC mediante los datos derivados a partir del $MasterSecret$. Su contenido está formado principalmente por el resumen digital de todos los mensajes intercambiados durante la fase de negociación (el contenido de este mensaje se analizará más en detalle en la sección 5.2.5).

Sesión anónima

Sólo hay una diferencia entre el modo identificado y el anónimo. Con el fin de preservar la identidad del cliente, el mensaje *PKValue* contiene en este caso sólo la clave pública del mismo.

4 **PKValue** $C \Rightarrow S$ k_c

Módulo TC

El módulo TC transforma los mensajes de la capa superior de acuerdo con lo negociado en la fase anterior. Los mensajes transmitidos con anterioridad a *InitSession* no están protegidos ya que las claves criptográficas se calculan a partir del $MasterSecret$.

El módulo proporciona además un formato común de encapsulamiento de los mensajes SM, RM, ARM, DSM y EM. En esta implementación se ha empleado la siguiente estructura para codificarlos:

AMBARMessage $C \Leftrightarrow S$ $tipo, longitud, datos$

El campo de *datos* contiene los mensajes, posiblemente protegidos, de la capa superior, cuyo tipo es *tipo* y de tamaño igual a *longitud*. Los mensajes protegidos están compuestos por dos campos: *contenido* y *MAC*. El *contenido*, el *tipo* y la *longitud* se autentican primero utilizando un algoritmo HMAC [42], y el código resultante se almacena en *MAC*. A continuación, tanto *contenido* como *MAC* se cifran utilizando el modo de cifrado CBC (Cipher Block Chaining) [122] y el sistema de relleno PCKS#5 [119]. Las claves empleadas para calcular los códigos de autenticación se denominan K_{MAC} y las claves de cifrado utilizadas son K_{SYMM_S} y K_{SYMM_C}. La forma de derivar estas claves a partir del $MasterSecret$ se muestra en el apéndice B.

Módulos RM, ARM y DSM

Con el fin de explicar el funcionamiento de estos módulos, se analizarán los distintos métodos de distribución a través de algunas secuencias típicas de mensajes. A lo largo de este apartado, se denominará *transacción* a los diferentes mensajes relacionados con una solicitud de acceso concreta, mientras que por *sesión* se entiende la secuencia de distintas transacciones.

Método de distribución push-calculation

En una sesión basada en el método *push-calculation* los clientes calculan la prueba de autorización tras la recepción de la política que protege los recursos gestionados por el controlador.

1	**Request**	$C \Rightarrow S$	$\{T_{ID}, T_{Step}, SFlag, Solicitud, [Asserts]^{0..N}\}^{k_{MAC}}_{k_{SYMM_s}}$
2	**Policy**	$S \Rightarrow C$	$\{T_{ID}, T_{Step}, ACL\}^{k_{MAC}}_{k_{SYMM_c}}$
3	**Calculation**	$C \Rightarrow S$	$\{T_{ID}, T_{Step}, Prueba\}^{k_{MAC}}_{k_{SYMM_s}}$
4	**Neg_Notification**	$S \Rightarrow C$	$\{T_{ID}, T_{Step}, Detalles\}^{k_{MAC}}_{k_{SYMM_c}}$
4	**Aff_Notification**	$S \Rightarrow C$	$\{T_{ID}, Detalles\}^{k_{MAC}}_{k_{SYMM_c}}$
4	**Resource**	$S \Rightarrow C$	$\{T_{ID}, Recurso\}^{k_{MAC}}_{k_{SYMM_c}}$

El mensaje *Request*, generado por el módulo RM, representa la solicitud de autorización formulada por el cliente. Contiene un identificador de transacción T_{ID}, un identificador

de secuencia dentro de la transacción T_{Step}, un valor $SFlag$ que indica si la solicitud está relacionada con la gestión de flujos de datos, un conjunto de credenciales relacionadas con la solicitud (las cuales pueden servir al controlador para decidir revelar su política) y la solicitud de acceso. Los datos están cifrados (si así se negoció) mediante la clave K_{SYMM_S} y autenticados con K_{MAC}. Todos los mensajes de esta sección están protegidos de la misma forma, por lo que no se volverá a hacer referencia a estas claves.

La respuesta del controlador, generada por el módulo RM, es el mensaje *Policy*. Incluye la lista de control de acceso que protege el recurso, el mismo identificados T_{ID} que aparecía en la solicitud y un valor T_{Step} incrementado en una unidad.

Una vez que el cliente recibe la política, se crea una prueba de autorización que contiene todos los certificados necesarios para formar una cadena de delegación desde la política hasta la clave del solicitante. Dicha prueba se envía al controlador como parte del mensaje *Calculation*. El uso de T_{ID} y T_{Step} es el ya comentado.

El último paso es la respuesta del servidor a la prueba. Si ésta fuera incompleta, el servidor mandaría un mensaje *Neg_ Notification*. Dicho mensaje podría contener los detalles (*Detalles*) de la negativa. Por otro lado, si la prueba fuera correcta, el controlador podría enviar un mensaje *Resource* (si se estaba solicitando acceder a un recurso concreto) o un mensaje *Aff_ Notification* (si la solicitud no lleva implícita la transmisión del recurso sino, por ejemplo, la ejecución remota de una operación sobre el recurso).

Método de distribución push-asserts

Cuando se selecciona el método de distribución *push-asserts* los controladores son responsables de todo el proceso de construcción de pruebas de autorización. Los solicitantes envían la solicitud y todas las credenciales necesarias para el acceso al recurso. Dicho envío de credenciales puede realizarse en demanda, es decir, en función de la información que éstos reciben mediante los mensajes *Policy* del controlador, o bien durante el envío de la solicitud en aquellos casos en los que el controlador no esté dispuesto a revelar ningún dato acerca de su política.

1	**Request**	$C \Rightarrow S$	$\{T_{ID}, T_{Step}, SFlag, Solicitud, [Asserts]^{0..N}\}^{k_{MAC}}_{k_{SYMM_s}}$
2	**Policy**	$S \Rightarrow C$	$\{T_{ID}, T_{Step}, ACL\}^{k_{MAC}}_{k_{SYMM_c}}$
3	**Asserts**	$C \Rightarrow S$	$\{T_{ID}, T_{Step}, [Asserts]^{1..N}\}^{k_{MAC}}_{k_{SYMM_s}}$
4	**Neg_Notification**	$S \Rightarrow C$	$\{T_{ID}, T_{Step}, Detalles\}^{k_{MAC}}_{k_{SYMM_c}}$
4	**Aff_Notification**	$S \Rightarrow C$	$\{T_{ID}, Detalles\}^{k_{MAC}}_{k_{SYMM_c}}$
4	**Resource**	$S \Rightarrow C$	$\{T_{ID}, Recurso\}^{k_{MAC}}_{k_{SYMM_c}}$

La principal diferencia con respecto al método anterior es el envío del mensaje *Asserts* por parte del solicitante tras la solicitud de la política del controlador. Los mensajes 2 y 3 pueden intercambiarse varias veces en función de la táctica de revelación de la política de control de acceso que siga el controlador. Por otro lado, también es posible que dichos mensajes no lleguen a intercambiarse en aquellos casos en los que el controlador no esté dispuesto a aportar ningún tipo de información. Esto implicaría que el solicitante debería

enviar todas las credenciales en el primer mensaje.

Método de distribución pull

En algunos casos, el controlador puede optar por recuperar las credenciales del solicitante a partir de un suministrador de información. Con este método de distribución *pull* el cliente sólo envía un único mensaje *Request* que contiene la solicitud de acceso.

1 **Request** $C \Rightarrow S \quad \{T_{ID}, T_{Step}, SFlag, Solicitud\}_{k_{SYMM_s}}^{k_{MAC}}$

Cuando el controlador recibe el mensaje intenta recuperar las credenciales necesarias para conceder el acceso. Los mensajes de respuesta ARM han sido omitidos por simplicidad.

Gestión de flujos de datos

Como se comentó anteriormente, el mensaje *Request* contiene un valor que indica si un flujo de datos debe ser establecido o cancelado. El valor *start_stream* representa la solicitud de un nuevo flujo de datos (cualquier posible flujo anterior sería cancelado), el valor *stop_stream* se usa para solicitar la finalización del flujo actual y el valor *no_stream* indica que la solicitud no está relacionada con los flujos de datos. El establecimiento de dichos flujos implica la colaboración de los módulos RM, ARM y DSM tal y como muestra la figura 5.6.

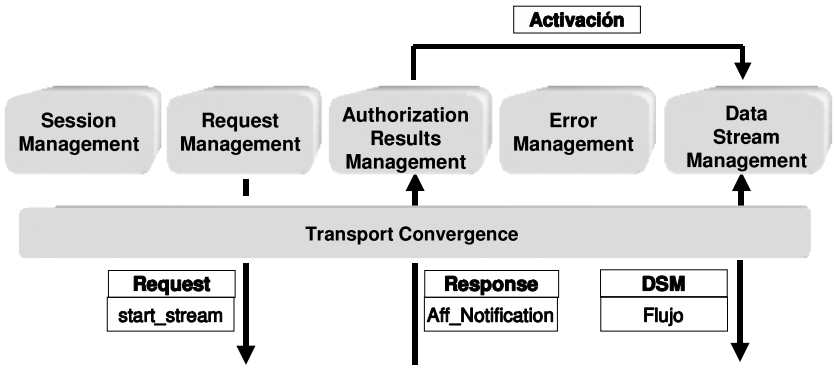

Figura 5.6: Gestión de flujos

Una vez que se establece un flujo, las aplicaciones pueden intercambiar datos libremente, los cuales estarán protegidos por la capa TC si así se negoció.

5.2.5 Análisis de seguridad del protocolo

Tal y como se ha visto en los apartados anteriores, el protocolo presentado ofrece mecanismos para la protección de los datos intercambiados. Esta sección presenta un breve

análisis técnico de la seguridad criptográfica de dicho protocolo que tiene como fin identificar las medidas tomadas contra ataques pasivos y activos bien conocidos. Análisis de seguridad más exhaustivos, por ejemplo empleando lógicas de autenticación [34], están fuera del ámbito de este trabajo.

Análisis del módulo TC

Las claves de sesión que protegen la información (claves de cifrado y de autenticación) son generadas a partir de las cargas aleatorias intercambiadas al principio de la sesión y del secreto compartido $PremasterSecret$. Además, se emplean claves de sesión independientes para cada sentido de la conexión.

Centrándonos en la autenticación, recalcar que se hace uso de las funciones HMAC, las cuales están basadas en resúmenes digitales. Las claves criptográficas utilizadas como parámetro para las funciones HMAC tienen una longitud mínima de 128 bits, lo cual proporciona un nivel de seguridad más que aceptable.

Este método de autenticación constituye también una buena defensa frente a algunos tipos de ataques activos. Uno de los ataques activos más comunes relacionados con el método de cifrado simétrico de bloque CBC es el denominado ataque de *Cut-and-paste* [181]. El ataque está basado en la sustitución de bloques del criptograma actual por bloques de criptogramas anteriores, formando así un nuevo criptograma que al descifrarse contendrá tres tipos de información: información correcta que no ha sido alterada, información falsa del criptograma anterior y datos aleatorios. Sin un método de control de la integridad de los mensajes, el receptor de los mismos podría interpretar como correcta tanto la información aleatoria como la proveniente de criptogramas anteriores. El protocolo AMBAR previene este tipo de ataques al calcular y transmitir siempre un código de autenticación (MAC) del mensaje en claro que va a ser cifrado, código que no puede ser falsificado por un tercero al estar derivado a partir de una clave simétrica de autenticación compartida por cliente y controlador.

