

Guia Ràpida d'Instal·lació de HELIUM

Aquest document explica com a partir d'un binari Helium o una compilació del codi de HELIUM, es pot instal·lar HELIUM sobre un servidor JBoss i el sistema gestor de bbdd PostgreSQL.

(A) COMPILAT: S'han de seguir les instruccions de l'Annex I d'aquest document.

(B) DESCARREGAR BINARI: Accedint al projecte helium de sourceforge ens podem descarregar el binari corresponent. Per exemple accedir via web a la següent adreça i descarregar el fitxer http://sourceforge.net/projects/gehelium/files/helium-2.x/helium-2.1.0/bin/jboss_postgresql_helium-2.1.0.zip/download al nostre home. Ho descomprimir-ho i obtindrem un directori ~/helium.

Aquí s'enumeren la llista d'accions per posar en marxa un sistema HELIUM:

1.- Crear usuari de BBDD:

1.1.- Connectar-se a la BBDD:

```
$ sudo bash
$ su postgres
$ psql -U postgres
```

1.2.- Nou usuari:

```
CREATE USER "helium" WITH ENCRYPTED PASSWORD 'helium' NOCREATEDB
NOCREATEUSER;
CREATE USER "WWW_HELIUM" WITH ENCRYPTED PASSWORD 'helium' NOCREATEDB
NOCREATEUSER;
CREATE USER "www_helium" WITH ENCRYPTED PASSWORD 'helium' NOCREATEDB
NOCREATEUSER;
```

1.3.- Crear la BBDD

```
CREATE DATABASE "helium" WITH OWNER=helium;
```

1.4.- Sortir

```
\q
```

2.- Permisos d'accés a la BBDD per part de l'usuari i d'altres hosts.

2.1.- S'ha d'editar el fitxer /etc/postgresql/8.4/main/pg_hba.conf (o /usr/local/pgsql/data/pg_hba.conf en slax) i afegir les següents línies:

```
## ===== HELIUM =====
# Dona permisos per a que l'usuari helium (BBDD helium) es pugui connectar
# amb la seva contrasenya
local    helium          helium          password

# Dona permisos a l'usuari helium (BBDD helium) per a que es pugui connectar
# a la BBDD des de qualsevol host.
host     helium          helium          www.xxx.yyy.zzz/24  trust
```

On posa `www.xxx.yyy.zzz` l'hem de substituir per la nostra IP. Per exemple si la nostra IP es 192.168.121.132 llavors on posa `www.xxx.yyy.zzz` hem d'escriure 192.168.121.132. Podem obtenir la nostra IP escrivint `ipconfig` en sistemes Windows o `ifconfig` en sistemes linux.

2.2.- Reiniciem el postgres per a que els canvis s'apliquin:

```
$ /etc/init.d/postgresql restart
```

Nota: Si no teniu accés des d'ordinadors diferents a localhost, segui el que es diu en la següent pàgina <http://www.cyberciti.biz/tips/postgres-allow-remote-access-tcp-connection.html>

3 .- Crear BBDD i configurar-la:

3.1.- Conectar-se al servidor de BBDD

```
$ psql -h localhost -p 5432 -U helium -W -d helium
```

o si ens connectem des d'un altra ordinador

```
$ psql -h www.xxx.yyy.zzz -p 5432 -U helium -W -d helium
```

on www.xxx.yyy.zzz és la IP on esta instal.lada la BBDD postgresql.

3.2.- Donam permisos al usuari "helium":

```
GRANT ALL PRIVILEGES ON DATABASE "helium" TO helium;  
GRANT ALL PRIVILEGES ON SCHEMA PUBLIC TO helium;
```

3.3.- Importam l'estructura de taules a la BBDD

```
\i ~/helium/scripts/bbdd/postgresql/helium_create_schema.sql
```

3.4.- Importam les dades

```
\i ~/helium/scripts/bbdd/postgresql/helium_create_data.sql
```

3.5.- Actualitzem l'estructura de la BBDD:

```
\i ~/helium/doc/resources/db/2.1/01_postgres_update.sql
```

3.6.- Modificar la contrasenya d'admin per admin ja que els scripts de creació de dades inicials (`helium_create_data.sql`) donen d'alta l'usuari admin amb una contrasenya encriptada en MD5, format incompatible amb el format MD5 de helium:

```
update hel_usuari set contrasenya='admin' where codi='admin';
```

3.7.- Sortir

```
\q
```

4 .- Definir variable d'entorn de la ubicació del JBOSS:

```
$ set JBOSS=/usr/local/jboss-3.2.8.SP1
$ export JBOSS
```

5.- Eliminar llibreries del JBOSS:

