Recursos informativos en las redes

Papi: una propuesta de *RedIris* para el acceso ubicuo a recursos de información

Por **Rodrigo Castro-Rojo** y **Diego R. López**

DESDE LA APARICIÓN, hace ya algunas décadas, de los protocolos internet ha sido común asociar la idea de una dirección (o un conjunto de direcciones) ip con una persona o, al menos, con una determinada institución. Sin embargo, en los últimos años se han manifestado dos tendencias que hacen cada vez más difícil establecer esta asociación.

Por un lado, la creciente movilidad de los usuarios y la penetración de internet hacen posible que las conexiones se produzcan prácticamente desde cualquier lugar: otras redes corporativas, conexiones residenciales, ciber-cafés, etc. En cada uno de estos lugares, el usuario utiliza una dirección ip completamente diferente y perteneciente, en cada caso, a conjuntos ip heterogéneos.

A esto se suma una tendencia tecnológica que ha sido constante: la aparición de dispositivos que hacen cada vez más difícil que un servidor pueda determinar la ip real de la que proviene una petición. El uso

Rodrigo Castro-Rojo y Diego R. López

de mecanismos de *enmascaramiento*[1] permite ocultar parte de las redes de una organización, mejorando su seguridad y optimizando los accesos al resto de internet. En el mismo sentido, el uso de caches web[6] se ha hecho muy común, en particular en la red académica de la que *RedIris* es responsable.

Como consecuencia, el acceso a los recursos web por parte de las organizaciones afiliadas a *RedIris* ha ido presentando más problemas.

Por otro lado, se han establecido cada vez más contratos de acceso a internet entre estas instituciones y las compañías proveedoras de contenidos[5, 3]. Durante la primavera del año 2000 *RedIris* organizó una reunión con representantes de diferentes organizaciones a ambos lados de estos problemas de conec-

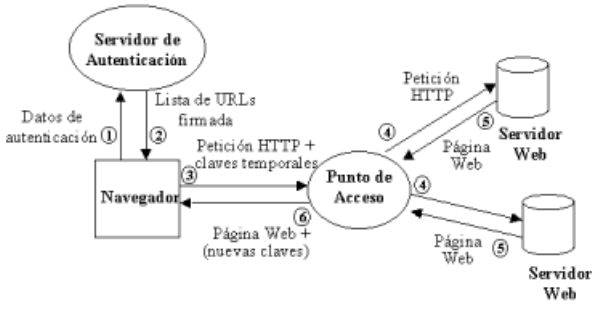

Figura 1. Arquitectura del sistema Papi, mostrando las interacciones entre el navegador del usuario y los componentes del sistema

tividad. El resultado fue una lista de requisitos para lograr una solución, así como el compromiso adquirido por *RedIris* de trabajar en un sistema de acceso que los cumpliera. Este fue el punto de partida del proyecto *Papi* (*Punto de acceso a proveedores de información*) en el que actualmente colabora *RedIris* con varias organizaciones afiliadas y con proveedores de contenidos.

2. Requisitos del sistema

Las exigencias que este nuevo modelo de acceso debía cumplir fueron compiladas por *RedIris* a partir de las peticiones de las dos clases de actores: clientes (instituciones afiliadas a *RedIris*) y proveedores (compañías de publicación). La lista de la tabla 1 es el resultado de la recopilación de las necesidades, en cierta medida contradictorias[4], de ambas partes.

3. La arquitectura del sistema *Papi*

No es normal que se ofrezca a usuarios individuales el acceso a los servidores de los proveedores de contenidos. La práctica común es permitirlo a un cierto número de ellos dentro de una organización o, como resulta más habitual, a todos los que poseen acceso a su red. Las instituciones cliente suelen disponer de bases de datos internas y de sistemas de autenticación propios, basados en datos que no pueden ser entregados a terceros. Por ejemplo: es muy frecuente permitir el uso de cierta información solamente a personas que pertenezcan a determinados departamentos o que tengan un rango concreto dentro de la organización.

