J. Lopez, A. Mana, J. J. Ortega, and L. Pino, "Una Solución Integral para la Autenticación de Usuarios y la Administración de Claves en Internet", *Novática*, vol. 134, pp. 20-26, 1998.
NICS Lab. Publications: https://www.nics.uma.es/publications

Una Solución Integral para la Autenticación de Usuarios y la Administración de Claves en Internet

Javier López, Lucía Pino, Antonio Maña y Juan J. Ortega
Dpto. Lenguajes y Ciencias de la Computación
E.T.S. Ingeniería Informática
Campus Teatinos, 29071 - Málaga
Tel: (95) 2131327, Fax: (95) 2131397, e-mail: jlm@lcc.uma.es

Resumen. *La seguridad es uno de los aspectos más conflictivos del uso de Internet. La falta de una política de seguridad global está frenando el desarrollo de Internet en áreas tan interesantes y prometedoras como el comercio electrónico o la interacción con las administraciones públicas. Las técnicas criptográficas actuales proporcionan un alto grado de confidencialidad; no obstante, es difícil garantizar la identificación segura de los usuarios y, además, la gestión de las claves de los mismos es poco eficiente y presenta graves problemas de escalabilidad y seguridad. En este trabajo se describe una solución a ambos problemas basada en una Infraestructura de Clave Pública que proporciona una administración simple y eficiente de las claves de los usuarios y posibilita la autenticación segura de los mismos. El sistema se ha probado con éxito de forma local y, en breve, será instalado para su prueba por parte de la comunidad de usuarios de RedIris.*

1. INTRODUCCION

Con el reciente y explosivo crecimiento de Internet una ingente cantidad de sistemas se están interconectando para formar una red global de comunicaciones. Muchos profesionales ven en Internet el medio universal de comunicación que puede reemplazar al teléfono, a la televisión o a la radio [McCu96]. El potencial está ahí, pero el progreso en campos como el comercio electrónico o la relación de los ciudadanos con las administraciones públicas ha estado obstaculizado por el propio diseño abierto de la red [Tsuj96]. Todavía es demasiado fácil interceptar, monitorizar y falsificar mensajes en Internet e, incluso, suplantar a otros usuarios. Es por ello que estos son reticentes a utilizar la red con datos financieros u otro tipo de información delicada.

Existen sistemas para la comunicación segura a través de redes públicas que utilizan sólo criptografía de clave simétrica; por ejemplo, el sistema *Kerberos* del MIT [Kohl89]. Sin embargo, tales sistemas no son escalables para grandes grupos de usuarios pertenecientes a diferentes organizaciones a pesar de las mejoras introducidas con ese objetivo [Davi95] [Gane95].

La criptografía de clave pública [DiHe76] aparece como la herramienta que mejor se adapta para satisfacer los requerimientos de seguridad de Internet, y se está convirtiendo rápidamente en la base de los sistemas de comercio electrónico en línea y otras aplicaciones que requieren seguridad y autenticación en redes abiertas.

Para la utilización global de un criptosistema de clave pública en Internet es crucial utilizar un medio práctico y fiable para la publicación y administración de esas claves: la *Infraestructura de Clave Pública* (PKI, Public Key Infrastructure). Sin una infraestructura que

funcione adecuadamente, la criptografía de clave pública es sólo marginalmente más útil que la tradicional criptografía simétrica.

La *certificación* de claves públicas es la función fundamental de todas las PKIs. Así, se define un *certificado* como el medio utilizado por una PKI para comunicar los valores de las claves públicas, la información sobre ellas, o ambas cosas. Es decir, en su forma más básica un certificado no contiene más que una clave pública, y en términos más generales es una colección de información firmada digitalmente por su emisor.

El usuario de una PKI confía en que las entidades emisoras publiquen certificados fiables que establezcan, con suficiente garantía, un nexo entre los sujetos y sus claves públicas (*certificados de identidad*), o que describan propiedades de esos sujetos (*certificados de atributo*). Cada uno de los posibles emisores de esos certificados se denomina *Autoridad de Certificación* (CA, Certification Authority).

La segunda operación básica de una PKI es la *validación* de certificados. La información del certificado puede cambiar a lo largo del tiempo por lo que el usuario del certificado necesita estar seguro de que los datos contenidos en él son fiables. Existen dos métodos básicos:

- *interactivo*: el usuario solicita directamente a la CA la confirmación de que el certificado es válido cada vez que lo va a usar.
- *diferido*: la CA incluye en el certificado una información relativa al periodo de validez (un par de fechas que definen un rango de vigencia).

Relacionado con el procedimiento de validación está el de *revocación* de certificados, el proceso por el que se le hace saber a los usuarios que la información contenida en el certificado ha quedado invalidada. Esto puede ocurrir cuando la clave privada del sujeto queda comprometida o cuando la información incluida en el certificado ha variado.

Si la validación del certificado se realiza de forma interactiva, es decir, cada vez que es necesario, entonces el problema de la revocación es trivial porque la CA indica simplemente que el certificado ya no es válido. Sin embargo, si se emplean periodos de validez, el método de revocación es crítico. Cuando no existen soluciones interactivas, el método más común es la utilización de una *Lista de Revocación de Certificados* (CRL, Certificate Revocation List). Una CRL no es más que una lista, firmada y emitida periódicamente por una CA, que contiene todos los certificados revocados. Se hace esencial que durante el proceso de validación el usuario chequee la última CRL para asegurarse de que el certificado no ha sido revocado.

