



Manual de Instalación en Fedora

Manual de Instalación en Fedora



Versión 2 – Enero 2015



Índice de contenido

1. Paquetes de Software y archivos necesarios.....	3
1.1 Obtener permisos de root.....	3
2. Instalación de Oracle Java JDK 6.....	3
3. Instalación de PostgreSQL.....	4
3.1 Descarga de los paquetes .RPM necesarios.....	5
3.2 Instalación y Configuración.....	5
3.3 Instalación de PostgreSQL como un servicio al arranque.....	6
3.4 Otros ajustes y tareas relacionadas con PostgreSQL.....	7
4. Instalación de Libertya.....	7
4.1 Configuración e instalación de binarios.....	7
4.2 Creación de la base de datos.....	7
4.3 Configuración de Libertya.....	8
4.4 Instalación del Servidor de Libertya como un servicio al arranque.....	9



1. Paquetes de Software y archivos necesarios

Software de base

- Las instrucciones del presente manual fueron probadas en una instalación por defecto de una distribución Fedora 20 con actualización de los paquetes de software al día 11/12/2014.

Libertya

- ServidorOXP_VAA.MM.zip
- dump_libertya_AAMMar.zip
- libertyad_fedora.sh (script para autoarranque del servidor)

Java

- Descargar desde la [web de Oracle](#) el SDK de Java según corresponda al sistema operativo utilizado:
 - jdk-6u45-linux-x64-rpm.bin (64 bits)
 - jdk-6u45-linux-i586-rpm.bin (32 bits)

1.1 Obtener permisos de root

Para todas las secciones donde se solicite obtener permisos de root, existen dos alternativas:

Para usuarios con privilegios de administración:

```
sudo -s
```

Para usuarios estándar, sin permisos de administración, es necesario loguearse en el sistema como root, para lo cual es necesario contar con el password del usuario root.

```
su -
```

2. Instalación de Oracle Java JDK 6

Obtener permisos de root. (Ver sección 1.1)

Copiar en un directorio, por ejemplo /root, el archivo binario de instalación descargado (jdk-6u45-linux-<Bits>-rpm.bin).

Importante: Los comandos de ejemplo que se muestran a continuación suponen la

instalación en un sistema operativo de 64 bits. Si su sistema es de 32 bits reemplazar los nombres de archivo por el que corresponde.

Dar permisos de ejecución al archivo binario y extraer los paquetes de instalación RPM.

```
chmod +x jdk-6u45-linux-x64-rpm.bin
./jdk-6u45-linux-x64-rpm.bin -x
```

Instalar el paquete RPM

```
rpm -ivh jdk-6u45-linux-amd64.rpm
```

Ver las versiones de java instaladas

```
update-alternatives --config java
```

Chequear cuantos programas proveen java, ver el número asociado a la izquierda. Si solo hay uno, debemos instalar el java ORACLE en el lugar 2, si hay 2 en el 3, etc.

Comando para instalar java ORACLE en este listado. (ejemplo, número 2).

```
update-alternatives --install /usr/bin/java java /usr/java/jdk1.6.0_45/bin/java 2
```

Luego marcar por defecto la opción 2

```
update-alternatives --config java
```

Comprobar si la versión por defecto es la de ORACLE

```
java -version
```

Debería devolver algo similar a lo siguiente

```
java version "1.6.0_45"
Java(TM) SE Runtime Environment (build 1.6.0_45-b06)
Java HotSpot(TM) 64-Bit Server VM (build 20.45-b01, mixed mode)
```

3. Instalación de PostgreSQL

Obtener permisos de root. (Ver sección 1.1)

3.1 Descarga de los paquetes .RPM necesarios

Descargar los paquetes

```
wget http://yum.postgresql.org/8.4/redhat/rhel-6-x86_64/postgresql84-libs-8.4.22-1PGDG.rhel6.x86_64.rpm
```

```
wget http://yum.postgresql.org/8.4/redhat/rhel-6-x86_64/postgresql84-8.4.22-1PGDG.rhel6.x86_64.rpm
```

```
wget http://yum.postgresql.org/8.4/redhat/rhel-6-x86_64/postgresql84-server-8.4.22-1PGDG.rhel6.x86_64.rpm
```

3.2 Instalación y Configuración

Instalar los paquetes utilizando el comando rpm

```
rpm -ivh postgresql84-libs-8.4.22-1PGDG.rhel6.x86_64.rpm
rpm -ivh postgresql84-8.4.22-1PGDG.rhel6.x86_64.rpm
rpm -ivh postgresql84-server-8.4.22-1PGDG.rhel6.x86_64.rpm
```

