Identidad Extendida en Redes Sociales

A. Tapiador, A. Fumero, J. Salvachúa y J. Cerviño

Universidad Politécnica de Madrid

***Resumen*— Hoy experimentamos la popularización de las plataformas web de gestión de redes sociales. La tendencia más clara va hacia la integración de las plataformas orientadas al contenido y las centradas en la gestión de contactos. Aun así, estas plataformas web siguen siendo aplicaciones aisladas que no comparten sus datos. La actividad de los usuarios en cada plataforma permanece inconexa. Este artículo propone una arquitectura distribuida de plataformas para la gestión online de redes sociales. Nuestra propuesta parte de un esquema de identidad centrado en el usuario y lo extiende agregándole información del usuario. Además, pretende cubrir la brecha entre la identidad distribuida y las capacidades para la publicación distribuida en múltiples plataformas de contenidos.**

***Palabras clave*— Identidad, Redes Sociales, Web, Web 2.0.**

I. Introducción

LAS redes sociales son uno de los conceptos clave que nos vienen a la cabeza cuando hablamos de la Web 2.0. La marginalización del valor añadido -su transformación casi en 'commodity'- de los servicios ofrecidos por las plataformas para la gestión online de las redes sociales (Social Networking Services, SNS en una de sus acepciones anglosajonas) es un hecho, una tendencia consolidada en este sector. El lanzamiento de la *plataforma* de Facebook ha levantado oficialmente la veda para el lanzamiento de una nueva ola de aplicaciones y servicios de valor añadido para una nueva generación de plataformas de SNS.

Cuando hablamos de plataformas de SNS en términos generales, estamos considerando bajo la misma denominación un amplio espectro de servicios en la Web que, básicamente, se distribuyen en el espacio que definen las redes orientadas al contenido en un extremo y las que se orientan a los contactos en el otro; extremos que pueden considerarse relacionados en cierta forma con redes sociales en sentido amplio y en sentido estricto respectivamente. La tendencia parece ser disponer de plataformas de SNS orientadas al contacto que integren servicios para compartir contenidos, siendo Facebook, otra vez, el mejor ejemplo. Al mismo tiempo que ocurre eso, seguimos teniendo un gran número de servicios independientes para compartir contenidos (e.g. Flickr, blip.tv, Youtube, Slideshare, etc.) y para la gestión de contactos (e.g. Xing, LinkedIn, etc.) que ya llevan un tiempo considerable entre nosotros.

El escenario comercial de las plataformas de SNS está viviendo un claro proceso de consolidación en términos de competencia (e.g. Xing A.G. ha adquirido las dos redes profesionales más utilizadas en España, Neurona y eConozco) y, al mismo tiempo, con el anuncio del lanzamiento de la iniciativa OpenSocial por parte de Google, todos los actores en ese escenario se posicionan para iniciar la carrera por aquella nueva generación de servicios de valor añadido.

Por lo tanto, en estos instantes tenemos un escenario repleto de múltiples ofertas de plataformas SNS. Los usuarios desarrollan su actividad personal o profesional en muchos sitios diferentes, pero toda la información generada permanece desconectada. Un usuario puede usar una plataforma de blogs para escribir sus reflexiones, una plataforma de imágenes para publicar sus fotografías, y una red social para establecer sus contactos. Todas estas plataformas ignoran la actividad que el mismo usuario desarrolla en el resto de sitios. Esto puede ser conveniente por motivos de privacidad. Sin embargo, en otras muchas ocasiones sí es deseable la interoperabilidad de los datos del usuario, de manera que, por ejemplo, se dispusieran de las imágenes para integrar en los artículos del blog, o aparezcan las últimas reflexiones del usuario en su ficha personal en la red social.

Este artículo describe una arquitectura distribuida de SNS que pretende dar solución a este problema. Propone un modelo distribuido para la integración en redes de contenidos y gestión de contactos. Está construida alrededor de entornos distribuidos de identidad como OpenID.