Sin embargo, el simple uso de un código de autenticación no es suficiente para detener posibles ataques de reenvío. Este tipo de ataques está basado en el envío por parte de un atacante de mensajes anteriores pertenecientes a la misma sesión que se desea atacar. El mecanismo utilizado por el protocolo AMBAR para evitar este tipo de ataques es la asignación de identificadores únicos a cada uno de los mensajes que se transmite, lo cual permite detectar la recepción de un mensaje reenviado. Dichos identificadores están presentes en cada uno de los mensajes RM, ARM, EM y DSM.

Análisis de los módulos RM y ARM

El módulo RM asigna un identificador único T_{ID} a cada transacción y un identificador de secuencia T_{Step} a cada mensaje intercambiado dentro de cada transacción. De esta forma, mensajes RM anteriores que pudieran ser insertados por un atacante (como por ejemplo la solicitud de un fichero de gran tamaño para provocar situaciones de denegación de servicio) son detectados e ignorados.

El módulo ARM también evita este tipo de ataques mediante la inclusión de números de secuencia en sus mensajes. Por ejemplo, un ataque basado en la transmisión de mensajes *Neg_ Notification* anteriores puede ser detectado examinando el valor de T_{ID}.

Análisis del módulo SM

El diseño de un protocolo de seguro de intercambio de información confidencial es un proceso complejo ya que no es fácil determinar si el protocolo no es susceptible de ningún tipo de ataque conocido. Este análisis se centra en tres tipos de ataques: falsificación de la identidad, alteración de los parámetros negociados y ataques de reenvío.

En relación con la falsificación de la identidad, el protocolo asume la existencia de autoridades de certificación confiables. Se da por supuesto que las entidades comunicantes disponen de los medios necesarios para comprobar la validez de los certificados intercambiados.

Respecto a los parámetros negociados, es relativamente sencillo para un atacante modificar la lista de valores contenidos en los dos primeros mensajes del protocolo, especialmente en lo que respecta a los algoritmos criptográficos, con el fin de forzar la utilización de las opciones más débiles. La sencillez de este tipo de ataque estriba en el hecho de que todos los mensajes transmitidos durante la fase SM antes de *InitSession* no están protegidos. Esta falta de protección hace que un atacante pueda interceptar y modificar los mensajes de negociación. Sin embargo, AMBAR detecta este tipo de ataque mediante la inclusión en el mensaje *InitSession* de un código de autenticación de todos los mensajes SM intercambiados. En el caso de que no se produzca ningún tipo de ataque, tanto cliente como servidor deben obtener el mismo código de autenticación. Por el contrario, si alguno de los mensajes fue alterado durante su trayecto los códigos de autenticación diferirán, lo cual invalidará completamente la negociación.

El último tipo de ataque está relacionado con el reenvío de información anterior. Un atacante podría guardar toda la información enviada por un cliente a un servidor, e instantes después iniciar una nueva comunicación con dicho servidor. Realmente, el atacante se limitaría a reenviar todos los mensajes que ha registrado anteriormente en respuesta a los mensajes del servidor. Incluso teniendo en cuenta que dicho atacante no es capaz de descifrar parte de la información que está enviando, inicialmente logra hacerse pasar por el cliente original con éxito. Sin embargo, el reenvío se detecta tras la recepción del mensaje *ActivateCrypto*, el cual incluye la firma digital del valor $MasterSecret$. Dicho valor está derivado a partir de las cargas aleatorias N_C y N_S, las cuales son distintas en cada ejecución del protocolo, imposibilitando así la reutilización de mensajes anteriores.

5.2.6 Ventajas de AMBAR

El marco AMBAR es capaz de proporcionar los mecanismos básicos de seguridad necesarios para llevar a cabo el intercambio de información de autorización en escenarios de control de acceso. Mediante este marco, es posible negociar los parámetros de autorización más apropiados para cada entorno, optimizar solicitudes de acceso haciendo uso de deci-

siones de autorización anteriores y de información previamente intercambiada, y proteger la integridad de los datos que se están intercambiando. Su uso libera a las aplicaciones de alto nivel de la necesidad de tener que codificar la información relativa a autorización como parte de los datos de alto nivel, lo cual permite aislar claramente el mecanismo de control de acceso del propósito específico de la aplicación.

5.3 DCMS: Sistema de gestión distribuida de credenciales

El segundo componente principal de la infraestructura de autorización que se presenta en este capítulo es el sistema DCMS (Distributed Credential Management System) [48, 49]. Hasta el momento se han detallado tanto las especificaciones referentes a certificados de credencial (ver sección 4.3) como el marco diseñado para poder intercambiar dicha información (ver sección 5.2). Sin embargo, es necesario un paso intermedio que conecte ambos mecanismos, es decir, un sistema capaz de crear y distribuir los certificados de credencial para que éstos puedan ser utilizados por las entidades finales a la hora de acceder a los recursos protegidos. Realizando una analogía con los sistemas de certificación de identidad, nos encontraríamos en una situación en la que, tras definir el formato de certificación X.509 y los protocolos TLS o S/MIME, sería necesario especificar todos los pasos relacionados con la gestión del ciclo de vida de dichos certificados, es decir, con la especificación de una PKI. En el caso concreto que aquí se describe, se ha definido un sistema de gestión del ciclo de vida de certificados SPKI/SDSI. Como se comentó en la sección 4.3.4, SPKI/SDSI supone la alternativa más seria hasta el momento en lo que a autorización basada en certificados se refiere, de ahí que se haya elegido como especificación a utilizar a la hora de construir el sistema.

Si bien hay gran multitud de propuestas que hacen uso de este tipo de certificados a la hora de implementar escenarios de aplicación concretos, como el acceso a objetos distribuidos CORBA [123], el control de acceso a redes WLAN [116], la protección de recursos en entornos de agentes móviles [154] o el WWW [45], la mayoría de estas iniciativas carecen de un sistema genérico de gestión de los certificados. En consecuencia, la forma en la que los usuarios solicitan los certificados de autorización, el medio por el cual se distribuyen, o la política de autorización seguida para tal efecto suele ser dependiente del sistema y está implementada, a menudo, de forma demasiado sencilla y no distribuida. Si bien este enfoque puede funcionar correctamente en determinados escenarios, entornos más complejos pueden sacar a relucir ciertas carencias en materia de escalabilidad o interoperabilidad. La generación y revocación de este tipo de certificados debería realizarse de forma estructurada y completamente distribuida.

La propuesta aquí presentada define cómo deben expresarse las solicitudes de certificación, proporciona mecanismos para satisfacer las distintas políticas de seguridad, identifica las entidades involucradas en un escenario de certificación y qué tipo de colaboración se establece entre ellas. DCMS constituye una aportación muy valiosa a la definición de siste-

mas capaces de proporcionar servicios de autorización a la mayoría de escenarios basados en delegación y roles, independientemente del entorno de aplicación en el cual se encuentren éstos ubicados. Cómo se verá en los siguientes apartados, DCMS se ha centrado principalmente en las operaciones de creación y distribución de certificados y políticas de autorización, si bien puede integrarse fácilmente con otras propuestas existentes en materia de revocación y validación de certificados SPKI [117] o de publicación en repositorios públicos [8, 95].

Esta sección está estructurada de la siguiente manera. En primer lugar se presentará un escenario de autorización genérico con el cual justificar las decisiones de diseño que han dado lugar a la definición de DCMS. A continuación, se presentará la estructura general del sistema, es decir, los principales componentes funcionales que lo componen y su relación. Posteriormente, se proporcionarán los detalles relativos a cada subsistema, en especial los relacionados con el formato de las solicitudes de certificación, políticas de autorización y entidades participantes. Finalmente, se expondrá cómo se integra el marco AMBAR con DCMS a la hora de actuar como mecanismo de intercambio de información de autorización.

5.3.1 Motivación

Con el fin de ilustrar cuáles han sido los criterios de diseño a la hora de construir DCMS, se mostrará a continuación un entorno de control de acceso basado en delegación, roles y certificados de credencial SPKI. El objetivo del estudio de dicho entorno es la extracción de las características comunes a cualquier escenario de control de acceso basado en estos elementos, lo cual nos permitirá determinar cómo debe estructurarse DCMS y qué mecanismos debe ofrecer de cara a proporcionar la máxima escalabilidad, interoperabilidad y adaptabilidad.

Los escenarios de control de acceso basados en el concepto de delegación y en el agrupamiento de usuarios mediante roles presentan una estructura similar a la mostrada en la figura 5.7.

Uso de autoridades de autorización

En estos entornos, los controladores delegan gran parte de su gestión del control de acceso en terceras partes confiables denominadas de forma genérica autoridades de autorización. De esta manera, la determinación de qué usuarios, o grupos de usuarios, están autorizados a acceder a los recursos se realiza de forma distribuida por parte de cada una de dichas autoridades, las cuales actuarán según lo especificado en su política de autorización. Es decir, se considera que una autoridad de autorización puede ser cualquier entidad final del sistema a la cual se le hayan conferido los privilegios de gestión de un conjunto de recursos por parte del controlador de los mismos. El número, la localización o la responsabilidad de cada una de ellas es un factor totalmente dependiente del entorno de aplicación específico, y posiblemente muy dinámico, lo cual es un aspecto a tener en cuenta a la hora de diseñar el sistema DCMS con vistas a ofrecer una solución que abarque todas las posibles configuraciones.

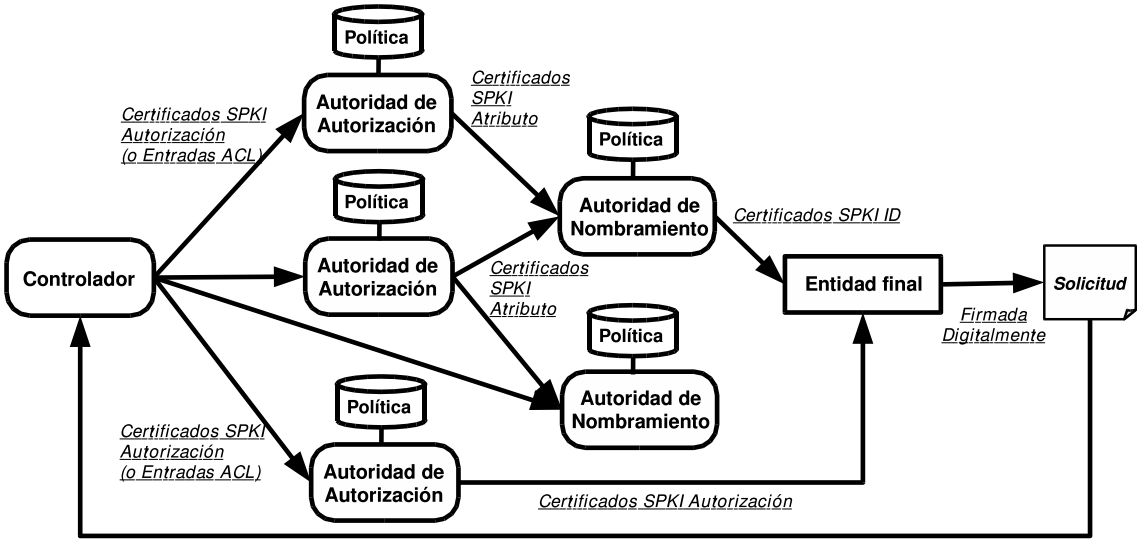

Figura 5.7: Elementos de un entorno de control de acceso basado en delegación y roles

Por otro lado, la forma mediante la cual los controladores pueden especificar esta delegación de la gestión en las autoridades puede variar mucho de un entorno a otro, sobre todo en función de la especificación de certificados de credencial que se esté empleando. En este caso concreto, debido al uso de la especificación SPKI, la delegación es posible plasmarla de dos formas distintas:

- *Mediante certificados de autorización.* Cuando el controlador correspondiente dispone de un par de claves asimétricas, es posible generar un certificado de autorización que tenga al controlador como entidad emisora y a la autoridad de autorización como entidad receptora. El conjunto de recursos que podrán ser administrados de forma descentralizada por la autoridad está contenido en el campo *tag*, siendo dicha gestión efectiva durante el periodo de validez contenido en el documento (salvo revocación). Para que la autoridad pueda actuar como tal, el certificado debe permitir la propagación de los privilegios a otras entidades del sistema.