Actualmente existen diferentes propuestas de infraestructuras de clave pública para Internet, muchas de las cuales están, por ahora, en la fase de desarrollo. Ninguna ha adquirido un uso generalizado en la red; de hecho, cada vez se extiende más la idea de que en un futuro próximo habrá distintos tipos de PKIs operando conjuntamente en Internet.

2. CARACTERÍSTICAS Y ANÁLISIS DE LAS INFRAESTRUCTURAS DE CLAVE PÚBLICA

La política de seguridad de una Infraestructura de Clave Pública debe establecer qué entidades pueden ejercer como CAs. Algunas de estas políticas proponen que las CAs sean unas entidades determinadas que han de cumplir una serie de requisitos. Dentro de las de este grupo, se considera como característica básica la disposición de las diferentes CAs dentro de la infraestructura. Varios sistemas utilizan una *jerarquía general* (figura 1) en la que cada CA

certifica al nodo del nivel inmediatamente superior (nodo padre) y a todos los que de ella dependen (nodos hijos), creando de esta forma las *cadenas de certificados*. Asociada a una PKI de este tipo siempre existe una Autoridad que establece una política global para la infraestructura. Esta autoridad, denominada *CA Raíz*, está localizada en el nodo origen de la jerarquía y crea las líneas de actuación que las demás CAs del sistema y todos los usuarios finales han de seguir.

En la figura 1 también se pueden observar los *certificados cruzados* (líneas discontinuas) que son certificados que no siguen la jerarquía básica. El uso de certificados cruzados permite que las cadenas de certificados sean muy pequeñas, y sólo requieren una verificación de firma además de la verificación del certificado del usuario (que siempre se hace). Pero en el caso de que existan muchas CAs, la certificación cruzada no produce una arquitectura viable porque el número de cruces es demasiado elevado.

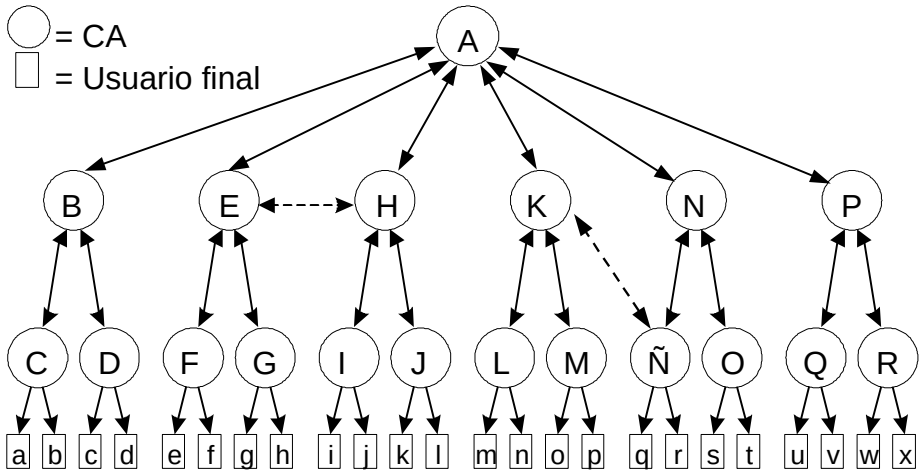


Figura 1. Jerarquía General de Certificación

Algunas propuestas de PKI para Internet, como la fallida *PEM* (Privacy Enhanced Mail [RFC 1422][RFC 1424]), utilizan una variante de la jerarquía general, la *jerarquía top-down*, en la que las CAs sólo certifican a sus nodos hijos y donde la CA raíz es la fuente de todos los caminos de certificación.

La propuesta PEM se fundamenta en que cada entidad posee un *nombre distinguido* según el estándar X.500 [ISO88], el cual supone la existencia de un directorio global que proporciona un formato para nombrar individuos, organizaciones, dispositivos, etc. El diseño de este directorio maestro mundial preveía los problemas de escalado y comenzó como una base de datos distribuida, mantenida desde múltiples puntos por múltiples personas. Una década después, esta solución no ha alcanzado su implementación global y todo parece indicar que eso no va a ocurrir.

Tomando como base el trabajo realizado en el PEM, ha surgido dentro de la *IETF* (Internet Engineering Task Force) el Grupo de Trabajo PKIX cuyo objetivo es el diseño de una infraestructura [PKIX97] que cubra las necesidades de las funciones de identificación automática, autenticación, control de accesos y autorización utilizando certificados con formato X.509 v.3 [ISO96]. Los documentos de trabajo elaborados por este grupo todavía no han sido adoptados como estándares porque no están cerradas muchas cuestiones sobre su implementación en Internet y porque las versiones desarrolladas han resultado incompletas.

Otros importantes ejemplos basados en una jerarquía top-down son las Infraestructuras de Clave Pública gubernamentales de EEUU [Chok94] [NIST96] y Canadá [CSE98].

Las recientes extensiones al *Sistema de Nombres de Dominio* (DNS, Domain Name System) [RFC1101] constituyen otra propuesta que permitiría la autenticación a través de firmas digitales. Esa propuesta es el llamado *DNS-Seguro* (Secure-DNS) [RFC2065]. Estas extensiones describen una infraestructura de clave pública, también jerárquica, integrada en la base de datos del DNS, lo que se consigue añadiendo a esta una serie de registros donde se almacenan las claves públicas de los usuarios de cada dominio. Pero, como se verá más tarde, los servidores de nombres plantean varios problemas para albergar las claves públicas porque en muchas ocasiones el DNS puede no estar estrechamente ligado a sus usuarios.