II. Identidad Distribuida

Los servicios de "nueva generación" de los que hablamos vienen a depender de una serie de funcionalidades básicas que ofrecemos desde nuestras plataformas. Una de esas funcionalidades clave es la identidad. En Internet, seguimos usando como estándar de facto la vieja combinación de usuario y contraseña para identificarnos y, dado que usamos un número creciente de servicios que aparecen al calor de la explosión de la Web 2.0, tenemos que gestionar no sólo una gran cantidad de contraseñas diferentes, sino un número igualmente creciente de perfiles con información personal en una serie de plataformas distribuidas por todo el mundo.

Desde el punto de vista del usuario sería deseable tener un solo perfil que se pueda validar en cualquier plataforma a la que se acceda. Los esquemas blandos de identidad, descentralizados, centrados en el usuario son una forma de

implementar esta funcionalidad. El más popular de esos esquemas es OpenID [1]. Está siendo implementado por las comunidades de desarrollo más activas de la Web 2.0, e.g. Wordpress o Blogger en el mundo de las plataformas para publicar blogs. Aún así, existen ciertas objecciones al protocolo [2], que hacen pensar en la necesidad de otro tipo de esquemas, centrados en el usuario, para los casos en que los requisitos de seguridad son más estrictos.

III. Arquitectura para una Identidad Extendida

La idea básica que hay detrás de nuestra propuesta es el salto cualitativo que representa el uso de OpenID, y por extensión de los esquemas de identidad centrados en el usuario, al identificarse en una plataforma SNS. Hasta ahora, al idenficarnos en una nueva plataforma se nos pide proporcionar usuario y contraseña, y normalmente también una dirección de correo electrónico. Si analizamos la información que el sitio web sabe sobre el nuevo usuario, tendremos unos credenciales para validar la entrada al sitio, y una dirección de correo electrónico para contactar con el usuario. Por otra parte, la identificación usando un mecanismo de identidad distribuída como OpenID, proporciona al sitio web una URL. Esta URL se puede dereferenciar, obteniendo, en principio, un documento HTML del que se puede obtener información adicional.

Usaremos ese ID para registrarnos en un montón de servicios en la Web. Esos servicios pueden incluir tanto servicios SNS en sentido amplio, como la publicación de blogs o la compartición de fotos, presentaciones, vídeo, etc., como servicios de redes sociales en sentido estricto dedicados a la gestión de contactos. Esos servicios obtienen más información acerca de nosotros de-referenciando el ID y descubriendo información asociada. También añadimos información adicional al ID desde esos servicios. De esa forma podemos conectar información y sitios web. Podríamos permitir, por ejemplo, que nuestro blog sepa de dónde son nuestras fotos (o algunas de ellas) o que nuestros amigos sepan dónde están nuestros vídeos.

A. Componentes de la Arquitectura

Nuestra solución es una arquitectura cliente-servidor y se compone de tres elementos: Client Agents, Identity Servers y Resources Servers.

1) Client Agent

Un Agente Cliente o *Client Agent* (CA) es cualquier entorno de navegación web en una máquina local o cualquier dispositivo controlado por el usuario. Ejemplos de CA son las aplicaciones para iPhone, los navegadores que se ejecutan en un PC, etc. Se presupone una conexión de red. El entorno en que se ejecuta el CA puede incluir ciertas funcionalidades añadidas a la navegación web básica, como el soporte de autenticación OpenID, la localización y asignación de recursos o la publicación de contenidos.

2) Identity Server

Un Servidor de Identidad o *Identity Server* (IS) proporciona a los usuarios los ID, pertenecientes a un determinado entorno de identidad (Identity Framework). Los usuarios necesitan autenticarse primero con su IS para poder usar su ID en el resto de plataformas de SNS.

En el mundo OpenID se conocen como "proveedores", "*OpenID Provide*r" e incorporan capacidades extendidas. Soporta además mecanismos para permitir que terceros accedan a recursos privados, utilizando protocolos como Oauth [3].