- *Mediante entradas de una ACL.* En el caso de no disponer de dicho par de claves, el controlador puede especificar la delegación mediante el uso de listas de control de acceso SPKI. Las entradas de una ACL contienen el mismo tipo de información que un certificado de autorización, excepto en lo que respecta al campo del emisor puesto que éste está implícito. La diferencia principal entre ambos mecanismos es que la constatación de la delegación no puede hacerse pública en este último caso, al ser la ACL un documento de uso local que carece de mecanismos de protección de integridad.

Una vez que las autoridades de autorización han obtenido la responsabilidad de gestionar un conjunto de los recursos del sistema, deberán proceder con la asignación de tales

privilegios al conjunto de entidades correspondientes. Dicho conjunto, dependiente totalmente de la autoridad en cuestión, forma parte de lo que se conoce como la política de autorización de dicha autoridad. La política contiene tanto el conjunto de entidades que pueden recibir los privilegios como qué parte de los mismos y durante qué intervalo de tiempo serán asignados. Es decir, la política de autorización puede verse como una sentencia que especifica cuáles son los certificados que la autoridad estará dispuesta a emitir cuando le sean solicitados. Es importante recalcar que aunque la autoridad pueda conocer de antemano los certificados que generará en un futuro, no los emite hasta que las entidades involucradas así lo soliciten. Esto evita que, sobre todo en entornos con gran cantidad de usuarios o recursos que proteger, se produzca una generación desmesurada de certificados de credencial que conlleve a la emisión y distribución de un porcentaje de autorizaciones muy superior al que se va a hacer efectivo frente a los controladores. Como consecuencia, la especificación y el cumplimiento de las políticas de autorización deben ser otros de los mecanismos incluidos en el sistema DCMS.

Por otro lado, se pueden identificar dos tipos de entidades receptoras de los privilegios administrados por una autoridad de autorización. En primer lugar, los privilegios pueden ser asignados a un nombre previamente definido. Este nombre puede hacer referencia a un grupo de usuarios (rol) o bien a un único usuario al cual se le ha asignado un identificador dentro del sistema. La asignación a un nombre de grupo es un mecanismo implícito de redelegación característico de los sistemas basados en roles, ya que cualquier miembro del rol obtiene inmediatamente el privilegio concedido. No obstante, los privilegios también pueden ser asignados directamente a entidades finales, es decir, a claves públicas asociadas a usuarios del sistema. Este enfoque puede emplearse en los casos en los que no se haga uso del concepto de rol, o más genéricamente, cuando no se emplee ningún tipo de identificador de usuarios además de las propias claves criptográficas.

Uso de autoridades de nombramiento

La autoridades encargadas de gestionar la pertenencia a roles se denominan bajo el nombre común de autoridades de nombramiento. Al igual que sucedía con las autoridades de autorización, una autoridad de nombramiento puede estar formada por cualquier entidad final del sistema a la cual se le hayan reconocido los privilegios de gestión de un conjunto de nombres del sistema. Es importante recalcar que dicho conjunto de nombres no tiene porque hacer siempre referencia a nombres de grupo, sino que puede tratarse también de un conjunto de identificadores únicos de usuario, de ahí que se les denomine con el nombre genérico de autoridades de nombramiento. En el caso concreto que aquí nos ocupa, las autoridades reflejan la pertenencia a roles o la asignación de identificadores mediante el uso de certificados de identidad SPKI.

Como puede apreciarse en la figura 5.7, la relación entre los controladores y las autoridades de nombramiento puede ser de dos formas. Por un lado, los roles o identificadores definidos por una autoridad de nombramiento pueden estar referenciados mediante los certificados de atributo, los cuales asocian privilegios a nombres y son emitidos por las autoridades de autorización. En este sentido, quedan autorizados a acceder a los recursos

gestionados por el controlador todos aquellos usuarios que ejercen el rol especificado en dicho certificado. Por otro lado, un rol puede ser autorizado directamente por parte de un controlador mediante una entrada de su ACL o mediante un certificado de atributo emitido por dicho controlador.

Al igual que sucedía con las autoridades de autorización, cada autoridad de nombramiento está regulada por una política, en este caso denominada de nombramiento, que especifica qué elementos del sistema pertenecen a un determinado rol y durante qué periodo. Por elementos del sistema se hace referencia no sólo a entidades finales o claves públicas sino también a otros roles contenidos en uno de mayor nivel, lo cual nos lleva a la definición de sistemas $RBAC_1$ (ver sección 4.2.3).

El papel de las entidades finales

Como se ha comentado, tanto las autoridades de autorización como las de nombramiento definen en sus políticas cuáles serán los criterios a seguir a la hora de emitir nuevos certificados de credencial. Es decir, los certificados son emitidos bajo demanda, sólo cuando las entidades receptoras de los privilegios así lo solicitan a algunas de estas entidades. En consecuencia, además de todos los mecanismos identificados hasta el momento, es necesario dotar al sistema DCMS de las herramientas necesarias para que el proceso de solicitud y distribución pueda llevarse a cabo con éxito.

Dicho proceso abarca tanto la definición de un formato de solicitud de certificación (en este caso, para los tres tipos de certificados SPKI) y un sistema de comunicación entre las entidades solicitantes y las autoridades. Además, y en relación con lo comentado en la sección 4.4.3 acerca de la reducción de cadenas de delegación y anonimato, será necesario proporcionar a los usuarios finales un sistema que permita reducir parte de sus certificados de credencial en aquellos entornos en los que dicha reducción se considere un requisito desde el punto de vista del anonimato o de la eficiencia. Tanto las reducciones como las solicitudes de certificación pueden ser realizadas a través de entidades intermedias denominadas *puntos de acceso*, las cuales introducen ventajas adicionales, tal y como se verá en la sección 5.3.3.

Las entidades finales, una vez que obtienen los certificados correspondientes a partir de las autoridades del sistema, generan solicitudes de acceso a los recursos protegidos por los controladores. Dichas solicitudes deben estar firmadas digitalmente mediante la clave privada asociada a la clave pública contenida en los certificados de credencial. Una vez que esto sucede, tanto las credenciales como la solicitud se envían al controlador para que contraste la veracidad de las mismas y compruebe que existe un camino de delegación desde su propia clave pública (o lista de control de acceso) hasta la clave pública del solicitante. Consecuentemente, la cadena de delegación se valida en el mismo punto en el cual se origina, lo cual es conocido como *bucle de autorización* [17, 33].

La redelegación en claves temporales generadas por las propias entidades finales será posible siempre que los mecanismos de control de la delegación así lo permitan. En el caso concreto de SPKI, esta redelegación será factible siempre que los certificados tengan activado el campo de la propagación. Es importante recalcar que dicha redelegación no

implica a ninguna autoridad, y que puede realizarse de forma totalmente descentralizada por parte de las entidades finales, las cuales determinarán qué parte de sus privilegios propagan a sus claves temporales.

5.3.2 Estructura general de DCMS

DCMS, tal y como muestra la figura 5.8, está divido en dos grandes bloques: el subsistema NMS (Naming Management System) gestiona todos los aspectos relacionados con los certificados de identidad SPKI, es decir, con la pertenencia a roles y la identificación de entidades finales; por otro lado, encontramos el subsistema AMS (Authorization Management System), responsable de la gestión de los certificados de atributo y de autorización SPKI, es decir, de la asignación de privilegios.

Además de estos dos bloques básicos, DCMS dispone de un servicio automático de reducción de autorizaciones (RMS, Reduction Management System), el cual podría encuadrarse dentro del subsistema AMS, aunque será estudiado de forma independiente.

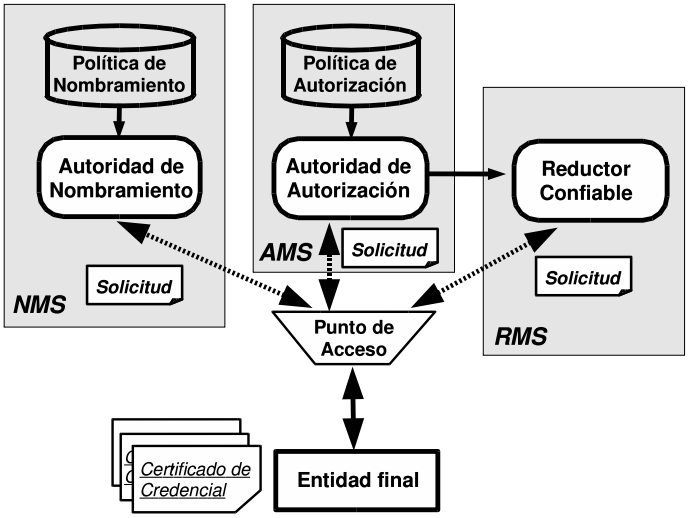

Figura 5.8: Estructura general de DCMS

Cada uno de estos bloques serán analizados siguiendo el mismo enfoque. En primer lugar, se identificarán los elementos que forman parte de su arquitectura. A continuación, se detallará tanto el formato empleado para representar las solicitudes de certificación como para reflejar las políticas de seguridad de las autoridades. Por último, se presentarán varios casos de uso que ilustran el funcionamiento de cada subsistema.

5.3.3 NMS (Naming Management System)

El subsistema NMS es responsable de las operaciones de certificación relacionadas con los certificados de identidad SPKI. Este tipo de certificados se utiliza normalmente para ligar un nombre a una determinada clave pública, así como para definir la pertenencia a grupos.

En relación con lo visto en la sección 5.3.1, un controlador podría tomar la determinación de autorizar el acceso a todos aquellos usuarios de un determinado grupo. En este caso, el sistema NMS será empleado por las entidades finales para obtener un certificado de pertenencia a dicho grupo, el cual es emitido por una autoridad de nombramiento concreta.

Entidades participantes

La figura 5.9 muestra los tres tipos de entidades que forman parte de NMS: solicitantes, puntos de acceso al servicio y autoridades de nombramiento.

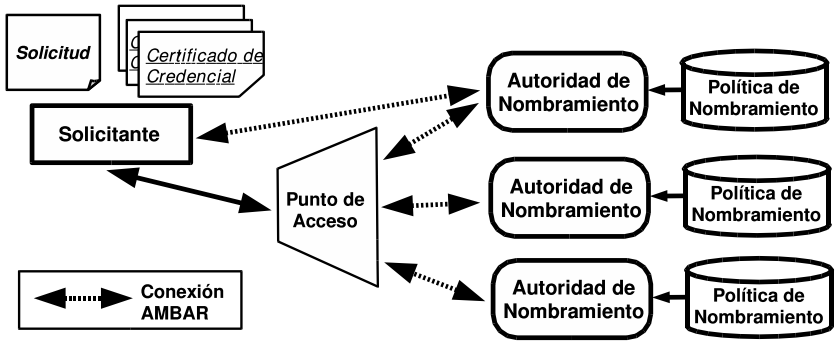

Figura 5.9: Entidades de NMS

- *Solicitante.* Son los usuarios que desean obtener un nuevo certificado de identidad SPKI. Para ello generan una solicitud de certificación y la envían a una autoridad de nombramiento en particular con el fin de obtener el certificado solicitado. Este envío puede realizarse a través de un punto de acceso o bien mediante una conexión directa AMBAR entre solicitante y autoridad. La solicitud podrá estar acompañada de otros certificados con el fin de satisfacer la política de seguridad de la autoridad.