Existen otras políticas de seguridad, que no contemplan una estructura prefijada de la PKI. En ellas cada usuario puede ejercer el papel de CA con autonomía plena sobre cómo asignar su confianza. El ejemplo más clásico de este tipo de PKIs sin estructura es el de PGP (Pretty Good Privacy) [Zimm95], en el que cada usuario basa su confianza en los certificados de otros usuarios, formando así una *red de confianza* (web of trust) como muestra la figura 2. El hecho de que cada usuario pueda emitir certificados permite una gran flexibilidad y facilidad de implantación porque cada cual certifica a aquellos que conoce de forma personal.

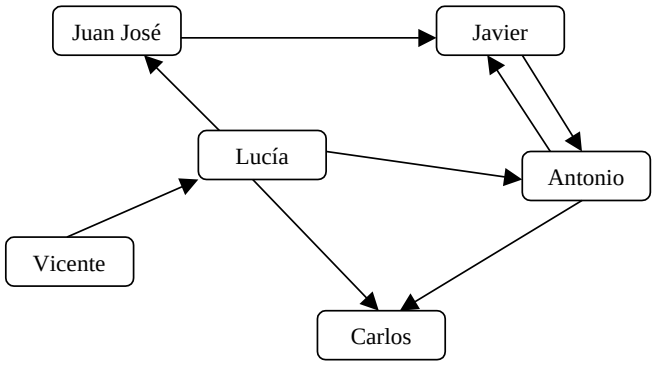


Figura 2. Ejemplo de Red de Confianza de PGP

Esta es la mejor opción para la comunicación entre un círculo cerrado de personas, como un grupo de amigos o los miembros de una empresa, pero la inexistencia de una entidad que ejerza de responsable ante situaciones problemáticas impide que este esquema se adapte bien al uso con fines comerciales por no proteger de una forma adecuada los intereses de empresas y consumidores. Además, algunos de los aspectos de la administración de claves, en especial la revocación, necesitan mecanismos específicos que introducen un factor importante de ineficiencia e inseguridad en el diseño.

Este esquema presenta también el gran inconveniente de la escalabilidad en tamaño, pues el número de certificados necesarios para lograr una comunicación global es muy grande y se generan cadenas de certificados de gran longitud. Tampoco es escalable en tiempo debido, principalmente, a los problemas asociados con el mantenimiento de las CRLs. Más aún, si se utilizan valores de confianza asociados a los arcos del grafo de certificación, entonces la tarea de encontrar el mejor camino (el del valor de confianza más alto) entre dos usuarios es un problema NP-completo.

Otras propuestas como la del *SPKI* (Simple Public Key Infrastructure) [SPKI98a], otro Grupo de Trabajo de la IETF, y la del *SDSI* (Simple Distributed Security Infrastructure) [RiLa97] son similares a la anterior en cuanto a que en su filosofía se abandona el uso de

infraestructuras globales de clave pública. El objetivo perseguido en ambas propuestas es definir un mecanismo que proporcione seguridad para un conjunto amplio de aplicaciones de Internet, incluyendo el cifrado de correo electrónico y de documentos WWW, protocolos de pago, protocolos IPsec y cualquier aplicación que requiera el uso de certificados de clave pública. Para ello pretenden producir una estructura de certificados y un procedimiento de operación con las que cubrir las necesidades de administración de confiabilidad de la comunidad de usuarios de Internet.

Estos dos esquemas comparten la idea de que cada usuario ha de estar identificado inequívocamente por un número, su clave pública, y no por un nombre común (como es el caso de los esquemas que se aproximan a la filosofía de X.500). La inexistencia de Autoridades de Certificación propiamente dichas hace que un certificado pueda ser creado y firmado por cualquier usuario. En el caso del SPKI un certificado se firma justo antes del envío hacia el verificador, mientras que en el caso del SDSI el certificado se firma para su verificación diferida y se almacena en el servidor del usuario, lugar donde el verificador acude para obtenerlo. Actualmente las dos propuestas están en proceso de fusión [SPKI98b].

3. LA NUEVA JERARQUIA PARA ADMINISTRACION DE CLAVES PUBLICAS

Para la creación de un nuevo sistema siempre es necesario realizar una serie de decisiones previas en su filosofía de diseño, estableciendo un balance entre los objetivos que se persiguen. En el caso actual del sistema de administración de claves estas decisiones condicionan la flexibilidad, la generalidad, la facilidad de uso e implantación y, sobre todo, la eficiencia y la seguridad. Por lo tanto, se definen los siguientes objetivos:

- proporcionar un medio seguro para la identificación de los usuarios y la difusión de sus claves públicas;
- utilizar una arquitectura de CAs que permita que los usuarios sean certificados por autoridades cercanas, de modo que esta certificación pueda basarse en los mismos elementos en los que basamos la confianza en el mundo real;
- diseñar una arquitectura que no plantee problemas de escalabilidad;
- evitar los problemas de sincronización propios de los esquemas que mantienen múltiples copias de claves o certificados;
- minimizar el tráfico originado por el sistema, especialmente en operaciones de mantenimiento; y
- eliminar los problemas asociados con la revocación (y por lo tanto con la validación) de certificados.

