

# ¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.  
 Ese apoyo que siempre quiso tener...

## 1. Desarrollo de componentes y proyectos a medida



## 2. Auditoría de código y recomendaciones de mejora

## 3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



## 4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,  
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)  
 Gestor de contenidos (Alfresco)  
 Aplicaciones híbridas

Tareas programadas (Quartz)  
 Gestor documental (Alfresco)  
 Inversión de control (Spring)

Control de autenticación y  
 acceso (Spring Security)  
 UDDI  
 Web Services  
 Rest Services  
 Social SSO  
 SSO (Cas)

JPA-Hibernate, MyBatis  
 Motor de búsqueda empresarial (Solr)  
 ETL (Talend)

Dirección de Proyectos Informáticos.  
 Metodologías ágiles  
 Patrones de diseño  
 TDD

BPM (jBPM o Bonita)  
 Generación de informes (JasperReport)  
 ESB (Open ESB)

**AdictosAlTrabajo**Terrakas 1x03  
¡¡Ya está en la web!!  
terrakas.comautentia  
Soporte a desarrollo informático  
Hosting patrocinado por  
**enredados**

Entra en Adictos a través de

E-mail

Contraseña

Entrar

Deseo registrarme  
Olvidé mi contraseña[Inicio](#) [Quiénes somos](#) [Formación](#) [Comparador de salarios](#) [Nuestro libro](#) [Más](#)» Estás en: [Inicio](#) [Tutoriales](#) Configurar múltiples contextos de seguridad en Spring Security 3.1.

Jose Manuel Sánchez Suárez

Consultor tecnológico de desarrollo de proyectos informáticos.

Puedes encontrarme en Autentia: Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE

[Ver todos los tutoriales del autor](#)

Fecha de publicación del tutorial: 2009-02-26

Tutorial visitado 2 veces [Descargar en PDF](#)

## Configurar múltiples contextos de seguridad en Spring Security 3.1.

### 0. Índice de contenidos.

- 1. Introducción.
- 2. Entorno.
- 3. Configuración.
  - 3.1. Configuración de la seguridad del contexto de Web Services.
  - 3.2. Configuración de la seguridad del área de clientes.
- 4. Referencias.
- 5. Conclusiones.

### 1. Introducción

A partir de la versión 3.1, Spring Security proporciona soporte para la configuración de múltiples contextos de seguridad, cada uno apuntando a su propio manager de autenticación.

En un frontal de aplicación, en una web pública, quizás tengamos la necesidad de securizar un área de clientes y, adicionalmente, proporcionar servicios web del tipo que elijamos (REST por ejemplo) securizados, con otra fuente de autenticación distinta a la del área de clientes. Ambas, dentro de la misma aplicación, empaquetadas dentro del mismo war, por ejemplo:

- <http://www.mi-comercio-online.com/>: es público,
- <http://www.mi-comercio-online.com/clientes/>: es privado con autenticación por formulario y usuario y contraseña previo registro,
- <http://www.mi-comercio-online.com/ws/rss/>: es privado con autenticación básica y un usuario y contraseña genérico o distinto al de registro de usuarios.

En este tutorial vamos a revisar cómo podemos realizar dicha configuración con el soporte de Spring Security.

Se presuponen ciertos conocimientos previos sobre Spring Security, en adictos al trabajo ya hemos publicado varios tutoriales sobre la materia.

### 2. Entorno.

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil MacBook Pro 15' (2.4 GHz Intel Core i7, 8GB DDR3 SDRAM).
- Sistema Operativo: Mac OS X Lion 10.7.4
- Spring Security 3.1.0.M1.