- *Punto de acceso al servicio.* Los solicitantes pueden hacer uso de los puntos de acceso a la hora de enviar sus solicitudes de certificación a las autoridades de nombramiento apropiadas. Si bien los puntos de acceso son elementos opcionales, pueden ser considerados elementos muy útiles para los solicitantes. En primer lugar, pueden encargarse de ocultar la localización concreta de las distintas autoridades, lo cual puede ser conveniente en escenarios con varias autoridades donde resulta complicado averiguar qué autoridad es la indicada para emitir ciertos certificados, especialmente los de pertenencia a grupos. Los puntos de acceso pueden adquirir dicha información a partir de sentencias firmadas digitalmente, similares a las políticas, que contienen información acerca de la estructura del sistema y sus propiedades. De hecho, resulta más sencillo distribuir ese tipo de información al conjunto de puntos de acceso que a todas las entidades del sistema. Por otro lado, su uso puede liberar a los solicitantes de la necesidad de tener que disponer de software AMBAR para comunicar la información de la solicitud. La comunicación entre los solicitantes y los puntos de acceso

es totalmente dependiente del sistema, y podría comprender el uso de conexiones seguras o de terminales públicos.

- *Autoridad de Nombramiento (NA, Naming Authority).* Estas autoridades emiten certificados SPKI de identidad en función de las solicitudes recibidas a través de los puntos de acceso o bien directamente de los solicitantes. Están controladas por políticas de nombramiento concretas que determinan los requisitos mínimos para obtener los certificados, las cuales pueden estar implementadas mediante listas de control de acceso SPKI o mediante otros mecanismos. Cuando una NA recibe una solicitud y los posibles certificados adicionales, ejecuta un algoritmo de descubrimiento de caminos de certificación [66] con el fin de determinar si la solicitud debe aprobarse o no. La comunicación con las autoridades de nombramiento se realiza mediante el marco AMBAR. Como se comentó en la sección 5.2, AMBAR proporciona la funcionalidad necesaria para intercambiar información relativa a autorización. De esta forma, las entidades pueden autenticarse, se procede a la protección de los mensajes y es posible realizar ciertas optimizaciones destinadas a evitar el envío o el cálculo de algunas autorizaciones. La integración entre DCMS y el marco AMBAR se analizará en la sección 5.3.6.

- *Política de Nombramiento.* Se trata de documentos digitales que condicionan la toma de decisiones de las autoridades. Una política de nombramiento establece el conjunto válido de solicitantes de certificados de identidad SPKI. Asimismo, indica tanto las entidades que pueden ser asociadas a un conjunto determinado de roles como la jerarquía que forman estos últimos. Conceptualmente son muy similares a las políticas de PKI que se analizaron en la sección 3.4, en el sentido de que determinan si una solicitud cumple los requisitos necesarios para ser procesada. Sin embargo, veremos en siguientes apartados que ambas difieren tanto en el formato empleado para codificarlas (aquí se emplearán s-expresiones) como en la entidad responsable de su especificación y la metodología empleada para ello.

Formato de las solicitudes

Las solicitudes contienen información acerca del emisor del nombre, el propio nombre, el solicitante y el periodo de validez. El sistema de codificación empleado está basado en s-expresiones [170] ya que no se considera necesario hacer uso de una nueva sintaxis distinta de la utilizada en SPKI (los elementos de información empleados dentro del sistema DCMS se encuentran especificados completamente en el apéndice C). Es importante constatar el hecho de que los elementos de datos contenidos en una solicitud son los mismos que forman parte de un certificado de identidad SPKI, y que por tanto podemos hacer uso de dicha estructura para expresar las solicitudes. Las s-expresiones definidas tienen el formato presentado en la figura 5.10.

- *cert-request.* Identifica la s-expresión como una solicitud de certificación.

```
(cert-request
  (issuer (name NA_i N_i^j))
  (subject P)
  (valid ..)
)
```

Figura 5.10: Solicitudes NMS

- NA_i es la clave pública de la autoridad de nombramiento encargada de generar el certificado solicitado. En este caso, emite certificados para el nombre N_i^j.
- N_i^j. N^j es uno de los nombres definidos en el espacio de nombres de la autoridad NA_i.
- P. Es la entidad que solicita el certificado de identidad. P podría ser:
  - Una clave pública.
  - Un conjunto de entidades referenciado mediante un nombre de grupo, por ejemplo `(name NA N)`.
- *valid*. Hace referencia al periodo de validez durante el cual se solicita la asociación de P al nombre N_i^j . La codificación empleada para dicho periodo sigue el estándar SPKI [68].

En el caso de que la solicitud sea aprobada, se generará un nuevo certificado cuyo emisor será NA_i, P será el *subject*, y N_i^j será el nombre ligado a P, el cual será válido, como máximo, durante el periodo de tiempo especificado.

Las solicitudes de certificación se codifican como secuencias de dos elementos. El primer elemento es la s-expresión que codifica la solicitud y el segundo es la firma digital de la solicitud. Las firmas se codifican empleando la estructura *signature* definida en [68], y se realizan siempre usando la clave privada del solicitante.

Formato de las políticas de nombramiento

Las políticas de nombramiento se codifican utilizando una estructura muy similar a la definida por la especificación SPKI para las listas de control de acceso. Cada una de las entradas de dicha lista especifica las condiciones que deben cumplirse para generar los certificados de identidad correspondientes. El formato de dichas entradas es el mostrado en la figura 5.11.

- R hace referencia al solicitante o conjunto de solicitantes válidos de los certificados especificados en el campo *tag*. R puede ser una clave pública, un nombre `(name NA N)` o incluso un conjunto de claves públicas expresado con la construcción `(* set)`. El campo *subject* de las entradas de la políticas es opcional, lo cual implica que en el caso de que no esté presente se asumirá que el conjunto de solicitantes válidos es

```
(entry
  (subject R)?
  (tag
    (cert-request
      (issuer (name NA_i N_i^j))
      (subject P)
      (valid V_1)
    )
  )
  (valid V_2)?
)
```

Figura 5.11: Políticas NMS

igual al especificado en el campo *subject* de la s-expresión *cert-request* contenida en el campo *tag*.

- El campo *tag* especifica qué entidades pueden recibir los certificados de identidad. Tiene una estructura muy similar a la de las solicitudes NMS, aunque presenta algunas diferencias respecto a ella:

  - N_i^j puede hacer referencia a un conjunto de nombres mediante el uso de las construcciones `(* set)` y `(* prefix)`.
  - P puede hacer referencia a un conjunto de entidades. Hay dos posibilidades a la hora de expresar un conjunto de entidades. Por un lado, es posible utilizar un nombre de grupo, por ejemplo `(name NA N)`. Por otro lado, es posible usar la construcción `(* set)`.
  - V_1 hace referencia al periodo máximo durante el cual el certificado de identidad tendrá vigor.

- El valor V_2 incluido en el campo *valid* indica el periodo de tiempo durante el cual podrá solicitarse la creación de los certificados especificados en el campo *tag*. Dicho campo es opcional, lo que implica que en el caso de que no esté presente se asumirá que el periodo de solicitud es el mismo que el de validez de los certificados, es decir, igual al especificado en el campo *valid* de la s-expresión *cert-request*.

Esta política de nombramiento puede interpretarse como que la entidad (o entidades) R puede solicitar que la entidad (o entidades) P quede ligada a alguno de los nombres contenidos en N_i^j mediante la emisión de un certificado de identidad por parte de la autoridad NA_i. Dicha solicitud podrá realizarse durante el periodo V_2 y el certificado resultante tendrá una vigencia mámima V_1.

Casos de uso

Con el fin de aclarar cómo cooperan las entidades NMS para generar certificados de identidad, en este apartado se analizarán dos solicitudes de certificación. En primer lugar, se expondrá cómo crear certificados de pertenencia a grupos. A continuación, se mostrará cómo puede utilizarse NMS para definir subgrupos o jerarquías de roles. Todos los ejemplos omiten el campo relacionado con los periodos de validez por simplicidad, así como el contenido de las firmas digitales.

Pertenencia a grupos

En este ejemplo, P es una entidad que solicita un certificado de pertenencia al grupo N^j, el cual está gestionado por la autoridad NA_i. Para ello, P formula la siguiente solicitud:

```
(sequence
   (cert-request
     (issuer (name NA_i N_i^j))
     (subject P))
   (signature ..)
)
```

Esta solicitud se envía a NA_i para obtener el certificado correspondiente. Ésta será autorizada sólo en el caso de que NA_i pueda encontrar una cadena de certificación desde su ACL hasta la clave pública del solicitante. En nuestro caso, la política de autorización de la autoridad está expresada mediante la siguiente ACL:

```
(acl
  (entry
    (subject (name NA_l N_l^k))
    (tag (cert-request
      (issuer (name NA_i N_i^j))
      (subject (* set P Q R))
    ))
  )
)
```

Esta ACL especifica que sólo aquellos miembros del grupo N_l^k pueden solicitar un certificado de pertenencia para N_i^j. En el caso de que P, Q o R sean miembros de N_l^k, éstos podrán solicitar su propio certificado. De lo contrario, N_l^k puede ser considerada como una tercera parte confiable autorizada a realizar la solicitud. En este caso se asumirá que P es miembro de N_l^k, lo que conlleva que deba enviar el siguiente certificado para ser autorizado:

```
(cert
```

```
  (issuer (name NA_l N_l^k))
  (subject P)
)
```

Por último, una vez que la autoridad ha comprobado que el usuario está autorizado, se emite el certificado solicitado.

```
(cert
  (issuer (name NA_i N_i^j))
  (subject P)
)
```

Definición de subgrupos

Los subgrupos se crean mediante certificados de identidad en los que el campo *subject* es también un nombre. Hay una diferencia significativa en lo que respecta a este tipo de certificados frente a los de pertenencia a grupos. Un certificado de pertenencia suele ser solicitado por parte de la entidad que desea pertenecer al grupo, pero el certificado de subgrupo no puede ser solicitado por el subgrupo en sí. Los solicitantes válidos son dependientes de la política de nombramiento concreta, aunque algunos candidatos válidos pueden ser la propia autoridad de nombramiento que define el grupo o incluso un miembro del mismo. En el ejemplo que aquí se muestra, el solicitante autorizado es la autoridad de nombramiento, si bien ésta ha delegado dicha autorización en una tercera entidad R con el fin de evitar usar su clave privada para firmar solicitudes de certificación.

Esta es la solicitud enviada por R a NA_i con el fin de definir a N_l^k como subgrupo de N_i^j (está firmada con la clave pública de R):

```
(sequence
   (cert-request
     (issuer (name NA_i N_i^j))
     (subject (name NA_l N_l^k)))
   (signature ..)
)
```

La política especifica que NA_l puede solicitar certificados de identidad para N_i^j, y que además puede delegar dicho privilegio.

```
(acl
  (entry
    (subject NA_l)
    (propagate)
    (tag (cert-request
      (issuer (name NA_i N_i^j))
```

```
      (subject (name NA_l N_l^k))
    ))
  )
)
```

R envía además el siguiente certificado de autorización con el fin de demostrar que NA_l delegó en ella el privilegio de realizar cualquier tipo de solicitud de certificación:

```
(cert
  (issuer NA_l)
  (subject R)
  (tag (cert-request *))
)
```

Finalmente, NA_i utiliza los datos obtenidos a partir de la decisión de autorización para crear el certificado.

```
(cert
  (issuer (name NA_i N_i^j))
  (subject (name NA_l N_l^k))
)
```

5.3.4 AMS (Authorization Management System)

El subsistema AMS es responsable de las operaciones de certificación relacionadas con los certificados de atributo y de autorización SPKI. Como se verá en los siguientes apartados, tiene gran número de similitudes con el subsistema NMS.

Entidades participantes

NMS y AMS están basados prácticamente en los mismos elementos. Tanto los solicitantes como los puntos de acceso forman parte también de AMS, mientras que las autoridades de nombramiento se ven sustituidas por las autoridades de autorización.