3.1 Estructura

Según se puede deducir del análisis del apartado segundo, para obtener un grado aceptable de seguridad es necesario que las claves estén certificadas por autoridades de certificación específicas y no por un usuario cualquiera. El esquema aquí descrito considera la existencia de múltiples CAs que operan de forma independiente.

En la figura 3 se muestra la estructura básica del sistema y los elementos que lo componen. La unidad básica dentro de la jerarquía es la *Unidad de Servicio de Claves* (USC),

donde se almacenan y mantienen las claves de los usuarios de la estafeta de correo electrónico correspondiente a un dominio de Internet. Al contrario que en otros sistemas, en éste la jerarquía está preestablecida, ya que existe una USC por cada estafeta de correo electrónico. De este modo el conjunto de USCs forma una jerarquía de nodos basada en la jerarquía de dominios [POLM97].

Cada USC dispone de una base de datos donde se almacenan las claves certificadas de sus usuarios. Además, por cada una de estas USCs existe una Autoridad de Certificación que es responsable de la certificación y mantenimiento de las claves y de la integridad del sistema. El sistema sólo almacena una copia certificada de la clave pública de cada usuario que, además, se encuentra en una localización conocida, la mencionada base de datos de su USC.

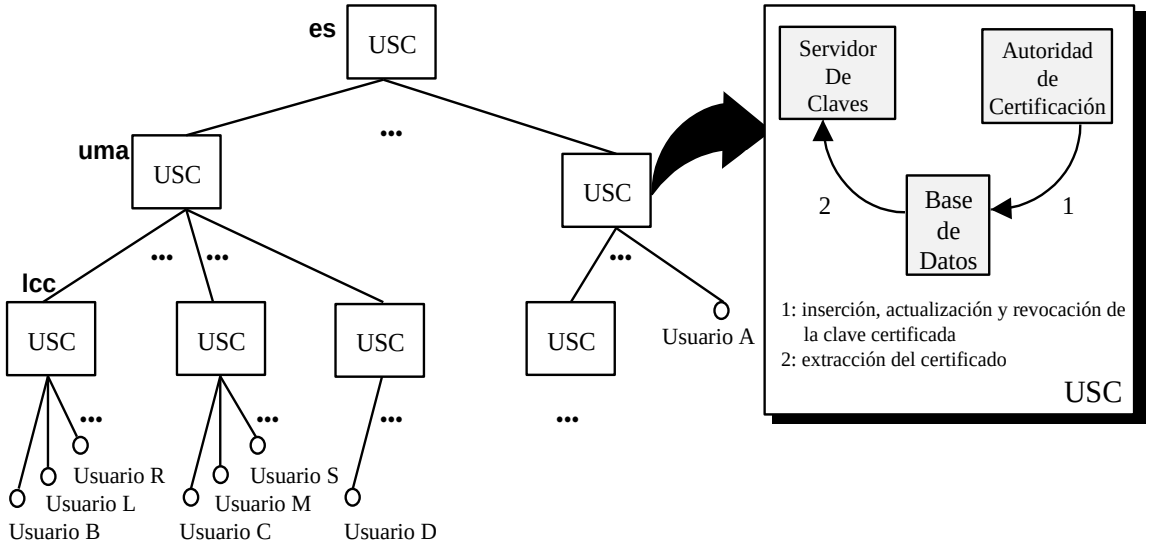


Figura 3. Jerarquía de nodos y componentes de una USC

Se puede observar que las claves certificadas son gestionadas sólo por la correspondiente CA, por lo que tanto la actualización como la revocación de tales claves son operaciones locales. Además, no es necesario disponer de la clave privada de un usuario para revocar la pública, por lo que en caso de pérdida de la primera el procedimiento es simple y no requiere medidas especiales de prevención. Más aún, si la clave a cambiar o revocar pertenece a una CA entonces no es necesario enviar los nuevos certificados e invalidar los anteriores, como ocurre en la mayoría de los sistemas con una organización jerárquica de CAs. Esto hace innecesario el establecimiento de un protocolo de notificación y re-certificación de claves para el caso de que la clave de una CA deba ser cambiada.

Debe hacerse notar que con este esquema también se elimina la necesidad de usar listas de revocación de certificados. Esto es debido a que los usuarios no distribuyen sus certificados, sino que estos se almacenan en la base de datos vinculada a la CA emisora, la cual puede modificar, eliminar o añadir nuevos certificados de las claves de los usuarios de forma directa.

Las autoridades de certificación de cada dominio están representadas por el usuario *CA@<dominio>*. El certificado de la clave de cualquier CA constituye un caso especial puesto que se registra en la base de datos de su propia USC y en la del nivel inmediatamente superior. Esto es debido a que una CA juega un doble papel, actuando por una parte como autoridad en su propio nivel y por otra como usuario en el nivel superior.

