

GUÍA DE CONFIGURACIÓN DE TOMCAT Y APACHE

Contenido

INTRODUCCIÓN.....	- 2 -
REQUISITOS.....	- 2 -
ARQUITECTURA.....	- 2 -
CONFIGURACIÓN SERVIDOR OPENCMS.....	- 2 -
TOMCAT y OPENCMS.....	- 2 -
SERVIDOR WEB / APACHE.....	- 3 -
CONFIGURACION FRONT-APACHE.....	- 5 -

INTRODUCCIÓN

Esta guía indica la configuración necesaria para tener configurado con multisite OpenCms con un servidor web apache sirviendo las peticiones. Esto nos permitirá servir contenidos estáticos directamente de disco y eliminar de las Url el nombre del contexto de aplicación y el del servlet (/opencms/opencms/)

REQUISITOS

Esta guía se basa en la configuración en Windows. Para explicar la configuración se parte de los siguientes supuestos:

1. Tenemos Apache2 instalado. Haremos referencia a la carpeta de instalación como: [\[Install-Path-Apache2\]](#)
2. Tenemos instalado Tomcat. en la carpeta [\[Install-Path-Tomcat\]](#)
3. Tenemos desplegado OpenCms bajo el contexto **opencms**, por tanto está instalado bajo la ruta **[Install-Path-Tomcat]\webapps\opencms**

ARQUITECTURA

En esta implementación Apache y Tomcat se encuentran configurados en la misma máquina y disponemos de una segunda máquina con un Apache que hará las veces de Front de OpenCms

CONFIGURACIÓN SERVIDOR OPENCMS

TOMCAT y OPENCMS

Para implementar la arquitectura es necesario realizar algunas modificaciones en la configuración de un Tomcat estándar:

Las modificaciones a realizar son:

1. Editar el fichero **[Install-Path-Tomcat]\conf\context.xml** y añadir el atributo **sessionCookiePath="/"** a la etiqueta **Context**:

```
<Context sessionCookiePath="/">
...
</Context>
```

Esto es necesario ya que por defecto las sesiones se guardan usando la ruta del Servlet y el contexto de aplicación, y nuestro objetivo es eliminarlo, por tanto debemos forzar las sesiones a que usen la raíz del sitio.

2. Configuración de Sites en OpenCms:

Cada site dentro de OpenCms corresponderá con una Url, por otro lado es necesario disponer de una url para el acceso al Workplace de OpenCms. Para realizar esta configuración es necesario editar el fichero **[Install-Path-Tomcat]\webapps\opencms\WEB-INF\config\opencms-system.xml** y agregar el dominio del sitio a la configuración de sites de OpenCms.

Supongamos que vamos a configurar los siguientes dominios:

- Sitio Web: **test.principal.com**
- Sitio Worklace Opencms: **opencms.principal.com** (Podría ser el nombre de la máquina)

```
<sites>
  <workplace-server>http://opencms.principal.com </workplace-
server>
  ...
  <site server="http://test.principal.com" uri="/sites/default/"
/>
</sites>
```

3. Ahora es necesario configurar Opencms para que gestione correctamente los distintos enlaces según el site, como vamos a eliminar la ruta al Servlet y a la aplicacion es necesario indicarle a OpenCms que no genere estas rutas al construir los enlaces. Para ello es necesario editar el fichero **[Install-Path-Tomcat]\webapps\opencms\WEB-INF\config\opencms-importexport.xml** para modificar la forma en la que se generan las rutas. Debemos localizar las etiquetas **<rfs-prefix>** y **<vfs-prefix>** editarla para que quede así:

```
...
<rendersettings>
  <rfs-prefix>export</rfs-prefix>
  <vfs-prefix></vfs-prefix>
  ...
</rendersettings>
...
```

SERVIDOR WEB / APACHE

En este punto es necesario configurar apache, de modo que sirva url en base a dominios.