Un solicitante AMS es una entidad que pide la generación de un nuevo certificado de atributo o de autorización. Para ello, debe construir una solicitud de certificación con información acerca de los privilegios que desea obtener (los privilegios son totalmente dependientes del entorno de aplicación). En AMS hay dos tipos de solicitantes distintos: por un lado tenemos aquellos usuarios que solicitan un certificado de autorización directa para una determinada clave pública; por otro, están aquellos que solicitan un certificado de atributo para un nombre determinado. Como veremos a continuación, los dos casos se tratan de forma distinta.

Formato de las solicitudes y las políticas

Las s-expresiones utilizadas en AMS para especificar solicitudes de certificación y las políticas de autorización están también basadas en la estructura definida por SPKI para los certificados de autorización y de atributo. Las principales diferencias con respecto a las expresiones de NMS son la presencia del campo *tag* para especificar el permiso concreto que se está solicitando o concediendo y la incorporación de un valor de control de la propagación. En los siguientes casos de uso veremos ejemplos de este tipo de s-expresiones.

Casos de uso de los certificados de autorización

En este primer ejemplo, P es una entidad que solicita un certificado de autorización con el tag tag^A a la autoridad AA_i.

```
(sequence
   (cert-request
     (issuer AA_i)
     (subject P)
     (tag tag^A))
   (signature ..)
)
```

La solicitud se envía a AA_i, y ésta examina su política de autorización para determinar si debe ser aceptada. Dicha política es la siguiente:

```
(acl
  (entry
    (tag (cert-request
      (issuer AA_i)
      (subject (* set P Q))
      (tag tag^B)
    ))
  )
)
```

Esta ACL, al omitir su campo *subject*, especifica que P y Q pueden solicitar un certificado de autorización que conceda los permisos expresados por tag^B (o un subconjunto de ellos). Si suponemos que tag^A es un subconjunto de dichos permisos, el usuario obtendría finalmente el certificado requerido.

```
(cert
  (issuer AA_i)
  (subject P)
```

```
  (tag tag^A)
)
```

Una de las principales ventajas de este enfoque es que es posible especificar un conjunto de privilegios, posiblemente infinito, sin la necesidad de tener que emitir todos los certificados asociados, ya que el conjunto necesario de éstos será emitido bajo demanda. La definición de conjuntos infinitos de certificados viene derivada de la utilización de expresiones basadas en el operador *.

Casos de uso de los certificados de atributo

Los certificados de atributo son especialmente útiles cuando se trabaja con roles, ya que pueden emplearse para especificar los permisos asignados a un determinado rol (recordemos que un certificado de atributo es aquél que contiene como campo *subject* un nombre y no una clave pública). Ahora bien, en relación con la gestión de este tipo de certificados, surge la siguiente duda: "*¿Quién debería ser el solicitante de un certificado de atributo?*". ¿Un usuario del rol al que hace referencia? ¿La autoridad que define el rol? ¿Otra entidad?

Para contestar a esta pregunta debemos considerar que los certificados son emitidos por las autoridades de autorización, y que por tanto los solicitantes válidos serán aquellos que estén reflejados en la política de seguridad de las mismas. DCMS mantiene las políticas inherentes del sistema tan al mínimo como es posible, con el fin de que sean los diseñadores del sistema los que establezcan su propia política. Por tanto, los solicitantes válidos pueden variar desde miembros del propio rol hasta gestores de rol (role managers). Esta última alternativa es muy interesante ya que divide de forma estructurada las tareas de administración entre varias entidades independientes. La forma en la que las autoridades expresan qué *role managers* pueden gestionar cada rol puede verse de la forma siguiente:

$$AA_i \Rightarrow RM_i^1(N_l^k, N_f^g), RM_i^n(N_j^h) \tag{5.1}$$

Esta expresión denota que la autoridad AA_i autoriza al *role manager* RM^1 a solicitar certificados de atributo para el grupo N^k definido por NA_l, y para el grupo N^g definido por NA_f. AA_i además autoriza a RM^n para solicitar certificados de este tipo para el grupo N^h definido por NA_j.

A continuación, vamos a ver cómo puede implementarse esta relación mediante AMS. En este ejemplo, RM_i^1 solicita un certificado de atributo para N_f^g con el privilegio tag^A. La solicitud enviada por RM_i^1 a AA_i es la siguiente.

```
(sequence
   (cert-request
     (issuer AA_i)
     (subject (name NA_f  N_f^g))
     (tag tag^A))
   (signature ..)
```

```
)
```

La política de autorización de la autoridad es la implementación de la expresión §5.1.

```
(acl
  (entry
    (subject RM_i^1)
    (tag (cert-request
      (issuer AA_i)
      (subject (* set
        (name NA_l N_l^k)
        (name NA_f N_f^g)))
      (tag tag^B)))
  )
  (entry
    (subject RM_i^n)
    (tag (cert-request
      (issuer AA_i)
      (subject (name NA_j N_j^h))
      (tag tag^C)))
  )
)
```

Finalmente, la autoridad utiliza los datos obtenidos de la decisión de autorización para crear el certificado solicitado.

```
(cert
  (issuer AA_i)
  (subject (name NA_f N_f^g))
  (tag tag^A)
)
```

5.3.5 Reduction Management System (RMS)

En la sección 4.4.3 se comentó la posibilidad de utilizar la reducción como mecanismo no sólo destinado a gestionar de forma eficiente un conjunto de credenciales sino también a proporcionar un servicio de anonimato a los usuarios del sistema. Se introdujo el concepto de *reductores confiables* como elementos capaces de simplificar un conjunto de certificados de credencial en un único documento que contuviera los privilegios derivados a partir del conjunto de partida.

Dentro de DCMS, se ha definido un sistema automático de reducción de credenciales denominado RMS (Reduction Management System), el cual ofrece la posibilidad de poner en contacto a los usuarios finales del sistema con el conjunto de reductores confiables dispo-

nibles. De esta forma, es posible enviar solicitudes de reducción y obtener los certificados resultantes como parte de los servicios disponibles en DCMS.

Entidades participantes

La principal diferencia entre RMS y los anteriores subsistemas en lo que se refiere a entidades participantes se encuentra en la presencia de los ya citados reductores confiables. Dichos elementos se relacionan con las entidades finales y las autoridades de autorización tal y como se muestra en la figura 5.12.

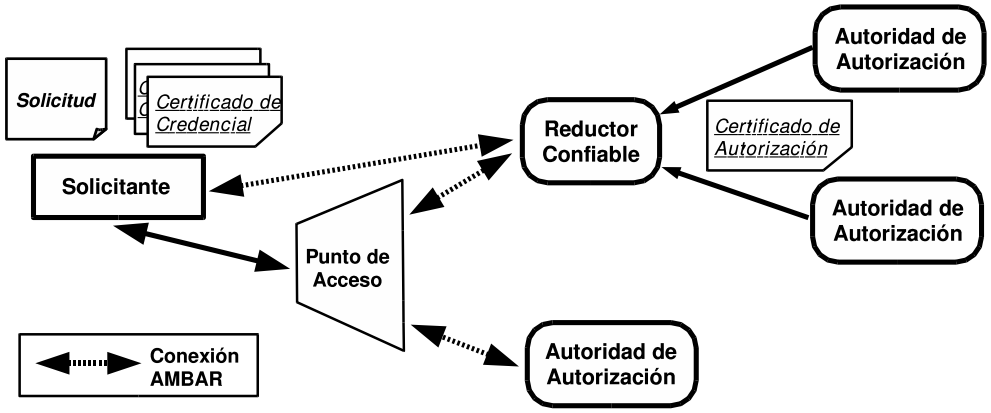

Figura 5.12: Entidades de RMS

Se considera reductor confiable a todo aquel elemento que ha recibido por parte de alguna autoridad de autorización el privilegio de emitir parte de los permisos que ésta gestiona. Dicha concesión se materializa mediante la creación de un certificado de autorización que contiene la siguiente información:

- *Emisor*: clave pública de la autoridad de autorización
- *Receptor*: clave pública del reductor confiable
- *Propagación*: activada
- *Tag*: especificación de los privilegios concretos que pueden reducirse a partir de los certificados originales. Mediante este campo es posible controlar que sólo sean reducidos aquellos privilegios que puedan ser ejercidos de forma anónima.
- *Validez*: periodo durante el cual se pueden realizar las reducciones correspondientes.

Como se muestra en la figura 5.12, los reductores pueden ser entidades confiables para más de una autoridad de autorización, lo cual implica que éstos sean capaces de realizar reducciones relacionadas con conjuntos distintos de recursos. Del mismo modo, las autoridades pueden prescindir de estos elementos a la hora de ofrecer el servicio de reducción,

asumiendo ellas mismas la responsabilidad de ofrecer el servicio de forma directa a los usuarios. El uso, o no, de reductores es totalmente dependiente del entorno de aplicación y de la política de seguridad seguida por cada autoridad, y se debe a cuestiones relacionadas con la seguridad, disponibilidad y eficiencia de dichas autoridades.

Formato de las solicitudes

Una solicitud RMS, tal y como muestra la figura 5.13, debe contener información acerca de la autoridad raíz de la cadena de delegación, el nodo final de la cadena y el conjunto de privilegios que se pretende que estén incluidos en el certificado reducido.

```
(sequence
  (chain-reduction
    (issuer AA_i)
    (subject P)
    (tag tag_R)
  )
  (certificate ...)
  ...
  (certificate ...)
)
```

Figura 5.13: Solicitudes RMS

Mediante esta solicitud, el principal P pide a AA_i la reducción del conjunto de certificados contenidos en la secuencia. El certificado resultante contendrá todos los privilegios contenidos en tag_R que puedan ser derivados a partir de dicho conjunto.

Los certificados a reducir pueden enviarse como parte de la solicitud o bien de forma separada en el campo *Asserts* del marco AMBAR. La elección de una u otra opción es un aspecto totalmente dependiente de la implementación.

Formato de las políticas de reducción

En el caso de los reductores confiables, la política de reducción se obtiene directamente a partir de los certificados recibidos por parte de las autoridades de autorización. La figura 5.14 muestra la política derivada a partir de un certificado de autorización emitido por la entidad AA_i para el conjunto de permisos tag_R con validez dentro del intervalo V.

A diferencia de las políticas de los subsistemas NMS y AMS, el campo *subject* de cada entrada de la política no especifica el conjunto de solicitantes válidos de la reducción sino que hace referencia a la entidad raíz de la cadena a reducir. El sistema RMS exige que los solicitantes de un certificado reducido sean siempre las entidades situadas al final de la cadena de delegación, es decir, los propios receptores de los privilegios.

```
(entry
  (subject AA_i)
  (propagate)
  (tag tag_R)
  (valid V)
)
```

Figura 5.14: Política de reducción

5.3.6 Integración del marco AMBAR y DCMS

El uso del marco AMBAR a la hora de realizar el envío de solicitudes y certificados aporta varias ventajas al sistema DCMS. Frente a protocolos como SSL [9], AMBAR libera a las entidades DCMS de la necesidad de encapsular la información relativa a autorización. Además, la posibilidad de mantener sesiones abiertas entre los puntos de acceso y las autoridades permite realizar optimizaciones en el envío de información entre ambos participantes, así como evitar cálculos de autorización frecuentes.

Es importante recalcar que no resulta conveniente que las autoridades utilicen sus claves privadas de firma para establecer conexiones AMBAR. Constituye una alternativa más correcta que éstas generen pares de claves temporales destinadas a proteger las comunicaciones. Para que dichas claves sean consideradas como válidas por el resto de entidades del sistema, las autoridades deben emitir certificados de autorización que les confieran el privilegio de actuar como su *interfaz de red*. Todos aquellos certificados que contengan el tag `(tag dcms-comm)` serán interpretados, tanto por los puntos de acceso como por los solicitantes, como *cartas de presentación* de las claves destinadas a establecer conexiones AMBAR.