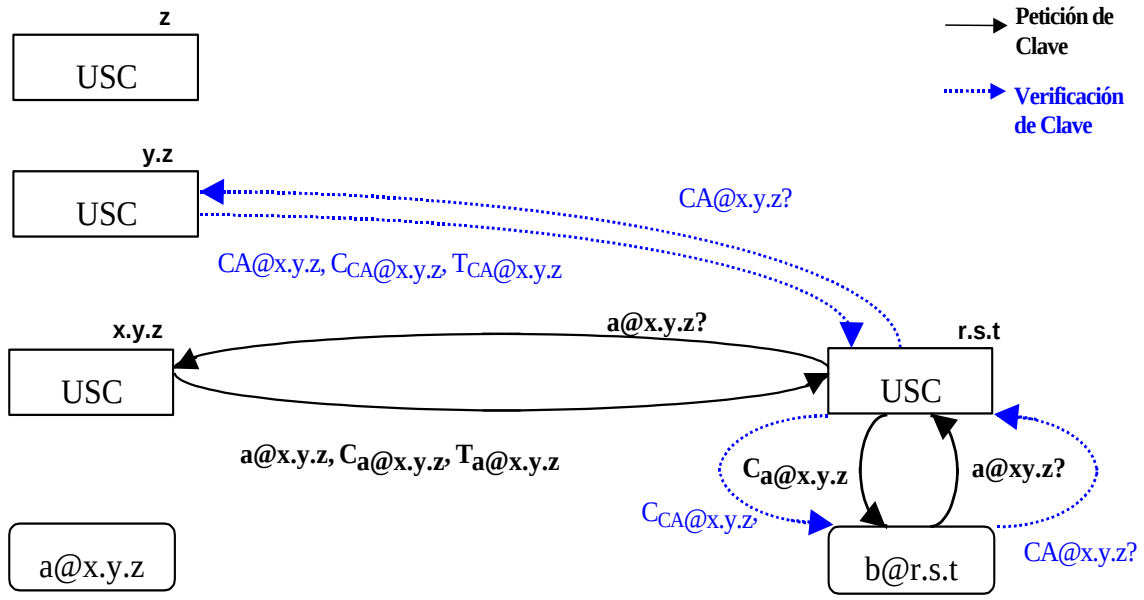
En el caso de los dominios de nivel raíz la clave de la CA correspondiente estaría certificada por la entidad registradora de dominios (por ejemplo, *I.A.N.A.* o *InterNIC.*).

3.2 Funcionamiento

Las peticiones de claves se dirigen directamente a la USC que contiene la clave. Esto es posible porque el sistema puede determinar de forma unívoca cuál es esa USC a partir de la dirección de correo electrónico del usuario. Las claves siempre están certificadas de forma que cualquier petición puede usarse para verificar la autenticidad de una clave si el solicitante lo desea.

Como se ha mencionado, cuando se realiza la petición de una clave ésta siempre lleva asociada la firma digital de la CA de su USC, por lo que a partir de ahora cuando nos refiramos a claves de usuarios se entenderá que éstas siempre forman parte de un certificado. Para verificar que un certificado es correcto se solicita la clave de la CA firmante a su USC superior, repitiéndose este proceso en sentido ascendente mientras sea necesario.

La figura 4 muestra el flujo de información que se produce cuando el usuario b@r.s.t (en adelante *b*) realiza la petición de la clave del usuario a@x.y.z (en adelante *a*).



T_j : Fecha de creación de la clave pública de *j*
C_j : Certificado del usuario *j*

Figura 4. Petición de una clave y verificación de la misma

El usuario *b* se dirige a su propia USC, la cual solicita a la USC de *a* la clave deseada. En caso de que el usuario *b* requiera mayor seguridad, (por ejemplo, quiera asegurarse de que la CA de la USC de *a* no está siendo suplantada por un impostor) puede solicitar la clave de dicha CA a la USC superior con objeto de verificar la firma que recibió en el certificado que contenía la clave de *a*. Si fuese necesario, la verificación puede continuar hasta llegar a un nodo de nivel raíz. En el caso de que en el dominio *y.z* no existiese USC, se solicita la clave de *CA@x.y.z* a la USC del dominio *z*.

3.3 Identificación de usuarios

Al desarrollar un sistema de administración de claves para Internet que garantice la identificación segura de los usuarios es necesario considerar la diferencia entre el mundo real (en el que existen las personas, las empresas y los ordenadores) y el mundo de Internet (en el que existen los nombres, las claves y los certificados). La *confianza* se origina en el mundo real basada en criterios muy dispares y debe ser posible trasladar dicha confianza para ser usada en el entorno digital.

Asimismo, hay que tener en cuenta que muchos de los certificados de identidad que se utilizan actualmente están basados en el contacto, a través de Internet, entre el certificador y el usuario. Pero esto no es suficiente, ya que puede ser necesario tener alguna prueba de la correspondencia entre la identidad real del usuario y su nombre en el ámbito de Internet. Por lo tanto, en esos casos, el concepto de confianza está malinterpretado desde el comienzo.

Particularmente, en nuestro sistema, la CA de un nivel certificará únicamente a sus usuarios, con los cuales puede establecer un procedimiento formal que permita verificar su identidad en el mundo real [Wrig98]. Por tanto, la CA asegura, al certificar la clave de un usuario, que éste ha satisfecho las normas aplicables según ese procedimiento. De esta forma se establece un nexo entre los documentos de identidad del mundo real, una dirección de correo electrónico y una clave criptográfica, lo que permite la identificación de ese usuario en Internet. La posibilidad de establecer un contacto directo con el usuario al que se certifica es importante ya que es la base para poder verificar la identidad del mismo.

Como se ha mencionado, cada usuario se identifica con una dirección de correo electrónico, por tanto los "nombres" de los usuarios son únicos a nivel global y la clave de un usuario corresponde con una dirección de correo.

