

## Utilització de Proxy en els serveis del Framework Canigó

### A qui va dirigit

Aquest how-to va dirigit a tots aquells que hagin de desenvolupar una aplicació Canigó que:

- s'hagi de desplegar a la xarxa XCAT
- accedeixi a serveis hostatjats fora d'aquesta xarxa

### Versió de Canigó

Els passos descrits en aquest document són d'aplicació a les versions de Canigó a partir de la 2.3.3

### Introducció

Les aplicacions desplegades en la xarxa XCAT poden tenir algunes restriccions pel que fa a l'accés a recursos externs a aquesta Xarxa.

Aquest how-to presenta algunes consideracions a tenir en compte en l'accés a serveis hostatjats en màquines fora de la xarxa, que pot veure's afectat per la existència d'un proxy.

En funció de la configuració concreta que s'hagi fet de l'entorn on es desplega l'aplicació, pot ser necessari que les peticions http sortints cap a l'exterior indiquin explícitament el proxy a utilitzar i continguin les dades d'autenticació (usuari / password) per al proxy.

Un exemple concret d'aquest tipus d'accés és el que fa el connector amb Catcert per realitzar les validacions de signatures amb el servei PSIS.

### Servei FTP via Proxy

El servei de FTP de Canigó està basat en Apache Commons Net. Entre els FAQs de Apache Commons Net ( <http://wiki.apache.org/commons/Net/FrequentlyAskedQuestions> ) n'hi ha un "*Does FTPClient support FTP connections through an FTP proxy server?*" que explica com configurar la utilització d'un proxy amb aquesta llibreria.

També és rellevant la documentació disponible en <http://java.sun.com/j2se/1.4.2/docs/guide/net/properties.html> sobre les propietats a configurar per a la utilització d'un Proxy:

- socksProxyHost
- socksProxyPort
  
- ftp.proxyHost
- ftp.proxyPort
- ftp.nonProxyHosts

### Servei de WebServices via Proxy

El servei de WebServices de Canigó te dos variants segons es configuri d'una manera o una altra.

1. El primer escenari es tracta dels WebServices integrats amb Canigó a partir d'una URL (definint la propietat "serviceURL"), i que fa servir XFire. Aquest escenari es configura en el bean WebServicesService, que accepta un conjunt de paràmetres per configurar la utilització d'un Proxy:

- **useHttpProxy**: true o false, habilita la configuració i us d'un proxy
- **proxyHost**: si el proxy està habilitat, defineix el host o adreça ip

## Utilització de Proxy en els serveis del Framework Canigó

- **proxyPort**: si el proxy està habilitat, defineix el port
- **userName**: si el proxy està habilitat, defineix l'usuari
- **password**: si el proxy està habilitat, defineix la clau d'accés

```
<bean name="WebServicesService"
class="net.gencat.ctti.canigo.services.webservices.impl.WebServicesServiceImpl">
  <property name="importedInterfaces">
    <list>
      <bean parent="ImportedInterfaceDefinition">
        <property name="name" value="testService" />
        <property name="serviceURL"
value="http://localhost:8080/canigo-proves/testService" />
        <property name="localInterface"
value="net.gencat.canigo.proves.TestServicePortType" />
      </bean>
    </list>
  </property>
  <property name="useHttpProxy" value="true" />
  <property name="proxyHost" value="192.168.47.162" />
  <property name="proxyPort" value="8080" />
  <property name="userName" value="timbaler" />
  <property name="password" value="delbruc" />
</bean>
```

Per exemple:

2. El segon escenari correspon a la configuració de la integració amb Canigó a partir d'una implementació *Axis* (definint la propietat "serviceLocator").

La configuració és més complexa que al cas anterior ja que haurem de modificar les classes generades amb *Axis*. El primer pas que haurem de fer serà configurar el servei de *WebService* a la vostra aplicació. Per exemple:

```
<bean name="WebServicesService"
class="net.gencat.ctti.canigo.services.webservices.impl.WebServicesServiceImpl">
  <property name="importedInterfaces">
    <list>
      <bean parent="ImportedInterfaceDefinition">
        <property name="name" value="testService" />
        <property name="serviceLocator" ref="testWebServiceBean"/>
        <property name="localInterface"
value="net.gencat.canigo.proves.TestServicePortType" />
      </bean>
    </list>
  </property>
</bean>
```

En negreta ressaltem que la classe *Locator* es tracta d'una referència a un bean. Aquest serà el que contingui la configuració del proxy tal i com s'indica a la següent configuració d'exemple:

```
<bean name="testWebServiceBean" class="net net.gencat.canigo.proves.TestServiceLocator">
  <property name="testWSPortAddress" value="${WS_URL_Locator}"/>
  <property name="configProxyHost" value="classpath:${PROXY_HOST}"/>
  <property name="configProxyPort" value="classpath:${PROXY_PORT}"/>
  <property name="configProxyUser" value="classpath:${PROXY_USER}"/>
  <property name="configProxyPassword" value="classpath:${PROXY_PASSWORD}"/>
  <property name="configTransferChunked" value="${TRANSPORT_CHUNKED}"/>
  <property name="logService" ref="loggingService"/>
</bean>
```