5.1.- Eliminar l'arxiu xml-apis.jar del directori \$JBOSS/lib/endorsed:

```
# rm $JBOSS/lib/endorsed/xml-apis.jar
```

5.2.- Eliminar l'arxiu cglib-full-2.0.1.jar del directori \$JBOSS/server/default/deploy/jboss-hibernate.sar:

```
# rm $JBOSS/server/default/deploy/jboss-hibernate.sar/cglib-full-2.0.1.jar
```

6.- Copiar EAR. Dins ~/helium/helium-ear/target s'haurà creat un fitxer "ear" anomenat "helium.ear" que em de copiar al jboss:

```
$ cp ~/helium/helium-ear/target/helium.ear $JBOSS/server/default/deploy/
```

7.- Copiar fitxers de configuració:

```
# cp ~/helium/doc/resources/appserver/jboss/login-config.xml
$JBOSS/server/default/conf/login-config-helium.xml

# cp ~/helium/scripts/datasources/helium-postgresql-ds.xml
$JBOSS/server/default/deploy/

# cp ~/helium/doc/resources/appserver/jboss/helium-mail-service.xml
$JBOSS/server/default/deploy/
```

Afegir configuració de seguretat per helium. Per fer això hem d'afegir al final del document \$JBOSS_HOME/server/default/conf/login-config.xml el següent codi xml abans de </policy>:

```
<application-policy name="seycon">
  <authentication>
    <login-module code="org.jboss.security.auth.spi.DatabaseServerLoginModule"
flag="required">
      <module-option name="dsJndiName">
        java:/es.caib.helium.db</module-option>
      <module-option name="principalsQuery">
        select hel_usuario.contrasenya from hel_usuario where hel_usuario.codi=?
      </module-option>
      <module-option name="rolesQuery">
        select hel_usuario_permiss.permis,'Roles' from hel_usuario_permiss where
        hel_usuario_permiss.codi=?
      </module-option>
      <module-option name="unauthenticatedIdentity">guest</module-
option>
    </login-module>
  </authentication>
```

```
</application-policy>
```

8.- Afegir propietat que apunta al fitxers de propietats de helium

8.1.- Crear un directori anomenat helium_config en /:

```
# mkdir /helium_config
```

8.2.- Crear dos directoris per emmagatzemar els fitxers de lucene i custodia:

```
# mkdir /helium_config/lucene  
# mkdir /helium_config/custodia
```

8.3.- Copiar el fitxer de propietats ~/helium/doc/resources/config/global.properties al directori creat en el punt anterior:

```
# cp ~/helium/doc/resources/config/global.properties /helium_config
```

8.4.- Editar algunes propietats. Obrir l'arxiu /helium_config/global.properties i modificar les següents entrades:

- app.hibernate.dialect=**org.hibernate.dialect.PostgreSQLDialect**
- app.lucene.directory=**fsDirectory**
- app.lucene.fs.basedir=**file:///helium_config/lucene**
- app.correu.remitent=**helium@ibit.org**
- app.custodia.plugin.filesystem.basedir=**/helium_config/custodia**

Per a la resta de propietats es pot llegir el manual que podeu trobar en
~/helium/doc/manuals/manual_instalacio.pdf.

8.5.- Afegir referencia al fitxer de propietats. Per això editarem el fitxer

\$JBOSS/server/default/conf/jboss-service.xml i al final (abans de </server>) afegiren el següent text:

```
<mbean code="org.jboss.varia.property.SystemPropertiesService"  
name="jboss:type=Service,name=BootProperties">  
<attribute name="Properties">  
<!-- En /app/caib/helium/helium.properties -->  
es.caib.helium.properties.path=file:///helium_config/global.properties  
</attribute>  
</mbean>
```

8.6.- Controladors Java d'accés a la base de dades. S'ha de copiar l'arxiu de connector de base de dades JDBC apropiat dins la carpeta \$JBOSS/server/default/lib. Segons el SGBD s'ha de fer:

- PostgreSQL: es pot descarregar la darrera versió de
<http://jdbc.postgresql.org/download.html> triant la versió JDBC3:

```
$ wget -P $JBOSS/server/default/lib  
http://jdbc.postgresql.org/download/postgresql-9.0-801.jdbc3.jar
```

- Oracle: Triar l'arxiu ojdbc14.jar que el poden descarregar de http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/htdocs/jdbc_10201.html i l'hem de copiar a \$JBOSS/server/default/lib.