Se pueden observar algunas similitudes entre nuestra propuesta y el DNS-Seguro, pues ambas coinciden en seguir la jerarquía de nombres de dominio para localizar las claves de los usuarios, pero este último utiliza los ficheros de los servidores de nombres en lugar de las estafetas de correo electrónico. La elección de las estafetas se justifica por las siguientes razones:

- Los DNS´s no están vinculados con los usuarios ya que es común que un mismo equipo se ocupe de la resolución de nombres de varios dominios, mientras que las estafetas de correo sí se identifican con el usuario y mantienen un contacto directo.
- Los DNS´s utilizan caché y mecanismos de tiempos de vida que pueden falsear los datos en caso de modificación.
- La CA de una DNS no puede tener un conocimiento preciso del usuario y esto posibilita que se puedan suplantar identidades.
- El diseño del DNS-Seguro no permite averiguar si una CA cambia de forma malintencionada las claves de sus usuarios.

Por otra parte, las críticas mas frecuentes al uso de direcciones de correo electrónico como identificadores de usuario (DN, Distinguished Names) son dos. En primer lugar, se argumenta que la relación entre un usuario real y una dirección de correo electrónico no es unívoca en ninguno de los dos sentidos. Esto es así debido a que, por una parte, un usuario puede tener varias direcciones de correo o varios alias y por otra existen direcciones (alias) que no corresponden a un usuario sino a un grupo. Un segundo problema consiste en la

posibilidad de modificar alias, ya que en determinados casos puede hacerse sin necesidad de disponer de un acceso con privilegios de administrador.

En nuestro diseño ninguna de estas críticas es aplicable ya que no se hace distinción entre un alias y un usuario normal. Esto es, el procedimiento que un usuario debe seguir para crear alias es el mismo que para crear una cuenta de correo, por lo que la verificación de la identidad del usuario es igualmente fiable. Veamos los casos posibles.

- Alias de Usuario. En este caso un usuario desea establecer un alias para su dirección de correo. Este alias llevará asociada una clave que puede ser la misma que usa en su dirección real o bien otra distinta (esto no afecta al sistema). El usuario registra en el sistema la existencia de dicho alias y su clave asociada (certificada por la CA local) en la base de datos de su USC como si de un usuario normal se tratase. Cada usuario debe registrar una clave para cada alias que desee establecer, cumpliendo en cada caso el procedimiento de registro. Así, dada una dirección de correo, sea esta real o un alias, siempre es posible obtener el usuario correspondiente. Por lo tanto, en este caso, la relación *dirección registrada en una USC* → *usuario* sí es unívoca. Un caso interesante del uso de los alias consiste en la creación de la CA de un dominio como un alias de otro usuario, el cual puede tener su propio par de claves pública-privada.
- Alias de Grupo. Si un grupo de usuarios desea establecer un alias para sus direcciones de correo deben completar el procedimiento de registro aportando sus identificaciones de tal modo que si desean establecer una clave para el alias esta debe ser compartida bajo la responsabilidad de dichos usuarios. Por tanto, aunque la relación *dirección registrada en una USC* → *usuario* no es en este caso unívoca, si existe un grupo definido e identificado de usuarios a los que está asociada. Las aplicaciones que necesiten restringir la comunicación a usuarios con identidad individual (por ejemplo algunas aplicaciones de comercio electrónico) son responsables de tomar las acciones necesarias para asegurarse de que acceden a tales usuarios individuales.

Respecto a la segunda crítica, si bien un usuario puede a veces modificar la dirección que corresponde a un alias determinado para redireccionarlo a un segundo usuario, esto no tiene ninguna relevancia a menos que el usuario registrado de ese alias entregue al otro su clave privada, lo cual sería responsabilidad del mismo. Por otra parte, no es posible que pueda certificar e insertar una clave diferente para dicho alias en la base de datos del USC puesto que no tiene acceso a la clave privada de la CA.

4. PROTOCOLO DE ACCESO AL SERVIDOR DE CLAVES

Este protocolo define el mecanismo de acceso al servidor de claves (USC) tanto por parte de usuarios particulares como de otros servidores de claves. A continuación se representan esas solicitudes como mensajes de un entorno Cliente/Servidor, teniendo en cuenta que tanto usuarios como USCs pueden actuar como clientes; por ejemplo, una solicitud puede ser del usuario *a@x.y.z* (como cliente) a la USC del dominio *x.y.z* (como servidor), o bien de esta USC (como cliente) a la USC del dominio *r.s.t* (como servidor). En adelante se utilizará *C* y *S* para denotar a un cliente o a un servidor genérico, y $C_{i.j.k}$ y $S_{i.j.k}$ para denotar a la USC del dominio *i.j.k* actuando como cliente o servidor respectivamente.

4.1 Descripción del protocolo

El protocolo está orientado a la conexión, por lo que podemos distinguir tres fases: establecimiento de la conexión, transferencia de claves y cierre de la conexión.

- *Fase de establecimiento de la conexión*

Se inicia con el siguiente mensaje:

C : HELLO [<clientID>]

donde la identificación del cliente <*clientID*> que solicita la clave es opcional, dependiendo de la implementación y de la política de seguridad.

