Com securitzar els mètodes de negoci

A qui va dirigit

Aquest how-to va dirigit a tots aquells desenvolupadors encarregats de la creació de la capa de negoci d'una aplicació Canigó.

En aquest document es presenta el procediment per configurar Canigó de forma que es restringeixi l'accés al mètodes de negoci als perfils adequats.

Versió de Canigó

Els passos descrits en aquest document poden ser d'aplicació a les **versions 2.x de Canigó**.

Requeriments

Tal i com Canigó promou, les classes que ofereixen les funcionalitats de negoci (Business Objects) han d'implementar una interfície en que es descriuen els mètodes de negoci.

Context

Es una bona pràctica en el desenvolupament d'aplicacions J2EE en general i Canigó en particular el separar les diferents capes de l'aplicació.

La securització de la capa de negoci és una pràctica molt recomanable ja que garanteix que no es podran fer accessos a funcionalitats restringides per usuaris no autoritzats.

Independentment de la securització de la capa de negoci és recomanable realitzar la securització de la capa de presentació de forma que no es presentin als usuaris opcions a les que no tenen accés.

Per la natura distribuïda de les aplicacions web, un usuari malintencionat o una capa de presentació amb problemes de disseny poden generar peticions al servidor per les que l'usuari no està autoritzat. A Canigó es realitza una primera verificació (filtre web que implementa la seguretat de la capa de presentació) i una segona verificació (AOP que verifica la seguretat de mètodes).

La securització de la capa de negoci juntament amb un disseny desacoblat de la mateixa fa que en un futur es puguin afegir o reemplaçar capes a les que dona servei (client pesat, webservice, etc).

Com securitzar els mètodes de negoci

Procediment

0. Valorar si cal aplicar seguretat a la capa de negoci

Tal i com s'ha explicat en l'apartat de context la securització de la capa de negoci és una bona pràctica i és recomanable per la majoria d'aplicacions. Tot i això queda a criteri de l'equip de disseny de l'aplicació la elecció de les pràctiques de seguretat idònies pels requeriments i context de l'aplicació.

1. Configuració de la matriu de permisos

En el disseny de l'aplicació ha d'existir una matriu de permisos que indiqui quins rols poden executar les diferents funcionalitats de negoci (casos d'ús).

Exemple:

	Rol	**Usuari**	**Gestor**	**Administrador**
Cas d'us				
Consultar usuari		No	Si	Si
Consulta propi usuari		Si	Si	Si
Esborrar usuari		No	No	SI

En alguns casos i per raons de disseny la granularitat pot inferior al cas d'ús i referir a un mètode d'alguna classe del domini que no implementi un cas d''us, i per tant no visible des de la capa de presentació.

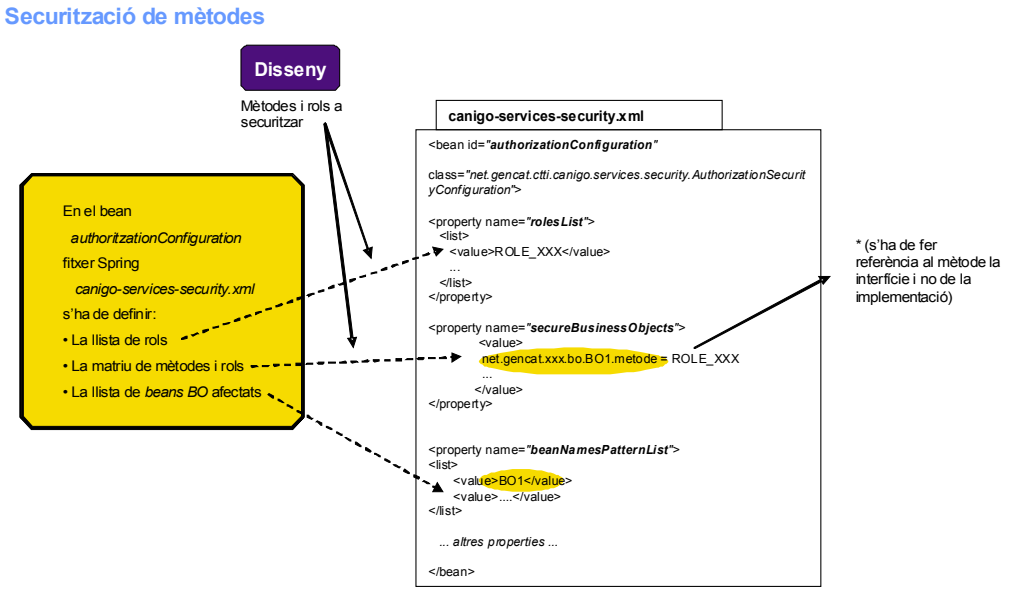
Tant si es tracta d'un mètode de negoci o un mètode d'una classe més interna, el sistema proposat en aquest document és d'aplicació. El requeriment és que la classe a securitzar s'exposi com un bean d'Spring i sigui explotada com a tal.