Sin embargo los proveedores de información necesitan mantener el control sobre quien accede a sus servidores, definiendo los filtros adecuados para garantizar que se cumplen los acuerdos establecidos con sus clientes. Es también una práctica común que estos filtros recolecten información para proporcionar estadísticas acerca del uso de sus datos.

Estas consideraciones han llevado a dividir el modelo en dos elementos independientes: el servidor de autenticación (*As*) y el punto de acceso (*PoA*). Esta estructura, mostrada en la figura 1, hace que el sistema de acceso sea mucho más flexible y que resulte posible integrarlo en entornos muy diversos. No es necesario establecer una correspondencia uno a uno entre *As*s y *PoA*s: un *PoA* puede manejar peticiones provenientes de una cantidad indeterminada de *As*s y dirigirlas hacia cualquier número de servidores web.

Tabla 1

1. El sistema no debe permitir la entrada de usuarios no autorizados a ningún servicio.

2. El acceso debe ser permitido o denegado con independencia de la dirección ip que sea origen de las peticiones.

3. Los procedimientos de autenticación y de control de acceso estarán lo suficientemente delimitados como para ser gestionados por organizaciones independientes.

4. Cada usuario debe ser autenticado por un servidor operado por la organización a la que pertenece. Cada institución debe ser capaz de manejar su propio servidor de autenticación. Una vez se ha identificado correctamente debe tener acceso a todos los servicios para los que está autorizado durante un período limitado de tiempo.

5. Los mecanismos de autenticación utilizados por el servidor deben ser todo lo flexibles que sea posible, de manera que cada organización pueda usar su propio esquema de autenticación.

6. Los proveedores de información tendrán la capacidad de definir reglas de control de acceso basadas en los atributos públicos de una petición: organización a la que pertenece el usuario, tiempo de acceso solicitado, etc.

7. Debe garantizar la movilidad de los usuarios.

8. Los mecanismos implementados por el modelo han de ser totalmente compatibles con otros procedimientos de control de acceso empleados por los proveedores de información.

9. Se podrá acceder a los servidores de los proveedores utilizando los navegadores más comunes (*Netscape*, *IE*, *Lynx*) y desde cualquier sistema operativo.

10. Los procedimientos para el control de acceso serán transparentes para el usuario.

11. Cuando un usuario acceda a un recurso usando el sistema, su identidad no debe poder ser deducida de la información requerida para permitirle el acceso. Por otro lado esta información debe permitir diferenciar las entradas realizadas por usuarios distintos.

Esto implica garantizar la privacidad del usuario con respecto a sus patrones de acceso. Únicamente la organización a la que pertenece puede identificarlo a partir de los datos empleados para acceder a los recursos. Por su parte, los proveedores de contenidos pueden acumular estadísticas anónimas, sin que les sea posible trazar el comportamiento individual de los usuarios.

El protocolo empleado por *Papi* tiene dos fases: autenticación y control de acceso. La primera comienza en el momento en que el usuario se conecta al servidor de autenticación para obtener un conjunto de nuevas claves temporales. Durante el período de vigencia de éstas el usuario no necesita volver a contactar con dicho servidor.

Dentro de la segunda, el punto de acceso verifica las claves temporales asociadas con la información solicitada. Cada vez que el usuario trata de acceder a un sitio controlado por el *PoA*, las claves temporales son enviadas automáticamente por el navegador (sin intervención del usuario), dado que son almacenadas como *cookies* http.

3.1. El servidor de autenticación (*As*). Su propósito es ofrecer al usuario un único punto para que se identifique, y proporcionarle (de manera completamente transparente) las claves temporales que le permitirán acceder a los servicios para los que esté autorizado. La estructura funcional del servidor incluye los siguientes elementos:

—El módulo de autenticación implementa el mecanismo que emplee la organización en la que el *As* se encuentre. El servidor está diseñado para que cada organización pueda usar su propio módulo de autenticación. La distribución actual de *Papi* incluye módulos basados tanto en una base de datos específica como en servicios de autenticación externos: servidores *pop* (*post office protocol*) y servidores *ldap* (*lightweight directory access protocol*).