## Utilització de Proxy en els serveis del Framework Canigó

Com podem observar, les propietats són variables, aquestes variables fan referència a les propietats que estan carregades pel Servei de Configuració a través del fitxer de propietats *TestWSLocator.properties* que crearem a la carpeta *spring/webservices/* de la nostra aplicació i que tindrà el següent contingut:

```
# URL del WS
WS_URL_Locator=<url_del_webservice>
# Dades de configuració del proxy. Si es defineix Host i port es configurarà l'us del proxy.
L'usuari i password son opcionals.
PROXY_HOST=hola
PROXY_PORT=8080
PROXY_USER=pepe
PROXY_PASSWORD=gotera
# Opció de configuració del transport HTTP, que permet deshabilitar la opció "Chunked" que
sembla que pot ocasionar problemes en algunes ocasions.
TRANSPORT_CHUNKED=false
```

L'aplicació a través del servei de configuració carregarà el fitxer i per tant, també hem d'afegir el fitxer de propietats a la configuració del servei de configuració (*canigo-services-configuration.xml*):

```
<bean id="configurationService" class="net.gencat.ctti.canigo.services.configuration.
springframework.beans.factory.config.HostPropertyPlaceholderConfigurer">
  <property name="basePropertyFiles">
    <list>
      ...
      <value>classpath:webservices/TestWSLocator.properties</value>
      ...
    </list>
  </property>
</bean>
```

Un cop haguem fet aquestes accions, hem de modificar les classes generades a partir d'Axis, les classes *Locator* i *Stub*:

- **TestServiceLocator:**

Afegirem les propietats que hem incorporat a la configuració del *WebService* (*testWSPortAddress*, *configProxyHost*, *configProxyPort*, *configProxyUser*, *configProxyPassword*, *configTransferChunked* i *logService*) amb els seus corresponents *getter's* i *setter's*.

També haurem de modificar les crides al constructor de la classe *Stub* ja que, com veurem més endavant, el modificarem per a que contempli els nous atributs de la configuració del proxy.

- **TestServiceBindingStub:**

Afegirem, com en el cas anterior, les noves propietats afegides per a la configuració del proxy, en aquest cas no cal que s'anomenin de la mateixa manera que al fitxer de propietats ja que el seu valor vindrà donat pel nou constructor.

Per a mantenir la compatibilitat amb altres configuracions, afegirem un nou constructor a la classe en el que donarem valor a les noves propietats. Una possible capçalera podria ser la següent:

```
public TestServiceBindingStub(
    java.net.URL endpointURL, javax.xml.rpc.Service service, LoggingService logService,
    String testWSPortAddress, String configProxyHost, Integer configProxyPort,
    String configProxyUser, String configProxyPassword, Boolean configTransferChunked)
    throws org.apache.axis.AxisFault;
```

Els atributs nous que hem afegit a la capçalera estan marcats amb negreta.

## Utilització de Proxy en els serveis del Framework Canigó

La classe *Stub* serà l'encarregada d'accedir a totes les funcionalitats publicades al *WSDL* i per tant, exposarà tots els mètodes que ens permetran accedir a aquests serveis remots a través d'Internet i en aquest cas, del proxy. Per tal de configurar la connexió amb el proxy, ens assegurarem que abans de fer la primera i només a la primera crida a un mètode remot, s'haurà executat el codi següent:

```
HttpTransportProperties.ProxyProperties proxyProperties = new HttpTransportProperties().new
ProxyProperties();
proxyProperties.setProxyName(configProxyHost);
if (configProxyPort != null) {
    proxyProperties.setProxyPort(configProxyPort.intValue());
} if (
    configProxyUser != null && configProxyPassword != null) {
    proxyProperties.setUsername(configProxyUser);
    proxyProperties.setPassword(configProxyPassword);
}
digitalSignatureServiceStub._getServiceClient().getOptions().setProperty(org.apache.axis2.trans
port.http.HTTPConstants.PROXY, proxyProperties);
```

Aquest codi crea un objecte *ProxyProperties* que ens permet encapsular les dades de connexió a través del proxy, un cop s'han emmagatzemat les dades guardarem l'objecte al *ServiceClient* que es troba al paquet d'*Axis2*.

Si es vol utilitzar la opció **TRANSPORT\_CHUNKED** cal afegir també la següent inicialització:

```
if (configTransferChunked != null) {
    if (configTransferChunked.booleanValue()) {
        digitalSignatureServiceStub._getServiceClient().getOptions()
        .setProperty(MessageContextConstants.CHUNKED, Constants.VALUE_TRUE);
    } else {
        digitalSignatureServiceStub._getServiceClient().getOptions()
        .setProperty(MessageContextConstants.CHUNKED, Constants.VALUE_FALSE);
    }
}
```

Per a fer aquest exemple ens hem basat a la configuració del connector amb *Catcert*, us podeu baixar les fonts a la següent ruta:

<http://canigo.ctti.gencat.net/repository/maven2/canigo/connectors/canigo-connectors-catcert/2.3.10-SNAPSHOT/>

3. El tercer escenari correspon a la configuració de la integració amb Canigó a partir d'una implementació **Axis / utilitzant AxisProperties**.