El mètode proposat per Canigó és genèric i pot ser d'aplicació a la majoria d'aplicacions. Si les necessitats de seguretat descrites en el disseny no poden ser assolides es recomana complementar el mètode proposat amb lògica de negoci o buscar alternatives tecnològiques alineades amb Canigó.

Un cop s'ha identificat la matriu de permisos s'haurà de reflectir en el fitxer de configuració canigo-*services-security.xml,* en la configuració del bean "authorizationConfiguration":

- pas 1: Afegir la llista de rols a la propietat "rolesList"
- pas 2: Afegir la llista de *beans* que implementen els mètodes a securitzar a la propietat "beanNamesPatternList". Aquesta llista ha de contenir els noms exactes amb que apareixen configurats a Spring.
- pas 3: Afegir la llista de parells (mètode, rol) autoritzats a la propietat "secureBusinessObjects". En aquest pas el mètode al que es fa referència ha de ser el de la interfície i no el de la implementació.

Com securitzar els mètodes de negoci



- Configuració de la matriu de permisos -

Important. Com a possible font de problemes, es recomana revisar la coincidència exacte de noms a l'hora de referir a mètodes, classes i beans. La implementació de proves és una pràctica excel·lent per assegurar que l'objectiu desitjat ha estat assolit.

2. Revisió de la infraestructura de Canigó

Per tal que la configuració descrita en el pas previ sigui tinguda en compte cal verificar que Canigó està correctament configurat:

A) Verifiqui que el servei de seguretat està habilitat:

En la càrrega del context d'Spring s'ha inclòs el fitxer de configuració del servei:

<import resource="canigo-services-security.xml"/>

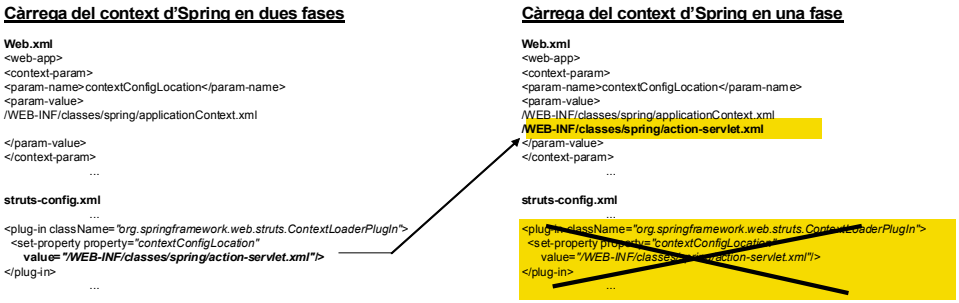
B) Es fa la carrega del context d'Spring en una única fase.

Verificar que en el fitxer struts-config.xml no s'està fent servir el plugin:

org.springframework.web.struts.ContextLoaderPlugIn

per carregar part del context. Si l'aplicació conté aquest plugin s'ha de comentar i passar les referències als fitxers d'Spring definides contextConfigLocation

Com securitzar els mètodes de negoci



C) Verificar que el fitxer de configuració conté la següent inclusió:

<import resource="classpath:/spring/acegi-beans.xml" />

3. Revisió del log de l'aplicació

Si tot ha anat be en el log de l'aplicació apareixerà un missatge com el següent per cada securització realitzada, on es detalla la signatura del mètode afectat i el rol.

```
INFO  [main] net.sf.acegisecurity.intercept.method.MethodDefinitionMap - Adding secure
method [public abstract void
net.gencat.ctti.canigo.samples.prototip.model.bo.CategoryBO.saveOrUpdate(net.gencat.ctti.can
igo.samples.prototip.model.Category) throws java.lang.Exception] with attributes
[[ROLE_ADMIN]]
```