Figura 2. Un ejemplo de página de autenticación

—El módulo de gestión de sitios genera, a partir de la información aportada por el usuario y de las reglas definidas por la organización, una lista de los sitios a los que está autorizado para acceder y el tiempo durante el que puede hacerlo.

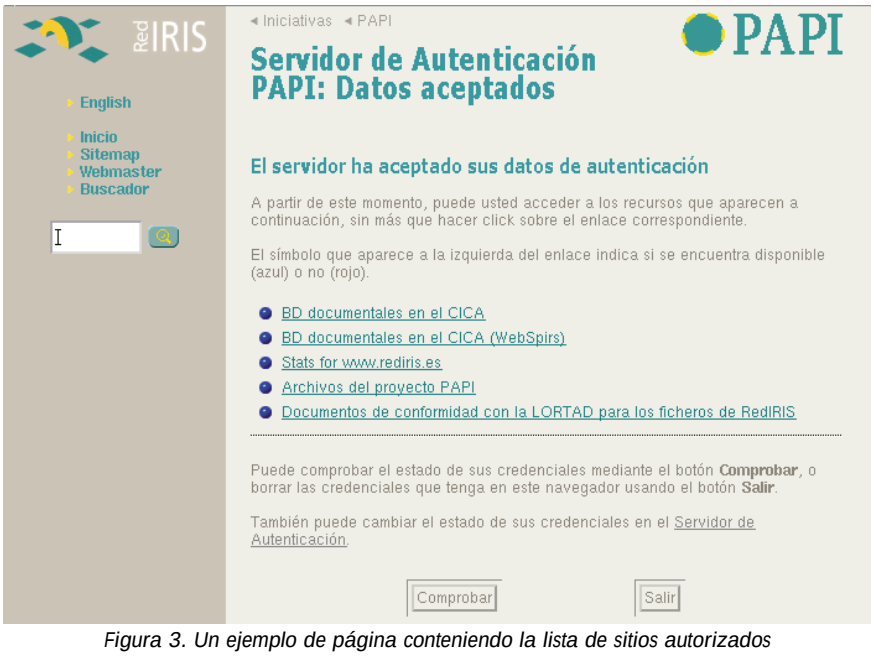

Figura 3. Un ejemplo de página conteniendo la lista de sitios autorizados

—La interfaz del servidor recibe la petición de autenticación desde el navegador del usuario y la pasa al módulo de autenticación. Si es identificado correctamente, los sitios y períodos de tiempo para este usuario son extraídos del módulo de gestión de sitios. Cada elemento de la lista se encripta usando una clave privada que representa al *As* dentro de la estructura *Papi*, y se envía de vuelta hacia el navegador del usuario como una lista de enlaces incluida dentro de una página html que contiene los resultados de la autenticación. Si el procedimiento falla se envía un mensaje de error.

Un ejemplo de página empleada por el servidor de autenticación para recoger los datos del usuario puede verse en la figura 2. En este caso se emplean los clásicos elementos “Usuario” y “Contraseña” pero el sistema puede utilizar cualquier otro (y en cualquier número). Una vez el usuario se ha autenticado correctamente, la lista de sitios autorizados es enviada a su navegador como se muestra en la figura 3.

3.2. El punto de acceso (*PoA*). Este elemento se encarga de controlar la entrada a un conjunto de sitios web para una determinada organización. El proveedor de información (o el propietario de los servidores web) es el responsable de gestionarlo. Los puntos de acceso *Papi* pueden ser adaptados a

cualquier servidor web, con independencia de su implementación. Es más, el acceso a un determinado servidor puede ser ofrecido a través de diferentes *PoA*s, a la vez que es posible que un *PoA* controle el acceso a más de un sitio web. Otra propiedad muy importante de este sistema es su completa compatibilidad con otros mecanismos de control, dado que no impone ninguna restricción en cuanto al número de procedimientos que puedan usarse con este propósito. En otros términos: el control de acceso basado en la tecnología *Papi* es ortogonal a procedimientos como la protección basada en passwords, filtros basados en direcciones ip, controles derivados de conexiones *TLS*, etc.

