

Integración de aplicaciones web a través de sindicación de contenidos

María Soledad Romero, Mariano García Mattío, Pablo Andrés Vaca

Maestría en Ingeniería en Sistemas de Información / Universidad Tecnológica Nacional / Facultad Regional Córdoba

Maestro M. López esq. Cruz Roja Argentina – Ciudad Universitaria – Córdoba – Argentina C.P. (X5016ZAA)

Tel. (54 0351 5986034)

{romeroma.soledad, magm3333, vacapablo72}@gmail.com

Resumen

RSS (Really Simple Syndication) o sindicación de contenidos es una característica de muchos sitios Web. Su propósito es organizar y difundir la información disponible en las páginas Web. Usa el principio de suscripción, el usuario que desea recibir determinados contenidos efectúa una suscripción de los mismos. Una de las ventajas principales de la suscripción consiste en que el usuario Web no invierte tiempo en buscar novedades ya que le llegan aquellas a las que está suscripto.

Con el advenimiento de internet, las aplicaciones tendieron a adaptarse por lo que el mercado de software invirtió fuertemente en desarrollo de entornos de programación que permitieran acompañar esta tendencia. Por consiguiente se desarrollaron productos para generar aplicaciones web. Desde el mercado de los dispositivos móviles también se incorporó a las aplicaciones web como lo habían hecho los navegadores. Ha crecido la integración de servicios entre aplicaciones. El usuario final de una aplicación web puede ingresar a la misma y a partir de estar autenticado acceder a otros servicios.

El problema radica entonces en brindar al usuario final de una aplicación Web (accedida a través de un navegador o de un teléfono celular) que requiere autenticación, la posibilidad de enviarle información personalizada y privada a través de RSS a fin de combinar las ventajas de RSS con las funcionalidades de la aplicación web, incluso aquellas que dan soporte a

workflow, sujeto a selección de categorías y/o combinación de categorías de información.

Palabras clave: RSS, autenticación, aplicaciones web, feed, middleware, publicación, REST, Rome Project, suscripción, token, workflow.

Contexto

El proyecto forma parte de la propuesta de Tesis para la Maestría en Ingeniería de Sistemas de Información y es avalado por la empresa One Click e Business S.R.L. para el desarrollo de un prototipo que permita integrar aplicaciones web.

Introducción

El paradigma de la comunicación está cambiando y podría decirse que “el usuario ya no va en busca de la información, sino que es la información la que va en busca de él” [1].

Actualmente los formatos de intercambio de información entre computadoras han evolucionado y tienden a continuar en esa dirección facilitando la comunicación y la integración. Contamos con formatos como JSON [2] y XML, open software y librerías que permiten agregar valor específico a los proyectos como Rome [3], arquitecturas que permiten desarrollos independientes a los ya funcionando como los servicios web o web services [4] y estándares de transformación

como XLST [5] que facilitan la ornamentación del contenido que llega al usuario final.

Las distintas versiones de RSS han ido surgiendo para adaptarse a los cambios en la tecnología. Sólo son estándares las versiones RSS 1.0 y RSS 2.0 y Atom. Cabe destacar que Atom y RSS se diferencian por el formato de fecha utilizado para transferir información. El vocabulario empleado en RSS en general no es reusable en otros vocabularios de XML, mientras que el vocabulario de Atom fue diseñado para permitir elementos de vocabulario que sean reutilizados fuera del contexto del documento que se envía al syndicar contenidos con Atom. Si bien RSS y Atom tienen distinta sintaxis por ser desarrollos de proveedores independientes, conservan un núcleo de elementos de código que les son comunes.

Acompañando esta evolución de RSS han surgido trabajos de investigación. En "Developing Categorized News Items Using Feeds and Web Services" [6], Saha, Sajjanhar, Gao, Dew y Zhao describen cómo en la última década el masivo crecimiento de Internet ha provocado cambios masivos en la vida cotidiana de muchos hombres. Proponen el uso de sindicación de contenidos basado en servicios web o web services generando un prototipo basado en el estudio de algoritmos a fin de reducir los tiempos de distribución de contenido a partir de varios sitios Web.

En "Survey on application tools of Really Simple Syndication (RSS): A Case Study at Klang Valley" [7], Lee, Ghani y Huang analizan programas agregadores de RSS de tipo escritorio y Web, concluyen con los diez más usados de cada tipo. Dada la popularidad de los últimos tiempos exploran sobre las ventajas y desventajas. También ofrecen una perspectiva sobre la necesidad de trabajar con menos versiones de RSS.