La configuració és molt senzilla ja que només s'hauran d'informar propietats la primera vegada que es configura el servei. Per exemple:

```
AxisProperties.setProperty("http.proxySet", "true");
AxisProperties.setProperty("http.proxyPort", Configuration.getProxyPort());
AxisProperties.setProperty("http.proxyHost", Configuration.getProxyHost());
```

*AxisProperties* es recolza amb la llibreria *commons-discovery*, per a mantenir les propietats, i aquestes propietats **només afecten a l'aplicació** a diferència que les *System.properties* que afecten a totes les aplicacions de la instància.

### Configuració de l'accés a PSIS via Proxy

El connector amb *Catcert* utilitza internament *axis2* i la llibreria *httpclient* per l'accés als Web Services. Aquesta llibreria permet activar les traces per monitorar el tràfic http (peticions i respostes) afegint a la configuració de *log4j*

```
<category name="httpclient">
    <level value="debug"/>
</category>
```

## Utilització de Proxy en els serveis del Framework Canigó

Aquesta llibreria (o, per ser més exactes les llibreries del JRE) accepten dos paràmetres per configurar l'ús d'un proxy, que es poden definir a l'arrancar la màquina virtual

```
-Dhttp.proxyHost=192.168.47.162 -Dhttp.proxyPort=8080
```

L'ús d'aquests paràmetres afecta a la màquina virtual Java i, per tant, al servidor d'aplicacions i a totes les aplicacions. Això pot provocar conflictes entre les necessitats de diferents aplicacions.

Per tal de permetre que cada aplicació pugui configurar-se de forma independent de les altres, s'han afegit un conjunt de paràmetres a la configuració del connector de Catcert (disponibles a partir de la versió 2.3.3 de Canigó), en l'arxiu ValidateWSLocator.properties, que queda com

```
# URL PSIS test
WS_URL_Locator=http://psisbeta.catcert.net/psis/catcert-test/dss
WS_URL_Locator_PDF=http://psisbeta.catcert.net/psis/catcert-test/dsspdf
# URL PSIS Producció
# WS_URL_Locator=http://psis.catcert.net/psis/catcert/dss
# WS_URL_Locator_PDF=http://psis.catcert.net/psis/catcert/dsspdf
# Dades de configuració del proxy. Si es defineix Host i Port es configurarà l'ús del proxy
PROXY_HOST=192.168.47.162
PROXY_PORT=8080
# L'usuari i password son opcionals
#PROXY_USER=timbaler
#PROXY_PASSWORD=delbruc
# Opció de configuració del transport HTTP, que permet deshabilitat la opció "Chunked"
# que sembla que pot ocasionar problemes en algunes ocasions
TRANSPORT_CHUNKED=false
```

Els paràmetres afegits son

- PROXY\_HOST i PROXY\_PORT per indicar el nom del node o l'adreça IP i el port on hi ha el proxy. Si no es defineixen aquests paràmetres, no es configura l'ús explícit d'un Proxy.
- PROXY\_USER i PROXY\_PASSWORD necessaris només si l'accés al Proxy requereix autenticació.

S'ha afegit un paràmetre més (TRANSPORT\_CHUNKED) per permetre deshabilitar la opció HTTP 1.1 Transport Chunked, que podria ocasionar problemes amb alguns tipus de proxies / servidors web. Per defecte està habilitada (true).

### Configuració de l'accés a altres WebServices

Si s'ha desenvolupat un client de Web Services sense utilitzar el servei de Web Services de Canigó, la utilització d'un proxy caldrà configurar-la en el client del web service que s'utilitzi i la forma de fer-ho és dependent de la tecnologia amb la que s'hagi desenvolupat el client.

Si el client està desenvolupat amb Axis2, pot veure's un exemple de la forma de configurar-lo en el codi font del connector Canigó amb Catcert (a partir de la versió 2.3.3). Concretament, en el mètode setServiceClientOptions de la classe ValidateWSBindingStubPSIS.

### Adreça del proxy

En el moment de realitzar aquest document (febrer 2009), l'adreça del proxy es 192.168.47.162.

En el moment de realitzar aquest document (febrer 2009), aquesta adreça es comuna pels entorns d'integració, preproducció i producció.

## Utilització de Proxy en els serveis del Framework Canigó

En el moment de realitzar aquest document (febrer 2009), es necessari sol·licitar la configuració de regles del firewall per tal de poder accedir a aquesta màquina.

En el moment de realitzar aquest document (febrer 2009), el proxy no requereix usuari paraula d'accés.

Abans de prendre aquesta informació com a referència ho haure de contrastar amb l'enllaç tecnològic del vostre projecte.