### 3. Configuración.

El contenido de nuestro applicationContext-security.xml tendrá un contenido como el que sigue:

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3     xmlns:sec="http://www.springframework.org/schema/security" xmlns:xsi="http://www.w3.org/2001/x
4     xmlns:context="http://www.springframework.org/schema/context"
5     xsi:schemaLocation="http://www.springframework.org/schema/beans
6         http://www.springframework.org/schema/beans/spring-beans.xsd
7         http://www.springframework.org/schema/security
```

### Catálogo de servicios Autentia



### Síguenos a través de:



### Últimas Noticias

- » [Autentia patrocina al Club KiteSurf Centro](#)
- » [Autentia patrocina el I Torneo Voley Playa Terrakas](#)
- » [Autentia colabora con la ONG Proyecto Ciclista Solidario](#)
- » [Curso de Kanban Core en Madrid con Masa K. Maeda](#)
- » [Failure demand](#)

[Histórico de noticias](#)

### Últimos Tutoriales

- » [Transiciones y animaciones con CSS3](#)
- » [Configuración y tuning de servidores de producción](#)
- » [Revisión de jBPM5](#)
- » [Database MessageSource: obtener los literales de una base de datos](#)
- » [Comparando diferencias entre ficheros con java-diff-utils](#)

### Últimos Tutoriales del

Impulsores Comunidad ¿Ayuda?

0 personas han traído clicks a esta página

sin clicks + + + + + + +

powered by [karmacrac](#)

