

INSTALACIÓN Y CONFIGURACIÓN DE SERVIDOR DE CATÁLOGO GEONETWORK CON BASE DE DATOS POSTGRESQL EN SERVIDOR DEBIAN

El instalador de Geonetwork multiplataforma es un archivo JAR y para su ejecución se necesita visualizar una interfaz gráfica; de modo que si hacemos una instalación en nuestra máquina personal simulando un servidor web lo mas probable es que no tengamos problema en ejecutarlo, pero, al trabajar en un servidor es muy probable que tengamos inconvenientes de ejecución por falla en dependencias.

Es así, que sobre la base de nuestra experiencia consideramos hacer la instalación de Geonetwork en un servidor montando un directorio remoto del servidor en nuestra máquina “local” para poder ejecutar el instalador sin mayores inconvenientes. Para ello procedemos de la siguiente manera:

INSTALACION JAVA6 (JDK)

1. Instalaremos JAVA6 desde los repositorios haciendo:

```
MI_USUARIO@MI_EQUIPO:~$ SUDO APT-GET INSTALL SUN-JAVA6-JDK
```

2. Nos aseguramos que usemos por defecto Java 6 que hemos instalado

```
MI_USUARIO@MI_EQUIPO:~$ SUDO UPDATE-JAVA-ALTERNATIVES -S JAVA-6-SUN
```

INSTALACIÓN DE TOMCAT 6

3. Instalamos el servidor de aplicaciones Java “Tomcat”

```
MI_USUARIO@MI_EQUIPO:~$ SUDO APT-GET INSTALL TOMCAT6
```

4. Hacemos ajustes a la configuración de Tomcat para lo cual editamos lo siguiente:

```
MI_USUARIO@MI_EQUIPO:~$ SUDO VI /ETC/DEFAULT/TOMCAT6
```

Debemos asegurarnos que dentro del archivo /etc/default/tomcat6 las líneas siguientes estén presente y que a la vez NO estén “comentadas”

```
JAVA_HOME=/USR/LIB/JVM/JAVA-6-SUN
```

```
TOMCAT5_SECURITY=NO
```

5. Reiniciamos Tomcat.

```
MI_USUARIO@MI_EQUIPO:~$ SUDO /ETC/INIT.D/TOMCAT6 RESTART
```

INSTALACIÓN DE GEONETWORK

6. Montamos el directorio remoto del servidor /usr/local en un directorio de nuestra máquina “local” usando el comando SSHFS (sino tenemos instalado SSHF lo instalamos desde repositorio con apt-get install)

```
USUARIO_LOCAL@EQUIPO_LOCAL:~/MNT# MKDIR INSTAL
```

```
USUARIO_LOCAL@EQUIPO_LOCAL:# SSHFS -P 22 ROOT@LOCALHOST:/USR/LOCAL /MNT/INSTAL
```

7. Descargamos el instalador de Geonetwork en el directorio local que hemos direccionado con el servidor

```
USUARIO_LOCAL@EQUIPO_LOCAL:# CD /MNT
```

```
USUARIO_LOCAL@EQUIPO_LOCAL:~/MNT$ WGET
HTTP://SOURCEFORGE.NET/PROJECTS/GEONETWORK/FILES/GEONETWORK\_OPENSOURCE/V2.4.0/GEONETWORK-INSTALL-2.4.0-0.JAR/DOWNLOAD
```

8. Instalamos geonetwork en el directorio del servidor que tenemos redireccionado:

```
USUARIO_LOCAL@EQUIPO_LOCAL:~/MNT$ JAVA -JAR GEONETWORK-INSTALL-2.4.0-0.JAR
```

9. Durante la instalación seleccionaremos **únicamente** el componente “core “ y ejecutamos el instalador

1. click Siguiente
2. Lea y Acepte los términos de licencia
3. Use la ubicación por defecto (/usr/local/geonetwork), click Siguiente
4. Install *Core*
5. Espere que se complete la instalación
6. NO instale la data de ejemplo “sample metadata
7. Instale los lanzadores
8. Ignore la opción “primeros pasos” o “getting started”
9. Hecho!

10. Al terminar hemos instalado Geonetwork en el directorio /usr/local del servidor, es decir, que tendremos en el servidor el directorio: /usr/local/geonetwork

De esta manera hemos instalado Geonetwork en nuestro servidor; sin embargo aún no funcionará porque no hemos hecho la configuración necesaria.