En "A Novel Approach: Secure Information Notifying System using RSS Technology" [8], Preechaveerakul y Wichuta Kaewnopparat exploran sobre las posibilidades de syndicar contenido diferenciándolo entre contenidos

públicos y contenidos privados. Desarrollaron un modelo con la premisa de agregar y distribuir información pública y privada utilizando mecanismos de encriptación de datos para dar soporte a la seguridad de la información. Consideran casos de usuarios móviles y autenticación de usuario por lo que el modelo plantea una base de datos para la administración de los perfiles de usuarios. Además hacen referencia al uso de desarrollos producidos por el proyecto.

El proyecto Rome [3] se basa en un modelo abstracto de "suscripción o distribución de fuentes de noticias". Permite analizar cualquier formato de suscripción de noticias, incluyendo las variantes de RSS y Atom en este modelo. Puede convertir la representación de cualquier modelo de formato de suscripción de noticia a otro formato de suscripción de noticia. A nivel interno, define los modelos objeto intermedios de los formatos específicos de suscripción de noticias, incluyendo todas las variantes Atom y RSS. Para cada formato, hay una clase separada JDOM basada en XML que analiza un modelo intermedio. Ofrece convertidores para convertir entre los modelos de noticias (feed) intermedio y el modelo de fuente de distribución abstracto. Incluye además el manejo de los objetos involucrados en la autenticación de usuario.

Se entiende por autenticación a cualquier proceso por el cual se verifica que alguien es quien dice ser. La autorización es cualquier proceso por el cual a una persona se le permite estar donde quiere ir, o tener información que desea tener "Authentication and Authorization" [9]. La autorización requiere un proceso de autenticación previo.

Es importante destacar que existen distintos tipos de autenticación: en "Authentication Types" [10], Robert Moskowitz, describe los diferentes flujos. La autenticación existe para establecer la confianza entre dos partes, o entre las entidades de autenticación. Estas entidades consisten en una identidad y una llave. La autenticación se establece mediante la realización de una operación criptográfica entre las partes, identidades y claves. La operación de

cifrado o algoritmo de autenticación, establece la naturaleza de la confianza entre las partes. Una red de transporte o el flujo de autenticación proporcionan la conexión entre las partes para el algoritmo de autenticación. Algunos flujos de autenticación soportan un mecanismo de control de autenticación normalizado, para simplificar el soporte para múltiples algoritmos.

En la mayoría de los sistemas de autenticación las capas están estrechamente unidas y no se ha previsto ningún canal. En realidad, la aparición del canal de autenticación hizo posible un amplio espectro de soluciones de autenticación. En el modelo de capas, no todas las combinaciones son viables. El modelo de capas sirve de marco para el seguimiento del trabajo de clasificar varias combinaciones de flujos, canales, algoritmos, entidades y el modelo de confianza que proporcionan.

En "Authentication Types" [10], Robert Moskowitz señala que existen tres grupos de flujos de autenticación: unidireccional, bidireccional, y unidireccional acoplado. En el flujo de autenticación unidireccional una parte siempre debe iniciar la autenticación. En el flujo de autenticación bidireccional aunque una de las partes siempre inicia la autenticación, no hay un rol predeterminado de iniciador y el respondedor, y los roles se pueden invertir en cualquier momento. Hay una distinción clara entre la autenticación unidireccional y bidireccional. En la unidireccional, hay máquinas de estado exclusivas para el iniciador y el respondedor. En bidireccional, hay una máquina de estado que puede ser el iniciador de autenticación o respondedor. Esta máquina de estado sólo se encarga del cambio de sentido y de las condiciones de carrera (cuando ambas partes inician o ambos tratan de obligar a la otra a ser el respondedor). Unidireccional acoplado es una situación en la que hay dos flujos unidireccionales que se tienen que completar antes de que cualquiera de los flujos se considere terminado. El flujo de acoplamiento crea un ambiente muy similar a la autenticación bidireccional y se ocupa de la condición de carrera haciendo caso omiso de ella. No maneja cambios de dirección, esperando que cada parte

asuma plenamente tanto en el rol de iniciador como de respondedor. Ejemplos de protocolos de autenticación unidireccional incluyen a IEEE 802.1x, y TLS. Ejemplos de autenticación bidireccional incluyen IKE y HIP. IEEE 802.1aa se utiliza en flujos acoplados unidireccionales para la autenticación IEEE 802.11i AdHoc.