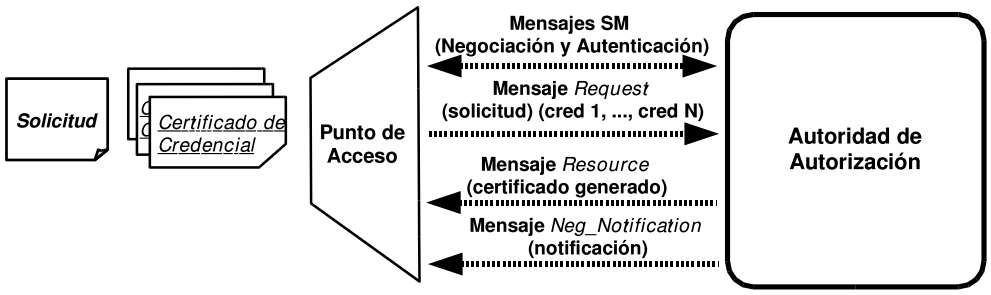

Figura 5.15: Comunicación AMBAR entre punto de acceso y autoridad

La figura 5.15 muestra los detalles de una comunicación AMBAR (en modo *pull*) entre un punto de acceso y una autoridad de nombramiento. Durante la fase SM, ambas entidades son autenticadas y se negocian las opciones de configuración de la comunicación. Las solicitudes y las credenciales se envían como parte del mensaje *Request* (perteneciente al módulo RM). La respuesta puede transmitirse empleando dos tipos de mensajes ARM. En caso de aceptación, el certificado generado se transmite dentro de un mensaje *Resource*,

mientras que en caso de rechazo éste se notifica por medio de un mensaje *Neg_ Notification*. Es importante dejar constancia de que la negociación se realiza sólo una vez por sesión y que conforme aumenta el número de solicitudes es más probable que éstas puedan ser optimizadas haciendo uso de datos anteriores.

Por otro lado, también puede negociarse el método de distribución de las credenciales necesarias para obtener el certificado. No es obligatorio que el solicitante proporcione toda la información necesaria durante el inicio de la transacción (tal y como se muestra en la figura 5.15). Otros modos posibles de distribución son, por ejemplo, el envío de credenciales tras la recepción de parte de la política de autorización de la autoridad (es decir, el envío de mensajes *Asserts* tras la recepción de mensajes *Policy*).

5.4 Metodología para la definición de estructuras de gestión de credenciales

La puesta en marcha de un sistema de control de acceso basado en roles y delegación requiere una identificación muy concisa de los elementos participantes y de la relación entre ellos. Se trata de identificar todos los recursos que se desea proteger, determinar qué acciones realizadas sobre ellos deben controlarse, descubrir cuáles son los roles fundamentales del sistema, la política de pertenencia a dichos roles, el conjunto de privilegios asociados a los mismos e identificar a las entidades encargadas de emitir los certificados correspondientes, entre otras tareas.

Cuando el desarrollo del sistema está condicionado por la utilización de enfoques de gestión estructurados como el seguido en DCMS, resulta apropiado determinar una metodología que permita coordinar tanto a las autoridades como a las entidades solicitantes o receptoras de certificados.

La especificación de metodologías para la construcción de sistemas de control de acceso basados en roles y delegación representa un campo abierto de investigación. Así pues, encontramos en la literatura algunos trabajos muy genéricos que proporcionan enfoques metodológicos de alto nivel, como las alternativas *top-down* y *bottom-up* presentadas en [20], o la identificación de niveles de control realizada en [177].

La metodología introducida en esta sección está también estructurada en niveles de gestión, tal y como muestra la figura 5.16. Como puede apreciarse, la mayor parte de los procedimientos llevados a cabo en esta metodología están organizados en dos bloques funcionalmente distintos, si bien ambos toman como punto de partida la delegación de la gestión en autoridades (nivel 0). Tal y como se verá en las próximas secciones, la determinación de los niveles pertenecientes a los bloques AMS y NMS sigue un enfoque *bottom-up* donde el diseño del sistema se realiza partiendo de los detalles más concretos hasta llegar a las características de alto nivel.

El bloque AMS está compuesto por 4 niveles de gestión distintos: identificación de relaciones entre operaciones, asignación de permisos a entidades receptoras, determinación de solicitantes y periodos de solicitud, y modos de acceso a la autoridad. El objetivo conjunto

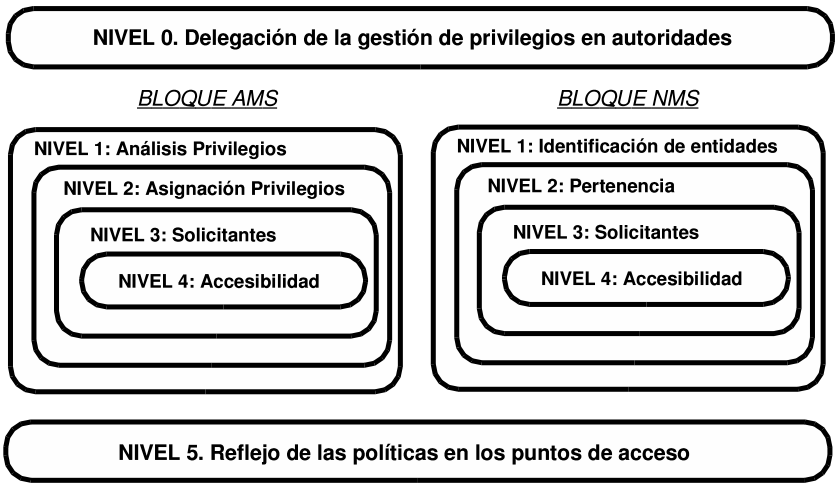

Figura 5.16: Metodología de definición de estructuras de gestión

de los procedimientos de este bloque es proporcionar a las autoridades de autorización un mecanismo para especificar sus políticas de autorización, es decir, para crear las listas de control de acceso que codifiquen dichas condiciones. El bloque será ejecutado por una autoridad siempre que ésta reciba mediante delegación la responsabilidad de asignar un conjunto de permisos relacionados con uno o más recursos del sistema.

Por otro lado, el bloque NMS está compuesto también por 4 niveles de gestión distintos: identificación del conjunto de entidades, determinación de la pertenencia, determinación de solicitantes y periodos de solicitud, y modos de acceso a la autoridad. El objetivo conjunto de los procedimientos de este bloque es proporcionar a las autoridades de nombramiento un mecanismo para especificar sus políticas de nombramiento, es decir, para crear las listas de control de acceso que codifiquen dichas condiciones. El bloque será ejecutado por una autoridad siempre que ésta reciba la responsabilidad de gestionar la pertenencia a un determinado rol del sistema.

Aunque inicialmente los procedimientos asociados tanto a los niveles AMS como NMS serán ejecutados siguiendo el orden impuesto, las continuas revisiones del sistema de control de acceso así como la necesidad de reflejar nuevas situaciones de autorización pueden hacer que la metodología deba emplearse de nuevo partiendo de alguno de los niveles intermedios. En la mayoría de los casos, la ejecución de los procedimientos de un nivel inferior (por ejemplo una nueva asignación de privilegios o la identificación de nuevos recursos a proteger) conllevará la realización de los de nivel superior (por ejemplo cambiar la forma de acceso a las auutoridades si los nuevos privilegios son considerados como críticos dentro del sistema), de ahí que en la figura 5.16 los niveles superiores aparezcan contenidos dentro de los de nivel inferior.

Por último, a partir de las políticas de autorización obtenidas, pueden derivarse los documentos de configuración del sistema que ayudan a los puntos de acceso DCMS a conocer la estructura del mismo. Como ya se comentó en la sección 5.3.3, estos documentos digitales permiten proporcionar a los usuarios finales un servicio de autorización más trans-

parente, a la vez que posibilita la ocultación de autoridades que sólo podrán ser accedidas a través de los correspondientes puntos de acceso.

5.4.1 Delegación de la gestión de privilegios en autoridades

El nivel 0 de la metodología está encargado de la especificación de los procedimientos a seguir para delegar la gestión de privilegios en distintas autoridades. Sus cuatro etapas principales tienen como objetivo determinar qué recursos protegidos por controladores serán gestionados de forma descentralizada por alguna de las autoridades del sistema.

Especificación de los recursos a proteger y sus operaciones

La primera acción a realizar es la especificación de los recursos que se desea proteger. Dicha especificación comprende la definición de un esquema de identificación de recursos, la identificación de las operaciones realizadas sobre dichos recursos, y la publicación de dicha especificación. Frente al punto de vista de algunos autores como [164], los cuales consideran que los recursos deberían ser identificados como objetivos lógicos con el fin de ocultar los detalles físicos de bajo nivel, la especificación que aquí se presenta está enfocada a la identificación de recursos de bajo nivel. Por supuesto, aplicando este mismo enfoque desde un punto de vista de más alto nivel sería posible especificar objetivos lógicos más complejos.

1. *Esquema de identificación de recursos.* Los recursos a proteger en el sistema deben ser identificados de forma que sean únicos dentro del mismo, o al menos dentro del controlador por el cual se encuentran protegidos. Esto posibilita tanto que las credenciales emitidas por las autoridades no sean ambiguas como que los usuarios finales puedan hacer referencia de forma unívoca al elemento al cual quieren acceder. A la hora de definir la notación debe considerarse la posibilidad de representar de forma apropiada los agrupamientos, colecciones o estructuras presentes entre recursos. Por ejemplo, las colecciones de ficheros se encuentran siempre organizadas de forma jerárquica, por lo que el uso de notaciones basadas en prefijos agiliza su gestión.

2. *Especificación de las operaciones sobre los recursos.* Cada tipo de recurso se caracteriza por tener asociadas un conjunto de operaciones que lo involucran. La ejecución de parte de ellas, las que tienen interés desde el punto de vista de esta metodología, debe ser protegida frente a usuarios no autorizados. La especificación de dichas operaciones abarca tanto la determinación de identificadores asociados a las mismas como la identificación de los parámetros relacionados y las posibles limitaciones temporales. Por ejemplo, en el caso de un sistema de ficheros, sería necesario concretar las operaciones a proteger (lectura, escritura, modificación, creación de directorios, etc), los parámetros de dichas operaciones (tamaño máximo de los ficheros, cuota de disco, etc) y las limitaciones temporales que pudieran imponerse (utilización exclusiva durante el horario laboral).

3. *Publicación de la especificación.* Una vez concretada la notación de los recursos y sus operaciones, es decir, la especificación de los permisos a gestionar por el sistema, ésta debe hacerse pública con el fin de que las autoridades de autorización sean capaces de emitir credenciales que conformen con la misma. Es importante recalcar que dicha especificación deber ser la misma tanto para los usuarios solicitantes como para autoridades y controladores, que o bien la emplean directamente o bien deben conocerla para realizar las traducciones pertinentes a su formato de representación interno. La publicación puede realizarse utilizando esquemas XML [31], estructuras ASN.1 [105] o bien mediante módulos software capaces de generar dichos permisos mediante mecanismos de más alto nivel.

Como resultado principal de esta etapa se extrae el formato del campo tag contenido en la s-expresión *cert-request* empleada para codificar tanto las solicitudes como las políticas de autorización.

Determinación de los controladores y los recursos implicados

Una vez que se conoce cómo hacer referencia a los recursos del sistema, el siguiente paso es determinar qué parte de ellos formarán parte del entorno de autorización. Los recursos identificados serán normalmente agrupados por conjuntos, y cada uno de estos conjuntos será asignado a un controlador distinto para que actúe como punto de cumplimiento de la política. La agrupación por controladores presenta varias ventajas: por un lado, recursos que requieren el mismo nivel de seguridad pueden ser protegidos utilizando los mecanismos ofrecidos por un mismo controlador; por otro lado, la visión de conjunto de dichos recursos puede simplificar la gestión de sus privilegios y la notación empleada para codificarlos.