```

8 http://www.springframework.org/schema/security/spring-security.xsd
9 http://www.springframework.org/schema/context http://www.springframework.org/schema/context
10
11 <!-- REST services security -->
12 <sec:http auto-config="false" authentication-manager-ref="wsAuthenticationManager"
13 use-expressions="true" pattern="/ws/**">
14 <sec:http-basic />
15 <sec:intercept-url pattern="/ws/**"
16 access="hasRole('ROLE_ADMINISTRATION')" />
17 </sec:http>
18
19 <sec:authentication-manager id="wsAuthenticationManager">
20 <sec:authentication-provider>
21 <sec:user-service>
22 <sec:user name="{security.front.ws.userName}" password="{security.front.ws.passw
23 </sec:user-service>
24 </sec:authentication-provider>
25 </sec:authentication-manager>
26
27 <!-- CUSTOMER zone security -->
28 <sec:http auto-config="false" access-denied-page="/error.xhtml" authentication-manager-ref="cu
29 use-expressions="true">
30
31 <sec:intercept-url pattern="/clientes/**"
32 access="hasRole('ROLE_CUSTOMER')" />
33
34 <sec:form-login login-page="/acceso" authentication-failure-url="/acceso?error=1"
35 login-processing-url="/acceder" default-target-url="/clientes" always-use-default-targ
36
37 <sec:logout invalidate-session="true" logout-url="/salir"
38 logout-success-url="/" />
39
40 </sec:http>
41
42 <sec:authentication-manager id="customerAuthenticationManager">
43 <sec:authentication-provider >
44 <sec:password-encoder hash="md5" base64="true" >
45 <sec:salt-source system-wide="{password.salt.encoder}" />
46 </sec:password-encoder>
47 <sec:jdbc-user-service data-source-ref="dataSource"
48 users-by-username-query="select email as username, clave as password, activo as er
49 authorities-by-username-query="select email as username, 'ROLE_CUSTOMER' as author
50 />
51 </sec:authentication-provider>
52 </sec:authentication-manager>
53
54 </beans>

```

## Autor

» Uso de componentes JSF de gráficos con el soporte de Primefaces.

» Uso de un componente JSF de subida de ficheros al servidor con el soporte de Primefaces.

» Creación de una anotación de validación personalizada para Bean Validator.

» Peticiones GET en JSF2: mapear parámetros y gestionar eventos de página.

» Cómo incluir un botón personalizado para nuestro CMS en la barra de menú de TinyMCE

## Últimas ofertas de empleo

2011-09-08  
Comercial - Ventas - MADRID.

2011-09-03  
Comercial - Ventas - VALENCIA.

2011-08-19  
Comercial - Compras - ALICANTE.

2011-07-12  
Otras Sin catalogar - MADRID.

2011-07-06  
Otras Sin catalogar - LUGO.

 Jose Manuel Sánchez  
sanchezsuaresj

jmbear Me encanta la iniciativa de @sueldospublicos por su seriedad y arrojo. ¡Bravo! Hacen falta más como ésta  
4 days ago · reply · retweet · favorite

rcanalesmora Mañana empieza del torneo de volej playa @terrakas. masvoleyplaya.com Fotito del cartel con nuestros guaperas yfrog.com/oOrtemp  
6 days ago · reply · retweet · favorite

rcanalesmora Viendo métricas de la aplicación que cerramos mañana: 89% de cobertura de código en pruebas automáticas. Ahora vas y lo cascas :- ) #tdd  
14 days ago · reply · retweet · favorite

sanchezsuaresj un buen uso para el  
 Join the conversation

### 3.1. Configuración de la seguridad del contexto de Web Services.

Debemos destacar:

- línea 12: la ubicación de la configuración de los dos contextos de seguridad (<sec:http) no es casual, debemos configurar primero el menos restrictivo; la configuración de los contextos será secuencial, con lo que el primero debe configurar el área más restringida, en nuestro caso /ws/\*\*, el resto de urls pasará al segundo contexto. Usamos la referencia a un manager de autenticación propio para este contexto,
- línea 14: la configuración de nuestro contexto de seguridad para los web services será basic, mostrará un diálogo del navegador para la petición de usuario y contraseña, esto nos permitirá acceder a los servicios web securizados de forma programática,
- línea 15: la url que intercepta debe tener la misma raíz que el pattern de configuración del contexto de seguridad, línea 12,
- línea 19: hemos configurado un manager de autenticación para el contexto de seguridad que utiliza una configuración simple de usuarios y contraseñas; en concreto un solo usuario cuyo nombre y contraseña se obtiene de un "property place holder configurer" de spring y se asigna manualmente el rol de administrador. Para este ejemplo nos basta, pero podríamos utilizar un manager de autenticación cuan complejo necesitemos.

### 3.2. Configuración de la seguridad del área de clientes.

Debemos destacar:

- línea 28: usamos la referencia a un manager de autenticación propio para este contexto y no usamos un patrón de capturas de url, con lo que será el contexto de seguridad por defecto, que recibirá las urls que no intercepten los contextos configurados previamente en el fichero,
- línea 31: interceptores de urls con el rol asociado a cada una de ellas,
- línea 34: configuración de la seguridad a través de un login de formulario, y las urls de redirección,
- línea 37: configuración del path de salida o logout,
- línea 42: configuración del manager de autenticación para el área de clientes, en nuestro caso establece las consultas a realizar para recuperar usuario y roles y un nivel de encriptación haciendo uso de un salt fijo, para que las contraseñas encriptadas no sean predecibles y susceptibles de un ataque de diccionario.

## 4. Referencias.

- <https://jira.springsource.org/browse/SEC-1171>
- <https://jira.springsource.org/browse/SEC-1847>

## 5. Conclusiones.

Seguimos usando de forma extensiva todos los módulos de Spring para configurar, en este caso y de forma sencilla, la autenticación y autorización de nuestras aplicaciones web.

Cada vez más, las aplicaciones web proporcionan servicios web ligeros, de tipo REST, para permitir la consulta de información o proporcionar otro tipo de interacciones. Con lo visto en este tutorial, podemos proporcionarlos securizados con un nivel de autenticación distinto al de otras áreas de seguridad de la aplicación.

Con todo ello, evitamos desplegar dos aplicaciones distintas o reutilizar un manager de autenticación de clientes para un servicio que puede que no se proporcione en exclusiva o solamente para ellos.

Un saludo.

Jose

jmsanchez@autentia.com

### A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)

### Por favor, vota +1 o compártelo si te pareció interesante

Share |

0

Animáte y coméntanos lo que pienses sobre este **TUTORIAL**:

» [Regístrate](#) y accede a esta y otras ventajas «



Esta obra está licenciada bajo licencia [Creative Commons](#) de Reconocimiento-No comercial-Sin obras derivadas 2.5

Copyright 2003-2012 © All Rights Reserved | [Texto legal y condiciones de uso](#) | [Banners](#) | [Powered by Autentia](#) | [Contacto](#)

W3C XHTML 1.0

W3C CSS

XML RSS

XML ATOM