El Identity Server actúa como un proxy de usuario. Un IS almacena la información crítica de autorización de un perfil de usuario. El perfil está compuesto por enlaces a los recursos del usuario (como pueden ser presencia, geolocalización o la información de carácter personal) y las colecciones (listas de contactos, álbumes de fotos, etc.). Además, proporciona información para editar esos recursos y añadir contenido a esas colecciones.

La información del perfil está sujeta a un mecanismo de control de acceso (*Access Control Lists*, ACL). Los usuarios pueden permitir o restringir, a otras identidades (ID) del mismo Framework, el acceso a sus recursos y colecciones almacenadas en el IS. Se supone que los otros usuarios pedirán más información al IS cuando descubran el ID de un usuario. Los ID deberían ser, por tanto, susceptibles de ser descritos por unos URI "de-referenciables".

Los CA pueden acceder a los principales recursos y colecciones. Cuando el usuario establece una sesión con su IS utilizando el CA, no sólo se autentica, sino que accede a información de publicación que puede utilizar el CA. Esa información le permite al CA publicar recursos en plataformas de contenidos distintas del IS. Entre éstas actividades están escribir entradas en un blog, publicar fotos, vídeos, etc..

3) Resources Server

Con Servidor de Recursos o *Resources Server* (RS) describimos cualquier servicio que proporciona la funcionalidad de gestionar recursos. Ejemplos de RS son los típicos servicios para la publicación de contenidos (blogs, podcasts, social bookmarking) así como los servicios orientados a la gestión de contactos. También podemos pensar en los contactos como recursos, perfectamente gestionables desde la perspectiva de un RS. En principio, cualquier plataforma SNS actuaría como RS. Los RS son lo que en el mundo OpenID se conoce como tercero de confianza o "*Relaying Party*". Delegan la autenticación en los IS; aunque tienen sus propias reglas de acceso o ACLs. La sincronización entre las ACL de los IS y las de los RS debe ser especificada. Un usuario inicia una sesión en un RS utilizando el *Framework* de identidad que le proporciona su IS, por ejemplo, OpenID. El RS obtiene la información del perfil de usuario consultando a su IS. Los RS, al igual que los IS, publican colecciones de recursos en un formato estándar. Los CA reciben de cada RS una lista completa de la información de usuario que él mismo genera en ese RS específico. Los

usuarios pueden "marcar" esa información en sus IS, controlando además quién puede acceder a la misma. De esta forma, construyen su perfil. El usuario controla en el IS qué información (colecciones y recursos) muestra a otros RS. De esta manera, los RS pueden localizar y mezclar recursos y colecciones de sus usuarios.

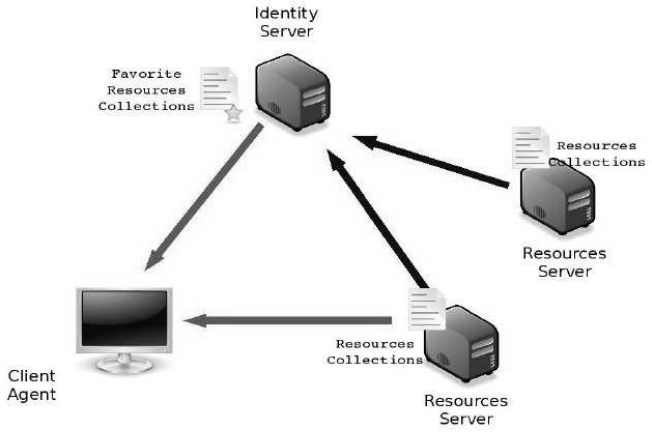

Fig. 1. Ejemplo de la arquitectura con un CA, dos RS y un IS

B. Flujos de Intercambio de información

La información de usuario en ésta arquitectura se compone de su ID, y los recursos y colecciones asociados a él. El ID es introducido por el usuario en el RS, el cual descubre la localización del IS, procedimiento descrito en la especificación del protocolo OpenID.