Un *PoA Papi* consta de dos clases de elementos fundamentales:

—Un módulo de generación de claves, a cargo de generar y enviar hacia el navegador del usuario las claves temporales necesarias para acceder a los recursos cuyo acceso controla el *PoA*. Son creadas de acuerdo con las peticiones enviadas por los servidores de autenticación reconocidos por el *PoA* y según las reglas locales.

—Uno o varios módulos de control de acceso. Cada vez que se recibe una petición para acceder a una parte del recurso protegido por el módulo en cuestión, éste se encarga de verificar las claves temporales que el navegador del usuario envía junto con la petición. Si las claves son correctas deja pasar la petición y sus resultados son devueltos al usuario.

4. Estado actual del sistema *Papi*

Ahora mismo existe una implementación de este modelo, disponible bajo licencia *GPL*[2], que está siendo evaluada y puesta a punto dentro de un proyecto piloto en el que participan el *Centro Informático Científico de Andalucía* (*Cica*), el *Consejo Superior de Investigaciones Científicas* (*Csic*), la *Universidad Autónoma de Madrid* (*UAM*), la *Universidad de Sevilla* y la *Universitat Oberta de Catalunya* (*UOC*) como entidades usuarias de información, junto a *Silverplatter* como proveedor de contenidos:

http://www.rediris.es/app/papi/

La tecnología *Papi* ha despertado el interés de otras redes académicas internacionales que están acometiendo proyectos similares con el objetivo de solucionar los problemas que detallábamos al comienzo de este artículo. En concreto, el equipo de desarrollo de *Papi* trabaja en estrecho contacto con los de *Sparta* (la nueva generación del sistema de autenticación centralizado *Athens*, empleado por la *UK Education & Research Networking Association*, *Ukerna*), en el Reino Unido, y *Shibboleth* (un proyecto de autenticación cruzada entre universidades auspiciado por *Internet2* en EUA).

Para terminar es importante resaltar que *Papi* proporciona un nuevo marco para armonizar los requisitos y necesidades tanto de los proveedores como de los consumidores de información cuando el acceso a ella se realiza en el marco de una relación contractual. La arquitectura y los protocolos que utiliza permiten superar las limitaciones que presentan las tecnologías empleadas hasta ahora para el control de acceso, garantizando tanto la independencia de los actores como la privacidad de los usuarios. Además todos los procedimientos se mantienen transparentes, de forma que su empleo no impone costes de aprendizaje y puede ser fácilmente implantado en cualquier organización.

5. Notas

1. **Danzig**, **P**.; **Swartz**, **K**. **L**. "Transparent, scalable, fail-safe web caching". *Network appliance technical report*. Consultado en: 11-09-01.
http://www.netapp.com/tech_library/3033.html

2. *Free Software Foundation*. "The GNU general public license". Consultado en: 11-09-01.
http://www.fsf.org/copyleft/gpl.html

3. **Fuchs**, **I**. "Remote authentication and authorization for *Jstor*". En: *Jstornews*, 1998, otoño, n. 2, 3.

4. **Giavarra**, **E**. "Licenses, contracts and intellectual property rights". En: *Jornadas sobre recursos electrónicos. Sociedad Española de Documentación e Información Científica (Sedic)*, 2000.

5. **Lynch**, **C**. "A white paper on authentication and access management issues in cross-organizational use of networked information resources". *Coalition for Networked Information*. Consultado en: 11-09-01.
http://www.cni.org/projects/authentication/authentication-wp.html

6. **Pearson**, **O**. "Squid (a user's guide)". Consultado en: 11-09-01.
http://squid–docs.sourceforge.net/latest/html/book1.htm

***Rodrigo Castro-Rojo** y **Diego R. López,** RedIris, Serrano 142, 28006 Madrid.*
rodrigo.castro@rediris.es
diego.lopez@rediris.es