CONFIGURAMOS TOMCAT6 PARA QUE “SIRVA” GEONETWORK

11. Configuramos TOMCAT SERVER en el servidor

```
MI_USUARIO@MI_EQUIPO:~$ SUDO VI /ETC/TOMCAT6/SERVER.XML
```

Estando en modo edición con la aplicación VI, buscamos dentro del archivo server.xml, la instrucción HOST ELEMENT y agregamos GEONETWORK e INTERMAP:

```
<!-- DEFINE THE DEFAULT VIRTUAL HOST
NOTE: XML SCHEMA VALIDATION WILL NOT WORK WITH XERCES 2.2.
-->
<HOST NAME="LOCALHOST" APPBASE="WEBAPPS"
UNPACKWARS="TRUE" AUTODEPLOY="TRUE"
XMLVALIDATION="FALSE" XMLNAMESPACEAWARE="FALSE">

<CONTEXT PATH="/GEONETWORK"
DOCBASE="/USR/LOCAL/GEONETWORK/WEB/GEONETWORK/"
CROSSCONTEXT="FALSE"
DEBUG="0"
RELOADABLE="FALSE"/>

<CONTEXT PATH="/INTERMAP"
DOCBASE="/USR/LOCAL/GEONETWORK/WEB/INTERMAP/"
CROSSCONTEXT="FALSE"
DEBUG="0"
RELOADABLE="FALSE"/>

<!-- SINGLESIGNON VALVE, SHARE AUTHENTICATION BETWEEN WEB
APPLICATIONS
DOCUMENTATION AT: /DOCS/CONFIG/VALVE.HTML -->
<!--
<VALVE
CLASSNAME="ORG.APACHE.CATALINA.AUTHENTICATOR.SINGLESIGNON" />
-->

<!-- ACCESS LOG PROCESSES ALL EXAMPLE.
DOCUMENTATION AT: /DOCS/CONFIG/VALVE.HTML -->
<!--
<VALVE CLASSNAME="ORG.APACHE.CATALINA.VALVES.ACCESSLOGVALVE"
DIRECTORY="LOGS"
PREFIX="LOCALHOST_ACCESS_LOG." SUFFIX=".TXT"
PATTERN="COMMON" RESOLVEHOSTS="FALSE"/>
-->

</HOST>
```

12. Cambiamos 'propietario' al directorio web de Geonetwork en el servidor

```
MI_USUARIO@MI_EQUIPO:~$ SUDO CHOWN -R TOMCAT6
/USR/LOCAL/GEONETWORK/WEB
```

13. Reiniciamos Tomcat en el servidor

```
MI_USUARIO@MI_EQUIPO:~$ SUDO /ETC/INIT.D/TOMCAT6 RESTART
```

14. Creamos en el servidor los enlaces simbólicos desde /var/lib/tomcat6/webapps

```
MI_USUARIO@MI_EQUIPO:~/VAR/LIB/TOMCAT6/WEBAPPS# LN -S
/USR/LOCAL/GEONETWORK/WEB/INTERMAP
```

```
MI_USUARIO@MI_EQUIPO:~/VAR/LIB/TOMCAT6/WEBAPPS# LN -S
/USR/LOCAL/GEONETWORK/WEB/GEONETWORK/
```

CONFIGURACIÓN DE BASE DE DATOS POSTGRESQL CON GEONETWORK

15. Nuevamente para trabajar desde nuestra máquina “local” haremos un redireccionamiento de puerto para usar interfaz gráfica con PgAdmin (sino tenemos instalado pgAdmin lo podemos instalar desde repositorio con 'apt-get install'. Vemos a redirigir el puerto 5432 del postgresQL remoto en local (previamente apagamos nuestra base de datos para tener disponible el puerto 5432 local)

```
USUARIO_LOCAL@EQUIPO_LOCAL:~$ SSH -P 22 -L 5432:LOCALHOST:5432
MI_USUARIO@HOST_SERVIDOR
```

16. Ahora tenemos como “local” el puerto 5432 del servidor; Con pgAdminIII y como DBA, creamos el rol de login “geonet/MI_CLAVE_geonet” sin privilegios y configuramos su variable search_path con los valores “geonet, public”.