9.- Instal·lar OpenOffice en mode servei

Crearem el fitxer openoffice.sh en el directori /etc/init.d amb el contingut que es descriu en l'ANNEX II. Per fer que el sistema executi cada vegada que arranca aquest script s'ha d'executar la següent comanda:

```
$ chkconfig -add /etc/init.d/openoffice.sh
```

10.- Configurar Administrador

Per començar a fer feina hem de definir un Entorn per l'administrador:

10.1.- Crear un Entorn

Anar al menú “Configuració” i triar l'opció titulada “Entorns”. Pitjar el botó “Nou Entorn”. A la pàgina que ens aparegui omplir les dades que ens realitza i pitjar “Crear”.

10.2.- Donar permisos per admin a l'Entorn

Al finalitzar el pas anterior Helium anirà a la llista d'Entorns. Hem de pitjar el botó “Permisos” de l'Entorn que acabem de crear. En la pantalla de permisos hem d'Afegir un Permis. Per això només omplirem allà on diu “Usuari/Rol” amb la cadena “admin” i marcarem “ADMINISTRATION”. Finalment pitjarem el botó “Afegir”.

10.3.- Automàticament el sistema ens assignarà l'entorn a l'usuari “admin”. Podem assignar els entorns que vulguem a cada usuari i podem triar el que vulguem des del menú principal de “Selecció Entorn”

ANNEX I: Compilar HELIUM des de subversion de sourceforge

Aquest manual explica com compilar l'aplicació HELIUM a partir del repositori de subversion de sourceforge. Es requereix java i maven. La compilació es realitzarà en un directori helium del home de l'usuari (~/.helium/).

- (1) Descarregar codi des de subversion de sourceforge (En aquest cas ens descarregam el principal). Estant en el *home* de l'usuari fer check out executant la següent comanda:

```
$ svn co
https://gehelium.svn.sourceforge.net/svnroot/gehelium/helium/branches/branch-2.1/ helium
```

- (2) Editar el fitxer helium-core/src/main/resources/net/conselldemallorca/helium/core/context/application-context-common.xml i modificar el tipus de codificació de MD5 a plain:

Codi Actual:

```
<bean id="passwordDigester"
class="net.conselldemallorca.helium.core.util.PasswordDigester">
    <constructor-arg value="MD5"/>
</bean>
```

Codi Correcte

```
<bean id="passwordDigester"
class="net.conselldemallorca.helium.core.util.PasswordDigester">
    <constructor-arg value="plain"/>
</bean>
```

- (3) Descarregar JTA ja que no pot estar dins dels repositoris Maven a causa de la seva llicència (Sun's Binary License). Per això hem d'anar a la pàgina de JTA accedint a <http://java.sun.com/products/jta/> i d'aquí baixar-nos el jar referenciat amb el següent text "Class Files 1.0.1B". Aquestes passes en descarregaran un fitxer jta-1_0_1B-classes.zip que guardarem en el home de l'usuari (~/.). L'instal·larem en el nostre repositori local emprant la següent comanda:

```
$ mvn install:install-file -Dfile=./jta-1_0_1B-classes.zip
-DgroupId=javax.transaction -DartifactId=jta -Dversion=1.0.1B
-Dpackaging=jar
```

- (4) Editar el fitxer build.properties (~/.helium/build.properties) i modificar la propietat appserv.jboss a true:

```
web.realm.name=Limit Technologies
appserv.jboss=true
```

- (5) Compilar-ho tot. Ens situem en el directori ~/.helium i per compilar executam:

```
$ mvn install
```

(La primera vegada que s'executi tardarà un poc ja que s'ha de descarregar el repositori de llibreries jar descrites en l'arxiu pom.xml de maven)

ANNEX II: Script de Servei de OPENOFFICE

```
#!/bin/bash
# openoffice.org headless server script
#
# chkconfig: 2345 80 30
# description: headless openoffice server script
# processname: openoffice
#
# Author: Vic Vijayakumar
# Modified by Federico Ch. Tomaszczik
#
OOo_HOME=/usr/bin
SOFFICE_PATH=$OOo_HOME/soffice
PIDFILE=/var/run/openoffice-server.pid

set -e

case "$1" in
start)
if [ -f $PIDFILE ]; then
echo "OpenOffice headless server has already started."
sleep 5
exit
fi
echo "Starting OpenOffice headless server"
$SOFFICE_PATH -headless -nologo -nofirststartwizard
-accept="socket,host=127.0.0.1,port=8100;urp" & > /dev/null 2>&1
touch $PIDFILE
;;
stop)
if [ -f $PIDFILE ]; then
echo "Stopping OpenOffice headless server."
killall -9 soffice && killall -9 soffice.bin
rm -f $PIDFILE
exit
fi
echo "Openoffice headless server is not running."
exit
;;
*)
echo "Usage: $0 {start|stop}"
exit 1
esac
exit 0
```

(Font <http://www.openvpms.org/documentation/install-openoffice-headless-service-ubuntu>)