Posteriormente, se realizan intercambios de información entre el Servidor de Identidad y el Servidor de Recursos, los cuales tienen dos sentidos, desde el punto de vista del RS.

1) **Pull**: El Servidor de Recursos obtiene información extendida del usuario consultando al Servidor de Identidades. Esto permite a la plataforma SNS aprender más sobre el usuario, averiguando las últimas entradas en su blog o su red de contactos, por ejemplo.
2) **Push**: El Servidor de Recursos publica información del usuario en el Servidor de Identidades. Esta dirección es interesante, por ejemplo, para que el Servidor de Identidades reúna la actividad del usuario en las distintas plataformas SNS en las que participa

A continuación se discuten las tecnologías existentes que se podrían utilizar para implementar los intercambios de información descritos entre IS y RS. Para el sentido de intercambio tipo **Pull**:

1) *OpenID Attribute Exchange* [4], es una extensión del protocolo OpenID que permite intercambiar atributos entre la Relaying Party (en nuestro caso, el RS), y el Identity Provider (en nuestro caso, el IS). Los atributos se componen de pares Identificador de Tipo – Valor, lo que limita esta tecnología para el uso propuesto en nuestra arquitectura.
2) *Contenido HTML*. Los Idenficadores OpenID son típicamente URLs de HTTP. El RS puede dereferenciar el identificador, obteniendo del IS un documento HTML. Dicho documento contendría la información extendida en dos modos posibles:
   1. Microformatos[5]: Son información semiestructurada incrustada en el código HTML. Existen ya microformatos para describir tarjetas de visita (hCard), eventos (hCalendar), etiquetas (rel-tag), y se están definiendo para describir otro tipo de objetos.
   2. HEAD Links: La cabecera del documento HTML permite etiquetas del tipo *link*. Estas etiquetas ya se usan en la actualidad para proporcionar información extendida de los documentos HTML. Un ejemplo son las suscripciones a un blog, en forma de RSS o Atom.
3) *HTTP Content-Type Negociation.* El protocolo HTTP proporciona un mecanismo por el cual el cliente puede pedir el formato del documento que está dereferenciando. Este mecanismo (junto con el anterior descrito de HEAD Links), permitiría obtener la información extendida en un formato más apropiado que HTML. Un ejemplo son los feeds de Atom [6], un formato concebido para las sindicación de contenidos. Otro ejemplo son esquemas (RDFs, OWL), la base de la Web Semántica. FOAF (*Friend Of A Firend*) [7] es un vocabulario para describir agentes (*Users, Groups, Organizations*) y sus atributos. La propiedad experimental foaf:openid permite la asociación de la información de usuario con su ID. SIOC (*Semantic Interlinked Online Communities*) [8] proporciona lenguajes para la descripción de recursos y colecciones de recursos.

Y para el sentido tipo **Push**

1) *OpenID Attribute Exchange*: La extensión de OpenID funciona en ambos sentidos. Igualmente puede utilizarse por el RS para guardar pares Identificador – Valor en el IS, con las limitaciones comentadas anteriormente.
2) *Atom Publishing Protocol.* El Atom Publishing Protocol (AtomPub - RFC 5023 [9]) es un protocolo diseñado por la IETF para publicar y editar recursos en la Web. Uno de los dos tipos de documentos definidos por la especificación son los documentos de servicio (*Service Documents*). Los *Service Documents* describen las colecciones (*Collections*) disponibles agrupadas en espacios de trabajo (*Workspaces*). Las colecciones son conjuntos de recursos. El *Service Document* describe qué tipo de recursos se pueden publicar en una colección. El protocolo *AtomPub* se puede usar, por tanto, por el RS para publicar eventos u otro tipo de recursos en el IS.