17. Creamos el esquema “geonet” cuyo propietario es geonet.

18. Ejecutamos la herramienta “gast” desde nuestra máquina local.

```
USUARIO_LOCAL@EQUIPO_LOCAL:~/MNT/INSTAL/GEONETWORK$ JAVA -JAR GAST/
GAST.JAR
```

19. Configuramos la conexión a la base de datos
 1. Seleccionamos **Configuration->DBMS** del menú a la izquierda
 2. Seleccionamos PostgreSQL de la lista de *DBMS*
 3. Ingresamos los siguientes parámetros:

```
SERVER LOCALHOST
PORT 5432
DATABASE 'NOMBRE_BASE_DATOS' (NODO_IDE_HF)
USERNAME GEONET
PASSWORD MI_CLAVE_GEONET
```

4. Seleccionamos Save

20. Instalamos la Base de Datos

1. Seleccionamos **Database->Setup** del menú de la izquierda

2. Click *Setup*
3. Click *Yes* para borrar base de datos actual (asegurese que está limpia la base de datos en el esquema 'geonet')
4. Ignore la advertencia de referencia cíclica

21. Cerramos GAST

22. Verificamos nuestro site GeoNetwork por el puerto de Tomcat6

```
HTTP://SERVIDOR_HOST:8180/GEONETWORK
```

23. Desmontamos el directorio compartido

```
USUARIO_LOCAL@EQUIPO_LOCAL:~$ FUSERMOUNT -U /MNT/INSTAL
```

24. Desmontamos el puerto redireccionado

Ahora solo nos queda por hacer que Geonetwork salga por el puerto 80 de nuestro servidor web Apache.

INTEGRAR TOMCAT6 CON APACHE2 USANDO mod_jk

25. Instalamos en el servidor mod_jk

```
MI_USUARIO@MI_EQUIPO:~$ SUDO APT-GET INSTALL LIBAPACHE2-MOD-JK
```

26. Configuramos el modulo jk para lo cual debemos crear un archivo de nombre jk.conf. Este archivo lo crearemos desde la consola usando la aplicación VI

```
MI_USUARIO@MI_EQUIPO:~$ SUDO VI /ETC/APACHE2/MODS-AVAILABLE/JK.CONF
```

Agregamos las siguientes líneas:

```
<IFMODULE MOD_JK.C>
JKWORKERSFILE /ETC/LIBAPACHE2-MOD-JK/WORKERS.PROPERTIES
JKLOGFILE /VAR/LOG/APACHE2/MOD_JK.LOG
JKSHMFILE /VAR/LOG/APACHE2/MOD_JK.SHM
JKLOGLEVEL INFO
</IFMODULE>
```

Guardamos y salimos

27. Configuramos ahora el 'workers' en Apache

```
MI_USUARIO@MI_EQUIPO:~$ SUDO VI /ETC/APACHE2/WORKERS.PROPERTIES
```

Agregamos las siguientes líneas:

```

# WORKERS.PROPERTIES -
#
# THIS FILE IS A SIMPLIFIED VERSION OF THE WORKERS.PROPERTIES SUPPLIED
# WITH THE UPSTREAM SOURCES. THE JNI INPROCESS WORKER (NOT BUILD IN
# THE
# DEBIAN PACKAGE) SECTION AND THE AJP12 (DEPRECATED) SECTION ARE
# REMOVED.
#
# AS A GENERAL NOTE, THE CHARACTERS $( AND ) ARE USED INTERNALLY TO
# DEFINE
# MACROS. DO NOT USE THEM IN YOUR OWN CONFIGURATION!!!
#
# WHENEVER YOU SEE A SET OF LINES SUCH AS:
# X=VALUE
# Y=$(X)\SOMETHING
#
# THE FINAL VALUE FOR Y WILL BE VALUE\SOMETHING
#
# NORMALLY ALL YOU WILL NEED TO DO IS UN-COMMENT AND MODIFY THE FIRST
# THREE
# PROPERTIES, I.E. WORKERS.TOMCAT_HOME, WORKERS.JAVA_HOME AND PS.
# MOST OF THE CONFIGURATION IS DERIVED FROM THESE.
#
# WHEN YOU ARE DONE UPDATING WORKERS.TOMCAT_HOME,
# WORKERS.JAVA_HOME AND PS
# YOU SHOULD HAVE 3 WORKERS CONFIGURED:
#
# - AN AJP13 WORKER THAT CONNECTS TO LOCALHOST:8009
# - A LOAD BALANCER WORKER
#
#
# OPTIONS ( VERY IMPORTANT FOR JNI MODE )
#
# WORKERS.TOMCAT_HOME SHOULD POINT TO THE LOCATION WHERE YOU
# INSTALLED TOMCAT. THIS IS WHERE YOU HAVE YOUR CONF, WEBAPPS AND LIB
# DIRECTORIES.
#
W
#
# WORKERS.JAVA_HOME SHOULD POINT TO YOUR JAVA INSTALLATION.
# NORMALLY
# YOU SHOULD HAVE A BIN AND LIB DIRECTORIES BENEATH IT.
#
W           E           6 S   O
#
# YOU SHOULD CONFIGURE YOUR ENVIRONMENT SLASH... PS=\ ON NT AND / ON
# UNIX
# AND MAYBE SOMETHING DIFFERENT ELSEWHERE.
#
P
#
# ----- ADVANCED MODE -----
# -----
#