APACHE

Para esta configuración necesitamos tener instalados los módulos **mod_jk**, **mod_proxy**, **mod_proxy_http**, **mod_rewrite** y **mod_deflate**(compresión gzip). Editamos el fichero **[Install-Path-Apache2]\conf\httpd.conf**:

1. Activamos los módulos que necesitamos:

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule rewrite_module modules/mod_rewrite.so
```

```
LoadModule deflate_module modules/mod_deflate.so
LoadModule jk_module modules/mod_jk.so
```

Nota: El módulo mod_jk no está preinstalado por defecto en algunas distribuciones de apache, por lo tanto es necesario descargarlo e instalarlo de forma manual siguiendo las instrucciones de instalación de la distribución correspondiente.

Para el caso de una instalación sobre Windows 32 bits de apache 2.2 se puede descargar de:

<http://ftp.cixug.es/apache/tomcat/tomcat-connectors/jk/binaries/windows/tomcat-connectors-1.2.40-windows-i386-httpd-2.2.x.zip>

Basta con descomprimir el archivo y mover el archivo mod_jk a la carpeta **[Install-Path-Apache2]\modules**

2. Configuramos el conector mod_jk:

```
<IfModule mod_jk.c>
  JkWorkerProperty worker.list=ocms
  JkWorkerProperty worker.ocms.type=ajp13
  JkWorkerProperty worker.ocms.host=192.168.100.5
  JkWorkerProperty worker.ocms.port=8009
  JkOptions +RejectUnsafeURI +ForwardKeySize +ForwardURISCompat
</IfModule>
```

3. Configuramos la compresión gzip:

```
<IfModule mod_deflate.c>
  SetOutputFilter DEFLATE
  SetEnvIfNoCase Request_URI \.(?:gif|jpe?g|png)$ no-gzip dont-vary
  SetEnvIfNoCase Request_URI \.(?:exe|t?gz|zip|bz2|sit|rar)$ no-gzip
  dont-vary
  <IfModule mod_headers.c>
    Header append Vary User-Agent
  </IfModule>
</IfModule>
```

4. Configuramos el dominio virtual:

OpenCms 9.5.x dispone de una utilidad que permite generar los VirtualHost en base a una plantilla, esta utilidad lo que hace es generar un fichero virtualhost por cada sitio.

Si configuramos apache para que incluya los virtualhost generados en ese directorio, podemos hacer que automáticamente se actualice la configuración y se reinicie apache.

En cualquier caso la configuración que se genera, para nuestro ejemplo es la siguiente:

```
<VirtualHost *:80 >
  ServerAdmin admin@example.com
  DocumentRoot "C:/tomcat/webapps/opencms"
  ServerName test.principal.com

  <Directory C C:/tomcat/webapps/opencms>
    Options FollowSymlinks
    AllowOverride All
    Order allow,deny
    Allow from all
  </Directory>

  SetEnvIfNoCase Request_URI ^/opencms/resources/.*$ no-jk
  SetEnvIfNoCase Request_URI ^/export/.*$ no-jk

  RewriteEngine On

  RewriteCond %{REQUEST_FILENAME} (.+)\.php(.*?)
  RewriteRule (.*) / [F]

  RewriteCond %{REQUEST_URI} !^/opencms/resources/.*$
  RewriteCond %{REQUEST_URI} !^/export/.*$
  RewriteRule !^/opencms/opencms/(.*)$
/opencms/opencms%{REQUEST_URI} [PT]

  RewriteCond %{REQUEST_URI} ^/export/.*$
  RewriteCond "%{DOCUMENT_ROOT}%{REQUEST_FILENAME}" !-f
  RewriteCond
"%{DOCUMENT_ROOT}%{REQUEST_FILENAME}/index_export.html" !-f
  RewriteRule .*
http://localhost:8080/opencms/handle404?exporturi=%{REQUEST_URI}&%
{QUERY_STRING} [P]

  #RewriteCond %{REQUEST_URI} ^/opencms/resources/.*$
  #RewriteRule ^/opencms/(.*)$ /$1

  JkMount /* ocms
</VirtualHost>
```

CONFIGURACION FRONT-APACHE

El objetivo es que todas las llamadas a OpenCms y los sitios que aloje pasen por un mismo punto que hará las veces de reverse-proxy y enrute a opencms.

Para ello es necesario tener Tomcat instalado como reverse-proxy y configurar correctamente un VirtualHost que enrute al servidor de OpenCms:

```
<VirtualHost *:80>
```

```
ErrorLog "[Install-Path-Apache2]/logs/test-error.log"

CustomLog "[Install-Path-Apache2]/logs/test-access.log" common

ServerName test.principal.com
#Otros Dominios que apunten al servidor OpenCms
ServerAlias resources.principal.com
ServerAlias test2.principal.com

ProxyRequests Off
ProxyPreserveHost On

# Denegamos el acceso a System/login*
#-----
<Location /system/login*>
    Order deny,allow
    Deny from all
</location>
<Location /system/login/*>
    Order deny,allow
    Deny from all
</location>
#-----

<location / >
    ProxyPass http://OpenCms-Server/
    ProxyPassReverse http://OpenCms-Server /
</location>

</VirtualHost>
```