Observando las aplicaciones web, las autenticaciones y autorizaciones ya no son otorgadas a una sola aplicación. En "Authentication and Authorization in the Google Data Protocol" [11] se introduce como aplicaciones de terceros requieren a menudo un acceso limitado a la cuenta de Google del usuario para determinados tipos de actividad. Para asegurarse de que los datos de usuario están protegidos, todas las solicitudes de acceso deben ser aprobadas por el titular de la cuenta. El control de acceso tiene dos componentes, la autenticación y la autorización.

Los servicios de autenticación permiten a los usuarios iniciar sesión en su aplicación utilizando una cuenta de Google. Algunos servicios también permiten a los usuarios iniciar sesión con otra cuenta, como un inicio de sesión OpenID.

Los servicios de autorización permiten a los usuarios proporcionar acceso a los datos de sus aplicaciones que tiene almacenados en las aplicaciones de Google. Google se toma en serio la privacidad, y cualquier aplicación que requiera acceso a los datos de un usuario debe estar autorizado por el usuario [11].

Los servicios de autenticación y autorización a menudo se conocen colectivamente como auth.

Del análisis de las herramientas disponibles y el estado del arte [3] [12] surge la inquietud de crear un prototipo para publicación de contenido que tenga valor agregado a la sindicación de contenidos:

-Un token de autorización permite al usuario suscriptor acceder a la información a la que tiene acceso en otra aplicación web donde los datos residen.

-Cada usuario puede establecer qué tipo de información desea recibir, esto es, una o más categorías predefinidas (reclamos, tareas, oportunidades, informes).

-Para una categoría o tipo de información predefinida el usuario puede establecer condiciones en base a las cuales desea recibir la novedad o feed.

- Cada usuario puede asignar puntos (del 1 al 10 en cada categoría) lo que determina la importancia de la misma.

- En función de los parámetros mencionados (categorías, peso y condición) el usuario suscriptor podrá recibir un feed o novedad ajustado a sus preferencias y necesidades de información.

- La presentación del feed o novedad puede customizarse con la incorporación de diversidad de estilo aplicado al html del documento o feed que llega al usuario final.

La figura 1 resume el objetivo del uso del token y la generación de contenido.

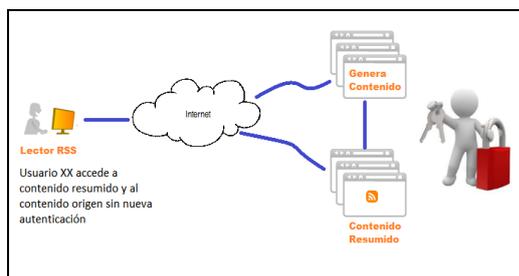


Figura 1 – RSS y Token

La arquitectura del middleware propuesto y el flujo que este middleware facilitará, está basado en la utilización de un token, el cual se constituye en la piedra angular del protocolo de aplicación. En la primera versión el token soporta información variable acerca del negocio, además de la información inherente al protocolo. El middleware permite establecer el algoritmo de cifrado del token mediante configuración.

Es deseable que en el futuro se pueda realizar el cambio de algoritmo de cifrado mediante un servicio estándar. También se requerirá trabajar en la ampliación de la arquitectura del protocolo de aplicación a fin de agregar características de expiración, validación de dominios, etc.

Líneas de investigación y desarrollo

Se investigan los trabajos realizados. Se trabaja en el diseño y desarrollo de un prototipo denominado pulse (en referencia a pulsar, pedir) utilizando Java como lenguaje de programación, web services, algoritmos de encriptación de claves y Apache Tomcat como servidor de aplicaciones. El prototipo cumple la función de publicar contenido en canales de suscripción asociados a RSS administrando la autenticación y autorización.

Resultados y objetivo

El objetivo principal es desarrollar un middleware [13] web que permita organizar y distribuir contenido sindicado en entornos Web, escritorio y móvil con la particularidad de incluir soporte para la autenticación relativa a la aplicación Web que contiene los datos que se desea distribuir. El middleware presta servicios mediante el estándar REST [2] para web services, se definirán interfaces claras de interacción con el middleware a efectos de poder consumir sus servicios desde cualquier lenguaje de programación.

Como objetivos secundarios cabe mencionar:

1. Explorar el estado actual del mercado en cuanto al uso de sindicación de contenidos se refiere.
2. Identificar alternativas de integración de aplicaciones web con sindicación de contenidos, en particular considerando el ingreso (usuario y contraseña) automático.
3. Desarrollar un proyecto de publicación de contenidos sindicados que sea independiente en cuanto a plataforma.
4. Proponer un esquema que contemple las distintas especificaciones de sindicación previamente identificadas.
5. Realizar pruebas de validación del esquema propuesto (distintos lectores).
6. Enriquecer el feed o contenido sindicado aportando alertas al usuario final.