```

```

#
#----- WORKER LIST -----
#-----
#
#
# THE WORKERS THAT YOUR PLUGINS SHOULD CREATE AND WORK WITH
#
W

#
#----- AJP13_WORKER WORKER DEFINITION -----
#-----
#

#
# DEFINING A WORKER NAMED AJP13_WORKER AND OF TYPE AJP13
# NOTE THAT THE NAME AND THE TYPE DO NOT HAVE TO MATCH.
#
W
W
W
#
# SPECIFIES THE LOAD BALANCE FACTOR WHEN USED WITH
# A LOAD BALANCING WORKER.
# NOTE:
# ----> LBFACTOR MUST BE > 0
# ----> LOW LBFACTOR MEANS LESS WORK DONE BY THE WORKER.
W

#
# SPECIFY THE SIZE OF THE OPEN CONNECTION CACHE.
#WORKER.AJP13_WORKER.CACHESIZE

#
#----- DEFAULT LOAD BALANCER WORKER DEFINITION -----
#-----
#

#
# THE LOADBALANCER (TYPE LB) WORKERS PERFORM WIGHTED ROUND-ROBIN
# LOAD BALANCING WITH STICKY SESSIONS.
# NOTE:
# ----> IF A WORKER DIES, THE LOAD BALANCER WILL CHECK ITS STATE
#       ONCE IN A WHILE. UNTIL THEN ALL WORK IS REDIRECTED TO PEER
#       WORKERS.
W
W

```

Guardamos y salimos

28. Activamos mod_jk

```
MI_USUARIO@MI_EQUIPO:~$ SUDO A2DISMOD JK
```

```
MI_USUARIO@MI_EQUIPO:~$ SUDO A2ENMOD JK
```

29. Creamos la configuración en Apache de acceso a Geonetwork

```
MI_USUARIO@MI_EQUIPO:~$ SUDO VI /ETC/APACHE2/SITES-  
AVAILABLE/DEFAULT
```

Agregamos las siguientes líneas:

```
#PUNTO MONTAJE APACHE-TOMCAT  
<IFMODULE MOD_JK.C>  
  
#GEONETWORK  
ALIAS /GEONETWORK /VAR/LIB/TOMCAT6/WEBAPPS/GEONETWORK  
JKMOUNT /GEONETWORK/* AJP13_WORKER  
  
#INTERMAP  
ALIAS /INTERMAP /VAR/LIB/TOMCAT6/WEBAPPS/INTERMAP  
JKMOUNT /INTERMAP/* AJP13_WORKER  
  
</IFMODULE>
```

Guardamos y salimos

30. Nos aseguramos de activar la configuración de sitios correcta

```
MI_USUARIO@MI_EQUIPO:~$ SUDO A2ENSITE DEFAULT
```

31. Reiniciamos Apache

```
MI_USUARIO@MI_EQUIPO:~$ SUDO /ETC/INIT.D/APACHE2 FORCE-RELOAD
```

32. Verificamos nuestro site GeoNetwork por el puerto 80 de Apache

```
HTTP://SERVIDOR\_HOST/GEONETWORK
```