Cuando el servidor recibe este mensaje inicia la conexión, comprueba si el cliente tiene permiso o no para establecerla y contesta con alguno de los siguientes mensajes:

S : +OK -- en caso de tener permiso

S : -ERR1 -- cuando el host del cliente no tiene permiso

S : -ERR2 -- cuando el cliente <userID> no tiene permiso

- *Fase de transferencia de claves*

Una vez completada la primera fase se solicitan las claves deseadas. La petición se realiza mediante el envío del mensaje:

C: GET KEY <userID>

donde <*userID*> es la dirección de correo electrónico del usuario cuya clave se solicita. Este campo es de la forma <*nombre*>@<*dominio*>, en el cual <*dominio*> identifica la USC cuya base de datos contiene la clave del usuario <*nombre*>.

Una vez que el servidor S con el que se ha establecido la conexión recibe ese mensaje pueden ocurrir los siguientes casos:

1. El campo <*dominio*> recibido coincide con el de S (o sea, la clave solicitada pertenece a un usuario local a S). En este caso la respuesta es:

   S : KEY <userID> <key> -- si se encontró la clave

   S : +OK;

2. El campo <*dominio*> no coincide con el de S.

   2.1. El campo <*nombre*> tiene el valor *CA*.

   2.1.1. Si el dominio de S es el dominio padre de <*dominio*> entonces la clave debe estar en la base de datos de S, en cuyo caso se responde de la misma forma que en el caso 1.

   2.1.2. Si el dominio de S se encuentra por encima de <*dominio*> en la jerarquía, entonces existe la posibilidad de que la clave de esa CA esté en la base de datos de S, en cuyo caso, se envía igual que en el anterior.
   Si no está en la base de datos, entonces ha de estar en uno de los nodos inferiores, por lo que se hace una petición al nodo hijo de S por encima de <*dominio*>.

   2.1.3. En otro caso se solicita al nodo padre de <*dominio*>, estableciendo una nueva conexión. Si en dicho nodo no existe una USC la petición se redirige al siguiente nivel superior.

2.2. El campo <*nombre*> no tiene el valor *CA*

2.2.1. Si <*dominio*> no existe se devuelve el siguiente mensaje de error:

S: -ERR

2.2.2. En otro caso se solicita al USC de <*dominio*>, estableciendo una nueva conexión. El resultado de esta conexión se transfiere al solicitante.

Cuando el cliente recibe un mensaje de error o no se encuentra la clave deseada, puede intentarse algún otro tipo de búsqueda (por ejemplo, utilizando *finger* o consultando un servidor PGP).

También se puede realizar una petición de múltiples claves mediante el siguiente mensaje:

C: MGET KEY <ID>

donde <*ID*> identifica las direcciones de correo electrónico de los usuarios cuyas claves se solicitan.

- *Fase de cierre de la conexión*

En esta fase se finaliza la conexión con el servidor de claves. Todas las conexiones establecidas han de cerrarse convenientemente.

El usuario cierra la conexión con el siguiente mensaje:

C: EXIT

El servidor confirma que se ha cerrado la conexión:

S: +OK

4.2 Proxy

Una parte importante del protocolo va a ser el algoritmo de búsqueda de claves. En esta parte es donde se debe considerar la inclusión de un proxy de claves y su política de funcionamiento.

Cada vez que se adquiere el certificado de una clave procedente de una USC de la misma infraestructura, se almacena en el proxy junto con la fecha y hora de almacenamiento.

Cuando el usuario solicita una clave a su USC se pueden dar las siguientes situaciones:

1. La clave no está en el proxy. En este caso accede a la USC apropiada como si el proxy no existiera.
2. La clave está en el proxy. En este caso, el resultado depende de la politica que elija la CA. La política más sencilla es la definición de un tiempo de caducidad pero puede dar lugar a problemas de incoherencia.
   La política más segura consiste en preguntar al servidor en el que reside la clave si ésta se ha modificado. En tal caso se obtiene la nueva clave y si no, se utiliza la que esta almacenada en el proxy.

   Se incluye, a continuación, un ejemplo de este caso:

El usuario *b@r.s.t* solicita a su USC la clave de *a@x.y.z*.

C : GET KEY a@x.y.z

Se comprueba que la clave está en el proxy, por lo que la USC de *r.s.t* inicia una conexión como cliente ($C_{r.s.t}$) con la USC de *x.y.z* ($S_{x.y.z}$) para comprobar si la clave que tiene en el proxy es válida.

$C_{r.s.t}$: HELLO

$S_{x.y.z}$: +OK

Así, en lugar de solicitar la clave mediante el comando GET, se solicita utilizando PGET de la siguiente forma:

$C_{r.s.t}$: PGET <fecha> a@x.y.z

donde el campo <fecha> es la fecha incluida en el certificado que se posee.

Si el certificado sigue siendo válido entonces se devuelve un mensaje de aceptación:

$S_{x.y.z}$: +OK

Si no es válido, $S_{x.y.z}$ envía el nuevo certificado y éste reemplaza al existente en el proxy de $C_{r.s.t}$.

$S_{x.y.z}$: KEY a@x.y.z <key>

$S_{x.y.z}$: +OK

La elección de una política apropiada para el uso del proxy puede incrementar significativamente la eficiencia de esta infraestructura. Dicha política debe adaptarse a las necesidades de los usuarios del dominio.