IV. Validación de la arquitectura propuesta

Una de las últimas tecnologías que han surgido en el campo del Software Social es *OpenSocial*. OpenSocial [10] es una API social pública lanzada por Google a finales del 2007. Proporciona métodos de gestión sobre tres tipos de recursos relacionados con la información personal cobre los usuarios: sus contactos, las actividades que desarrollan en distintas plataformas SNS, y

soporte para datos persistentes.

El ejemplo de OpenSocial se integra perfectamente en la arquitectura propuesta. Los contactos son fuentes Atom, al igual que la relación de actividades. La publicación de actividades sigue el protocolo AtomPub. Por último, el soporte de datos persistentes tiene un gran parecido a la extensión OpenID Attribute Exchange.

Como parte de la validación de la arquitectura, se está trabajando en un plugin [11] para la plataforma de desarrollo web ágil Ruby on Rails. El plugin proporciona al marco de desarrollo toda la funcionalidad de autenticación, con varios métodos entre los que se encuentra OpenID, autorización, generación de contenidos y contactos. Además, el plugin proporciona las tecnologías descritas en los *Flujos de Intercambio* de Información entre IS y RS.

Este plugin se está usando para el desarrollo de varias plataformas SNS, entre las que se encuentra una aplicación del departamento dedicada a la gestión de sesiones de videoconferencia.

En este entorno se validará la arquitectura propuesta en este artículo.

V. Conclusiones

El artículo propone una arquitectura para solucionar el problema de la fragmentación de la información de los usuarios de plataformas SNS. La arquitectura propuesta se basa en OpenID, un protocolo de identidad distribuída basada en los usuarios. El Servidor de Identidades guarda la información autoritativa del usuario. Los Servidores de Recursos hacen uso del Servidor de Identidades para obtener mayor información acerca de los usuarios, así como para publicar nueva información sobre la actividad que el usuario realiza en la plataforma. Existen varias tecnologías disponibles para la implementación del flujo de información entre servidores. En este sentido, se está trabajando en un plugin para la plataforma de desarrollo ágil Ruby On Rails que implenta varias de estas tecnologías, el cual se está usando para el desarrollo de varias plataformas SNS que permitirán la validación de la arquitectura propuesta.

La arquitectura propuesta integra las últimas propuestas tecnológicas en el campo, como el API de Google OpenSocial.

Referencias

[1] D. Recordon, D. Reed, “OpenID 2.0: a platform for user-centric identity management”, *Proceedings of the second ACM workshop on Digital identity management*, pp. 11-16, 2006

[2] S. Brands, "The problem(s) with OpenID". Disponible en: http://idcorner.org/2007/08/22/the-problems-with-openid/

[3] Oauth, An open protocol to allow secure API authentication in a simple and standard method from desktop and web applications. Disponible en: http://oauth.net/

[4] D. Hart, J.Bufu, J.Hoyt, OpenID Attribute Exchange 1.0 – Final. Disponible en: http://openid.net/specs/openid-attribute-exchange-1_0.html

[5] R. Khare, Microformats: the next (small) thing on the semantic Web?, *Internet Computing, IEEE*, Volume 10, Issue 1, pp. 68-75, Jan.-Feb. 2006.

[6] M. Nottingham, R. Sayre, The Atom Syndication Format, RFC 4287, dec. 2005. Disponible en: http://tools.ietf.org/html/rfc4287

[7] D Brickley, L Miller, FOAF Vocabulary Specification, 2007. Disponible en: http://xmlns.com/foaf/spec/

[8] Semantic Interlinked Online Communities. Disponible en: http://sioc-project.org/

[9] J. Gregorio, B. de hOra, The Atom Publishing Protocol, RFC 5023, oct. 2007. Disponible en: http://tools.ietf.org/html/rfc5023

[10] OpenSocial Disponible en: http://code.google.com/apis/opensocial/

[11] CMSplugin. Disponible en: http://cmsplugin.rubyforge.org/