No es un requisito que los controladores se encuentren en línea, ni tampoco que dispongan de su propio par de claves criptográficas. La metodología es igualmente aplicable a escenarios donde, por ejemplo, los controladores de los recursos no dispongan de conectividad, bien por tratarse de dispositivos muy sencillos (control de iluminación, apertura de puertas, etc.) o bien porque el entorno no lo requiera.

Determinación de las autoridades

A continuación, cada controlador (más concretamente la persona encargada de administrarlo) debe determinar cuáles serán las entidades que ejercerán como autoridades gestoras de los permisos involucrados. Dicha determinación es completamente dependiente del sistema, si bien suele hacer referencia a entidades del sistema con cierto peso administrativo, como responsables de administración, jefes de sección o figuras similares.

Junto con la determinación de las autoridades, los controladores deben identificar el conjunto de permisos que delegan en ellas. Dicha identificación implica la delimitación de un subconjunto de los recursos protegidos por el controlador y la selección de una serie de operaciones aplicables sobre los mismos. Una vez planificada esta asignación, sólo queda especificarla mediante alguno de los mecanismos que ya han sido analizados, es decir, mediante certificados de credencial o listas de control de acceso en el caso de SPKI. Es

importante recalcar que la delegación puede realizarse bien directamente sobre una única entidad o sobre un conjunto de elementos en forma de grupo. En el primer caso, nos encontramos con un sistema de gestión basado en autoridades de autorización, las cuales propagan los privilegios mediante certificados SPKI de autorización o de atributo. En el segundo caso, el disfrute de los privilegios está ligado a la pertenencia a cierto grupo, lo que implica que la gestión quede en manos de la autoridad de nombramiento correspondiente. Estas dos vertientes dan lugar a los bloques AMS y NMS, respectivamente, de la metodología.

En el momento en el que un controlador toma la determinación de delegar en cierta entidad la gestión del acceso a sus recursos, dicha entidad se convierte automáticamente (si no lo era ya) en una autoridad dentro de la estructura de gestión. Esto implica que dicha autoridad debe seguir los procedimientos asociados a los distintos niveles que forman parte bien del bloque AMS o del bloque NMS, según corresponda, con el fin de gestionar los privilegios que acaba de recibir.

5.4.2 Procedimientos asociados a las autoridades de autorización

El bloque de procedimientos asociados a las autoridades de autorización tiene como objetivo construir las políticas de autorización de las mismas. Lo especificado en este bloque debe ser seguido por dichas autoridades cada vez que se delega en ellas la gestión de nuevos privilegios o bien se modifican las condiciones de los actuales. Esto puede suceder tras la ejecución de los procedimientos de nivel 0, es decir, tras la identificación de nuevos recursos o controladores, o bien tras la propagación de la gestión desde una autoridad a otra. A continuación se exponen los cometidos de cada uno de los niveles pertenecientes a este bloque.

Nivel 1: Identificación de relaciones entre las operaciones

La especificación de los recursos realizada como parte del nivel de gestión 0 tiene como resultado determinar la notación empleada a la hora de codificar los recursos e identificar el conjunto de operaciones que pueden realizarse sobre los mismos. Entre dicho conjunto de operaciones podemos encontrar relaciones de exclusión, inclusión o agrupamiento que son muy útiles a la hora de asignar bloques de privilegios. Concretamente, este nivel 1 del bloque AMS intenta encontrar:

- Conjuntos de operaciones sobre un mismo recurso consideradas como mutuamente excluyentes. Es decir, operaciones que salvo en caso de control total sobre el recurso no suelen estar ligadas de forma conjunta a una misma entidad, como por ejemplo consultar las calificaciones de un examen y establecerlas.

- Conjuntos de operaciones claramente relacionadas, es decir, operaciones que suelen estar asociadas de forma conjunta a una misma entidad, como por ejemplo fichar para iniciar la jornada laboral y para terminarla.

- Rangos válidos de valores para los parámetros involucrados en las operaciones a controlar. Por ejemplo, la delimitación del número máximo de usos de cierto recurso o el número mínimo de unidades de valor necesarias para tener acceso a un documento digital.

- Intervalos temporales válidos para realizar las operaciones. Con esto no se hace referencia al periodo del tiempo durante el cual se puede disfrutar de los privilegios, es decir, al periodo de validez del certificado de credencial asociado, sino al intervalo durante el cual las operaciones pueden solicitarse. Algunos ejemplos de este tipo de intervalos se encuentran en los escenarios de control de acceso físico, los cuales pueden llegar a controlar el horario durante el cual puede abrirse cierta puerta, o en entornos de control de acceso a una red de comunicaciones en los cuales se contrate el acceso durante una determinada franja horaria.

Nivel 2: Asignación de permisos a entidades receptoras

Una vez delimitados y agrupados los permisos a asignar, el siguiente nivel AMS lo constituye el proceso de identificación y propagación de privilegios a entidades receptoras. Para ello, se llevan a cabo los siguientes pasos:

1. *Identificación de los roles y entidades individuales implicadas.* En primer lugar se determina el conjunto de elementos a los que se les desea asignar las autorizaciones pertinentes. Dichos elementos, los cuales deben encontrarse bajo la gestión de la autoridad, pueden hacer referencia tanto a entidades individuales como a roles específicos.

2. *Determinación de los bloques de permisos asociados a cada receptor.* Una vez determinados los elementos receptores, se establece qué parte de los privilegios gestionados por la autoridad son asignados a los elementos identificados mediante el paso anteriormente explicado.

3. *Control de la propagación.* La autoridad debe determinar si los permisos emitidos pueden ser propagados por parte de las entidades receptoras. En el caso del sistema DCMS, este control es booleano por ser éste el único mecanismo de limitación en la propagación que proporciona SPKI.

4. *Determinación del periodo de validez y método de validación.* Además, debe acotarse el periodo temporal durante el cual podrán disfrutarse los privilegios asignados, siendo dicha limitación muy dependiente del entorno de aplicación en cuestión. Dependiendo de la relevancia de los recursos involucrados, puede establecerse también cuál debe ser el sistema de validación de certificados a emplear.

5. *Determinación de los reductores confiables.* Por último, las autoridades deben determinar si delegan en una tercera entidad (un reductor) la capacidad de simplificar

cadenas de delegación relacionadas con el conjunto de privilegios que se están administrando. Como ya se vio en la sección 4.4.3, dicha delegación puede realizarse mediante certificados de autorización o mediante la extensión de las listas de control de acceso de los controladores. La elección de un mecanismo u otro depende del entorno de aplicación.

Una vez seguidos los procedimientos de los dos primeros niveles del bloque AMS, quedan totalmente especificadas las s-expresiones *cert-request* que aparecerán en las políticas de autorización de la autoridades.

Nivel 3: Determinación de solicitantes y periodos de solicitud

Dos son los pasos principales de este nivel de gestión. Por un lado, la especificación de las entidades que están autorizadas a solicitar los certificados de credencial de la autoridad. Dicho conjunto de entidades solicitantes es especialmente importante en los casos en los que las entidades receptoras hagan referencia a roles específicos (ver sección 5.3.4). En el caso de que sea posible que las propias entidades receptoras actúen como solicitantes (situación muy común durante la asignación a entidades individuales), la especificación de los solicitantes puede omitirse.

Por otro lado, debe determinarse el periodo temporal durante el cual podrá solicitarse la emisión de los certificados. Dicho periodo temporal podría coincidir con el periodo de validez del certificado, aunque no es obligatorio.

Al finalizar con los procedimientos de este nivel 3, las listas de control de acceso que codifican las políticas de autorización quedan totalmente especificadas.

Nivel 4: Modos de acceso a la autoridad

El último nivel del bloque AMS hace referencia al modo mediante el cual la autoridad ofrece sus servicios de emisión de certificados al resto de entidades del sistema. Los siguientes parámetros condicionan dicho modo de acceso:

- *Modo de distribución de credenciales.* Como vimos en la sección 5.2.3, el marco AMBAR proporciona varias alternativas en lo que a distribución de información de autorización se refiere. La autoridad debe determinar cuál es el modo que más satisface sus necesidades, atendiendo a parámetros como la disponibilidad de repositorios públicos de credenciales o la sensibilidad de la información contenida en la política.

- *Acceso anónimo o identificado.* Dependiendo de los privilegios gestionados, la autoridad debe determinar si permite la solicitud anónima de los mismos.

- *Acceso directo o a través de punto de acceso.* Ya se ha comentado que en ciertas situaciones puede resultar conveniente ocultar las autoridades a los solicitantes. Para ello, pueden emplearse los puntos de acceso al sistema con el fin de tener controlado el conjunto de entidades que establecen contacto con las autoridades de autorización.

El conjunto de puntos de acceso autorizados, o bien la determinación de un acceso libre por parte de cualquier entidad, es otra de las decisiones que forman parte de este nivel de gestión.

La figura 5.17 resume la misión de cada uno de los niveles de gestión del bloque AMS. Como puede apreciarse, invirtiendo el orden en el cual se muestran dichos niveles respecto a la figura 5.16 es posible plasmar cuál es el objetivo global de todo el bloque AMS. Del mismo modo, se aprecia el paralelismo entre los objetivos parciales de cada nivel y los elementos de información involucrados en el proceso de gestión (s-expresiones, formato de los tags, entradas de la ACL).

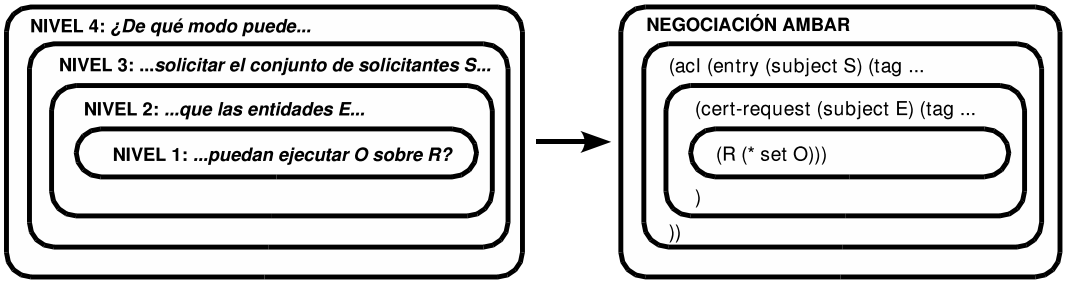

Figura 5.17: Objetivo global del bloque de procedimientos AMS

5.4.3 Procedimientos asociados a las autoridades de nombramiento

El bloque de procedimientos asociados a las autoridades de nombramiento tiene como objetivo construir las políticas de nombramiento asociadas a las mismas. Lo especificado en este bloque debe ser seguido por dichas autoridades cada vez que se les asigna la gestión de un conjunto de identificadores, los cuales suelen hacer referencia a roles concretos dentro de la organización. Esto puede suceder tras la ejecución de los procedimientos de nivel 0, es decir, tras la identificación de nuevos recursos o controladores, o bien tras la identificación de nuevos conjuntos de funciones que puedan ser asociados a un nuevo rol. A continuación se exponen los cometidos de cada uno de los niveles pertenecientes a este bloque.

Nivel 1: Identificación del conjunto de elementos

El primer nivel de gestión hace referencia a la identificación de los elementos que se encuentran dentro del ámbito de la autoridad. Dichos conjunto de elementos está compuesto tanto por entidades individuales como por roles ya definidos dentro del sistema. Otros autores [164] contemplan además la posibilidad de crear agrupaciones de nivel superior a los roles, a las cuales denominan *unidades organizativas*.

Nivel 2: Determinación de los identificadores (pertenencia)

Una vez identificados los elementos a gestionar, el siguiente nivel consiste en la determinación de los identificadores que pueden ser asignados a cada uno de ellos. En el caso de que dichos identificadores se empleen para construir grupos de usuarios (roles), debe determinarse el conjunto de todas las entidades individuales y roles de menor nivel que pertenecen al nuevo rol.