4.3 Características de la implementación

La implementación probada se basa en las utilidades de PGP para el almacenamiento y gestión de las claves. PGP es un software de dominio público, muy difundido en Internet, que proporciona unas herramientas fáciles de utilizar para la creación de pares de claves pública/privada, la firma de claves, la comprobación de firmas y el cifrado/descifrado de información. PGP también proporciona la gestión de las claves que se almacenan en la base de datos de la USC.

El servidor de claves utiliza el protocolo TCP en el puerto 850. Los clientes se conectan a este puerto para comunicarse con el servidor según el protocolo definido anteriormente.

5. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha descrito un nuevo sistema de administración de claves públicas y autenticación de usuarios para Internet. Se han expuesto las deficiencias de los sistemas actuales y se ha mostrado cómo este sistema puede corregirlas, aportando otras ventajas como son la revocación directa (en tiempo real) sin necesidad de CRLs y la facilidad de actualización de claves de forma transparente a todos los usuarios.

La implementación del sistema se ha probado con éxito de forma local y, en breve, será instalado para su prueba por parte de la comunidad de usuarios de RedIris, lo cual permitirá la obtención de una información contrastada sobre posibles ampliaciones o mejoras.

Actualmente se está estudiando la viabilidad de la introducción de un método que permita certificaciones cruzadas con el objeto de permitir mayor flexibilidad y conseguir verificaciones más rápidas. La solución que se estudia pretende evitar los problemas asociados con el cruce de certificados tales como la inviabilidad de la arquitectura cuando el número de cruces es demasiado elevado, la aparición de caminos de certificación cíclicos y la corrupción de la convención jerárquica de nombres como base del sistema.

6. REFERENCIAS

[Chok94] Chokhani S. *Toward a National Public Key Infrastructure*. IEEE Communications Magazine, 1994, pp. 70-74

[CSE98] Communications Security Establishment. *Government of Canada Public Key Infrastructure - White Paper*, 1998.
http://www.cse-cst.gc.ca/cse

[Davi95] Davis, D. *Kerberos Plus RSA for World Wide Web Security*. First USENIX Workshop on Electronic Commerce, 1995, pp. 185-188.

[DiHe76] Diffie, W. y Hellman, M. *New Directions in Cryptography*. IEEE Transactions on Information Theory. IT-22, n. 6. 1976, pp. 644-654.

[Gane95] Ganesan, R. *Yaksha: Augmenting Kerberos with Public Key Cryptography*. Internet Society Symposium on Network and Distributed Systems Security. IEEE Press, 1995, pp. 132-143.

[ISO88] ISO International Standard 9594. *Information Technology - Open Systems Interconnection Reference Model: The Directory*, 1988.

[ISO96] ISO/IEC JTC1/SC 21. *Draft Amendments DAM 4 to ISO/IEC 9594-2, DAM 2 to ISO/IEC 9594-6, DAM 1 to ISO/IEC 9594-7, and DAM 1 to ISO/IEC 9594-8 on Certificate Extensions*, 1996.

[Kohl89] Kohl, J. *The Use of Encryption in Kerberos for Network Authentication*. Advances in Cryptology - CRYPTO'89. LNCS 435. Springer, 1989, pp. 35-43.

[McCu96] McCurley, K. *Cryptography and the Internet: Lessons and Challenges*. Advances in Cryptology - ASIACRYPT'96. LNCS 1163. Springer, 1996, pp. 50-56.

[NIST96] National Institute of Standards and Technology. *A Public Key Infrastructure for U.S. Government Unclassified but Sensitive Applications*. 1996.

[PKIX97] PKIX Working Group Internet Draft. *Internet Public Key Infrastructure Certificate Policy and Certification Practices Framework*, 1997.
http://www.ietf.org/internet-drafts/draft-ietf-pkix-ipki-part4-02.txt

[LPMO97] Pino, L.; Ortega, J.; López, J.; Maña, A. *A System For Public Keys Service In the Spanish Research & Academic Network*. VII Annual Conference of the Internet Society, 1997.

[RFC1101] Mockapetris, P.V. *DNS Encoding of Network Names and Other Types*, 1989.

[RFC1422] Kent, S. *Privacy Enhancement for Internet Electronic Mail. Part II:Certificate-Based Key Management*, 1993.

[RFC1424] Kaliski, B. *Privacy Enhancement for Internet Electronic Mail. Part IV: Key Certification and Related Services*, 1993.

[RFC2065] Eastlake, D. y Kaufman, C. *Domain Name System Security Extensions*, 1997.

[RiLa96] Rivest, R. y Lampson, B. *SDSI – A Simple Distributed Security Infrastructure*, 1996.
http://theory.lcs.mit.edu/~cis/sdsi.html

[SPKI98a] SPKI Working Group Internet Draft. *Simple Public Key Certificate*, 1998.
http://www.ietf.org/internet-drafts/draft-ietf-spki-cert-structure-05.txt

[SPKI98b] SPKI Working Group Internet Draft. *SPKI Certificate Theory*, 1998.
http://www.ietf.org/internet-drafts/draft-ietf-spki-cert-theory-02.txt

[Tsuj96] Tsujii S. *Electronic Money and Key Management from Global and Regional Points of View*. Advances in Cryptology - ASIACRYPT'96. LNCS 1163. Springer, 1996, pp. 173-184.

[Wrig98] Wright, B. *Making Numbers Ceremonial: Signing Tax Returns with Personal Identification Numbers.* Comunicación Personal, 1998.

[Zimm95] Zimmerman, P. *The Official PGP User's Guide*. MIT Press, 1995.