7. Establecer una arquitectura y un flujo estándar para el uso de la sindicación integrada con aplicaciones web que requieren el delivery o entrega de contenido personalizado y basado en autenticación.

El resultado obtenido hasta el momento es la publicación de contenido sindicado desde una aplicación web a través del uso de los servicios del prototipo. Cabe destacar que se ha utilizado software open source para el desarrollo del 100% del trabajo.

Formación de Recursos Humanos

El presente trabajo forma parte de la investigación incluida en la Tesis de Maestría en Ingeniería en Sistemas de Información cuyo plan de tesis fue aprobado en la Comisión de Posgrado de la Universidad Tecnológica Nacional, Facultad Regional Córdoba. La carrera de Maestría en Ingeniería en Sistemas de Información que se dicta en la UTN – FRC está acreditada por Resolución N°: 1277/12 de la CONEAU. Se pretende, más allá del trabajo de tesis propiamente dicho, realizar como producto del trabajo de investigación, seminarios de capacitación y transferencia de conocimientos en la universidad. Esto va a permitir a alumnos y egresados adquirir nuevos conocimientos, nuevas habilidades. Por otro lado, el trabajo de tesis será el puntapié de otros proyectos de investigación, lo que contribuirá a formar nuevos profesionales en este campo.

Referencias

- [1] “¿Qué es RSS?”
<http://www.uma.es/secretariageneral/Feed/queesRSS.php>
 Universidad de Málaga – Secretaría General
 Fecha de último acceso: 10/11/2012
- [2] “RESTful Web Services”
 Autores: Leonard Richardson, Sam Ruby
 Editor: O’Reilly Media
 Edición: 2007
- [3] “The Rome Project”
<http://rometools.org/>
 Autores: Mark Woodman, Martin Kurz
 Fecha de último acceso: 23/11/2012
- [4] “Java Web Services: Up and Running”
 Autor: Martin Kalin
 Editor: O’Reilly & Associates Inc.
 ISBN: 978-0-596-52112-7
- [5] “XLS Transformation” – W3C Recommendation
<http://www.w3.org/TR/xslt>
 Editor: James Clark
 Fecha de último acceso: 23/11/2012
- [6] “Developing Categorized News Items Using Feeds and Web Services”
 Autores: Subrata Saha, Atul Sajjanhar, Shang Gao, Robert Dew - School of Information Technology Deakin University - Burwood, VIC 3125, Australia
 Ying Zhao – School of Information Science and Technology – Beijing University of Chemical Technology – Beijing, 100029, P. R. China
 2010 10th IEEE International Conference on Computer and Information Technology (CIT 2010)
- [7] “Survey on application tools of Really Simple Syndication (RSS): A Case Study at Klang Valley”
 Autores: TehPhoey Lee, Abdul Azim Abdul Ghani, Chan Yu Huang - Faculty of Management and Information Technology, UCSI, Malaysia.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4631980&ag=1
 Fecha de último acceso: 20/08/2011
- [8] “A Novel Approach: Secure Information Notifying System using RSS Technology”
 Ladda Preechaveerakul and Wichuta Kaewnopparat
 Computer Science Department
 Faculty of Science, Prince of Songkla University
 Hat Yai, Songkhla, Thailand
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5189906
 Fecha de último acceso: 20/08/2011
- [9] “Authentication and Authorization”
<http://httpd.apache.org/docs/2.2/howto/auth.html>
 Fecha de último acceso: 22/02/2013
- [10] “Authentication Types”
 Robert Moskowitz, ICSAlabs
 Autor: July 16, 2003 - Draft version 1.2
<http://www.ieee802.org/1/files/.../Authentication%20Types%20%201-2.doc>
 Fecha de último acceso: 22/02/2013
- [11] “Authentication and Authorization in the Google Data Protocol”
<https://developers.google.com/gdata/docs/auth/overview#OAuth2>
 Fecha de último acceso: 22/02/2013
- [12] “Developing feeds with RSS and Atom”
 Autor: By Ben Hammersley
 Editor: O’Reilly & Associates Inc.
 ISBN: 9780596008819
- [13] “A Survey of Middleware”, 18th International Conference on Computers and Their Applications, March 26-28, 2003, Honolulu, Hawaii.
<http://triton.towson.edu/~karne/research/middlew/surveym.pdf>
 Autores: Toni A. Bishop y Ramesh K. Karne
 Fecha de último acceso: 20/07/2012