Por otro lado, en el caso de que la autoridad esté realizando una labor de identificación de claves, es decir, de asignación de nombres localmente únicos a cada uno de los elementos identificados, será necesario determinar cuál va a ser la política de asignación a seguir. Tanto la pertenencia como la asignación de nombres son dos procesos totalmente dependientes del entorno de aplicación, lo que implica que no sea posible a este nivel proporcionar detalles más concretos de cómo realizar dichos procedimientos.

Una vez seguidos los procedimientos de los dos primeros niveles del bloque NMS, quedan totalmente especificadas las s-expresiones *cert-request* que aparecerán en las políticas de nombramiento de la autoridades.

Nivel 3: Determinación de solicitantes y periodos de solicitud

Este nivel de gestión es totalmente análogo al nivel 3 del bloque AMS. El mayor énfasis debe ponerse en la identificación de las entidades solicitantes de los certificados de creación de subgrupos (ver sección 5.3.4), donde el conjunto válido de solicitantes no es tan evidente como en el caso de la pertenencia de una entidad final a un rol.

Al igual que sucedía en el bloque AMS, la ejecución de los procedimientos de nivel 3 finaliza la especificación de las listas de control de acceso que codifican las políticas de nombramiento.

Nivel 4: Modos de acceso a la autoridad

El control del acceso a la autoridad debe realizarse siguiendo los mismos criterios que ya fueron especificados para el bloque AMS. Ahora bien, es importante recalcar que, por norma general, las autoridades de nombramiento tienen un carácter más global que las autoridades de autorización. Resulta común a gran cantidad de escenarios de autorización el hecho de que un mismo rol reciba varios privilegios por parte de distintas autoridades. Además, la relación entre usuarios y roles suele ser más dinámica que la existente entre los roles y los permisos, ya que el conjunto de funciones asociadas a un rol suele cambiar menos frecuentemente que el conjunto de usuarios pertenecientes a un determinado rol. En consecuencia, esta mayor dinamicidad y globalidad deben considerarse a la hora de determinar los mejores modos de acceso a las autoridades para cada escenario.

La figura 5.18 resume la misión de cada uno de los niveles de gestión del bloque NMS. Al igual que sucedía con el bloque AMS, invirtiendo el orden en el cual se muestran dichos niveles respecto a la figura 5.16 es posible plasmar cuál es el objetivo global de todo el bloque. Del mismo modo, se aprecia el paralelismo entre los objetivos parciales de cada

nivel y los elementos de información involucrados en el proceso de gestión (s-expresiones, identificadores, entradas de la ACL).

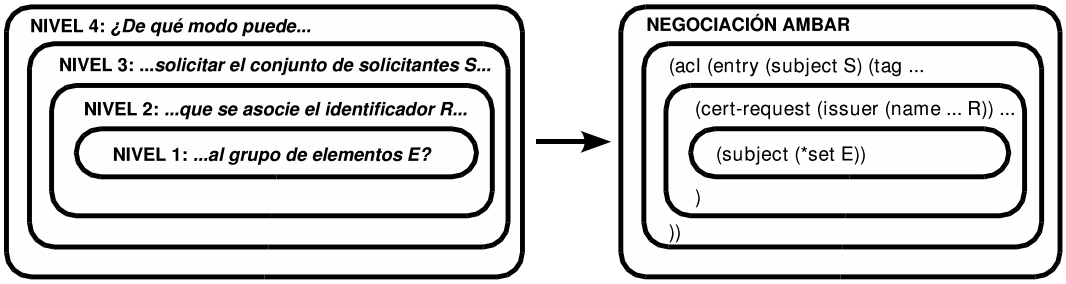

Figura 5.18: Objetivo global del bloque de procedimientos NMS

5.4.4 Reflejo de las estructuras de gestión en los puntos de acceso

Los puntos de acceso pueden operar de dos formas distintas. En primer lugar, pueden actuar como simples interfaces de comunicación entre los solicitantes y las autoridades, es decir, como software de comunicación AMBAR que evita a los usuarios disponer de una implementación del marco y que a la vez controla las direcciones desde las cuales pueden establecerse conexiones con las autoridades. En este modo de operación los puntos de acceso no necesitan disponer de ningún tipo de información especial acerca de las autoridades además de su dirección de red y sus parámetros de funcionamiento del marco AMBAR.

La otra posibilidad de funcionamiento implica la prestación de un servicio adicional de orientación a los solicitantes. Entre las ventajas que se pueden introducir encontramos:

- *Encaminamiento automático de solicitudes a las autoridades apropiadas.* El punto de acceso puede orientar al usuario acerca de la autoridad encargada de emitir los certificados relacionados con cierto tipo de privilegios o con la pertenencia a cierto grupo. De esta forma, a partir de datos como el privilegio o el rol solicitado puede completarse la especificación del emisor de la credencial.

- *Omisión de la solicitud de permisos.* Las peticiones pueden ser abortadas directamente en el punto de acceso en aquellos casos en los que los privilegios solicitados no estén siendo emitidos por ninguna autoridad o bien su validez haya expirado.

- *Comprobación de solicitudes creadas fuera del punto de acceso.* En el caso de que los solicitantes acudan con peticiones ya creadas, es decir, firmadas digitalmente utilizando una aplicación externa, es posible comprobar si los datos incluidos en dicha solicitud son correctos. Por ejemplo, puede verificarse si la entidad especificada como emisora tiene en realidad competencia para gestionar los privilegios solicitados o si éstos han caducado.

Con el fin de proporcionar estos mecanismos adicionales de orientación, los puntos de acceso deben conocer parte de la información contenida en las políticas de autorización y nombramiento de las autoridades DCMS. Tal y como se comentó en la sección 5.3.3, dicho conocimiento podría estar basado en la utilización de políticas similares a las comentadas para el caso de la PKI en la sección 3.4.

El documento que contiene las políticas de las autoridades debe incluir aquella parte de la información que no se considere sensible. Para el caso de las autoridades de autorización, se tratará del conjunto de privilegios gestionados y el intervalo de tiempo durante el cual tendrán vigor. Se omite así la información considerada como confidencial, por ejemplo el conjunto de entidades receptoras o la posibilidad de propagar dichos privilegios. En el caso de las autoridades de nombramiento, la información publicada corresponderá con el conjunto de identificadores que gestionan.

Así pues, durante el proceso de registro de una autoridad en un punto de acceso, cuestión que es totalmente dependiente de la implementación concreta del sistema DCMS, se procede a la especificación de los parámetros de comunicación AMBAR a emplear con dicha autoridad y, opcionalmente, a la provisión de un documento que contenga el conjunto de credenciales gestionados por dicha autoridad. Dicho documento deberá ser actualizado con cada cambio en la política de la autoridad.

5.5 Conclusiones

La infraestructura presentada proporciona un amplio abanico de servicios en lo que a gestión del ciclo de vida y uso de los certificados de credencial SPKI se refiere. Se ha visto cómo, partiendo de una PKI de identidad que proporcione las funciones básicas de gestión de claves de usuarios, es posible derivar un sistema que asigne privilegios a dichas claves y que permita distribuir la información a los controladores que protegen recursos sensibles.

La gestión del ciclo de vida llevada a cabo mediante DCMS proporciona un tratamiento completo a todas las cuestiones relacionadas con la certificación de privilegios. Sus cualidades más relevantes son:

- División del problema en 3 bloques conceptuales principales: gestión de la pertenencia a roles y su jerarquía, gestión de la asignación de privilegios a entidades finales o conjuntos de entidades, y gestión de la reducción de autorizaciones.

- Su diseño totalmente descentralizado y basado en delegación permite adaptarlo correctamente a escenarios en los que la gestión de los privilegios se realiza por parte de entidades con escasa conexión entre sí.

- Las políticas inherentes del sistema son mínimas. Es posible especificar entidades solicitantes distintas a las receptoras de las credenciales, periodos de solicitud distintos a los de disfrute, delegar la posibilidad de solicitud en terceras partes confiables, establecer entidades reductoras distintas de las autoridades raíz, etc.

- Los formatos de las solicitudes y de las políticas están basados en s-expresiones, sin que haya necesidad de introducir nuevos formatos de codificación distintos a los empleados por los controladores a la hora de proteger sus recursos. Además, esto proporciona cierta interoperabilidad al sistema ya que es capaz de ser ejecutado en cualquier plataforma que proporcione soporte a la especificación SPKI.

- Mediante las políticas de autorización es posible especificar conjuntos, posiblemente infinitos, de privilegios que pueden ser asociados a las entidades del sistema sin necesidad de tener que emitir los certificados previamente. El hecho de que la emisión se realice bajo demanda permite solventar algunos de los problemas de escalabilidad presentes en escenarios complejos.

- El mecanismo de reducción automática de certificados, junto con el uso de claves temporales, permite eliminar el enlace que pudiera existir entre la identidad de un usuario y sus privilegios. De esta forma, en escenarios en los que el anonimato está permitido o es un requisito, es posible ocultar la relación existente entre los certificados generados por la PKI y los del sistema DCMS. Adicionalmente, el servicio de reducción introduce mejoras en la eficiencia del tratamiento global de las autorizaciones ya que permite simplificar cadenas largas de certificación.

Por otro lado, el diseño del marco AMBAR se ha realizado considerando la diversidad de escenarios de control de acceso en los cuales resulta necesario llevar a cabo un proceso de distribución de información relativa a autorización. Entre sus características más relevantes encontramos:

- Un mecanismo de negociación de los parámetros de autorización. De esta forma, es posible adaptar el marco a escenarios con distintos requisitos de seguridad en lo que se refiere a confidencialidad, especificaciones a utilizar, anonimato, revelación de políticas y entidades responsables del cálculo de autorizaciones.

- Implementación de técnicas de optimización de sesiones, tanto en lo que respecta a la transmisión de información como al cálculo de autorizaciones.

- AMBAR se integra correctamente dentro del sistema DCMS, proporcionándole a este último un mecanismo de comunicación entre elementos que permite realizar el intercambio de solicitudes de certificación, políticas de seguridad y certificados de credencial.

Finalmente, la metodología de definición de estructuras de gestión presenta un enfoque estructurado que permite construir de forma ordenada sistemas de autorización basados en DCMS. Entre sus propiedades encontramos:

- Se proporcionan procedimientos que permiten la identificación y definición de las operaciones relacionadas con los recursos a proteger. De esta forma, es posible abordar de forma estructurada la definición de los privilegios del sistema.

- Se presentan conjuntos de procedimientos específicos tanto para la gestión de la pertenencia a roles como para la asignación de privilegios a los mismos. Por tanto, es posible abordar el diseño de ambos aspectos de forma independiente, lo que permite dividir las tareas de puesta en marcha o modificación del sistema.

- Los procedimientos se estructuran en niveles de gestión, identificándose además las dependencias entre dichos niveles con el fin de extraer cuáles son las implicaciones de la aplicación de dichos procedimientos.

- Existen una correlación clara entre los pasos de la metodología y los elementos de información del sistema DCMS que se van generando como consecuencia de su aplicación. Esto implica que la definición de las políticas pueda realizarse de forma ordenada, permitiendo la modificación de cualquiera de sus elementos sin más que aplicar los procedimientos relacionados.

- Se distingue claramente entre las políticas de emisión de credenciales, también denominadas políticas de autorización y de nombramiento, y las políticas relacionadas con el reflejo de las estructuras de gestión en los puntos de acceso. En consecuencia, no sólo se proporcionan los medios para especificar el comportamiento de las autoridades sino que también es posible plasmar cuál va a ser la dinámica del sistema, es decir, cómo se van a producir las relaciones entre los distintos elementos participantes.

El capítulo siguiente demostrará las posibilidades reales de la infraestructura a la hora de ser integrada en escenarios de aplicación que abarcan tanto el control de acceso físico a recintos como la subscripción electrónica basada en pagos electrónicos. Se analizará además cómo se ha implementado tanto el marco AMBAR como el sistema DCMS.