Iniciar el cluster de base de datos. Esto inicializa el directorio donde estarán las bases de datos copiando los archivos y directorios requeridos por el servidor (archivos de configuración, etc).

```
su - postgres
/usr/pgsql-8.4/bin/initdb /var/lib/pgsql/8.4/data
exit
```

Editar el archivo *pg_hba.conf* (ubicado en el directorio */var/lib/pgsql/8.4/data*) para permitir acceso desde cualquier lugar

```
# "local" is for Unix domain socket connections only
local all all trust
# IPv4 local connections:
host all all 127.0.0.1/32 trust
host all all 0/0 trust
# IPv6 local connections:
host all all ::1/128 trust
```

Editar el archivo *postgresql.conf* (ubicado en el directorio PGDATA) habilitando la siguiente línea para escuchar conexiones en todas las interfaces de red



```
listen_addresses = '*'
```

El archivo para inicio del servicio se encuentra ubicado en `/etc/init.d/postgresql-8.4`. El mismo contiene una línea para generación del archivo `.pid` del servicio que genera un error al iniciar. Para corregir dicho error, ejecutar el siguiente comando.

```
sed -i 's/{PGMAJORVERSION}.${PGPORT}.pid/{PGMAJORVERSION}.pid/g'  
/etc/init.d/postgresql-8.4
```

Para abrir el puerto 5432 donde escucha el servicio de base de datos en el firewall

```
firewall-cmd --zone=public --add-port=5432/tcp --permanent  
firewall-cmd --reload
```

Probar si el servidor inicia correctamente.

Iniciar el servidor

```
service postgresql-8.4 start
```

Probar conectarse con PgAdmin desde el mismo servidor o desde otra PC de la LAN.

Luego detener el servidor para continuar

```
service postgresql-8.4 stop
```

3.3 Instalación de PostgreSQL como un servicio al arranque

Ejecutar el siguiente comando

```
systemctl enable postgresql-8.4.service
```

3.4 Otros ajustes y tareas relacionadas con PostgreSQL

- [Ajustes básicos a la instalación de postgresql](#)
- [Como crear y recuperar un backup de la base de datos](#)

4. Instalación de Libertya

Obtener permisos de root. (Ver sección 1.1)

4.1 Configuración e instalación de binarios

Crear y editar el archivo `/etc/profile.d/libertya.sh` para agregar las variables `JAVA_HOME` y



OXP_HOME necesarias

```
export JAVA_HOME=/usr/java/jdk1.6.0_45
export OXP_HOME=/ServidorOXp
```

Descomprimir ServidorOXp_VXX.YY.zip en el raíz /

```
unzip ServidorOXp_V14.02.zip
mv ServidorOXp /ServidorOXp
cd /ServidorOXp
chmod +x *.sh
cd utils
chmod +x *.sh
```

4.2 Creación de la base de datos

Cambiar el login al usuario postgres

```
su postgres
```

Crear el rol *libertya*

```
psql -U postgres -c "CREATE ROLE libertya LOGIN ENCRYPTED PASSWORD
'md52b2861c4d594f29f9ce9107c4560f3ae' SUPERUSER CREATEDB CREATEROLE VALID
UNTIL 'infinity' IN ROLE postgres; UPDATE pg_authid SET rolcatupdate=true WHERE
rolname='libertya';"
```

Crear el lenguaje *plpgsql* en la base de datos *template1* para que cada vez que se cree una nueva base de datos la misma ya contenga el lenguaje requerido por las funciones que contiene la base de datos de *libertya*.

```
createlang -U postgres -d template1 plpgsql
```

Crear base de datos *libertya*

```
psql -U postgres -c "CREATE DATABASE libertya WITH ENCODING='UTF8'
OWNER=libertya;"
```

Volver a usuario root

```
exit
```

Levantar el dump del esquema *libertya*



```
unzip dump_libertya_1402ar.sql.zip
psql -U libertya -d libertya -f dump_libertya_1402ar.sql
```

4.3 Configuración de Libertya

Ejecutar la interfaz de configuración de Libertya y setear los parámetros

```
/ServidorOXP/Configurar.sh
```

Si no se dispone de interfaz gráfica entonces utilizar el configurador por línea de comando. Primero copiar el archivo *LibertyaEnvTemplate.properties* a *LibertyaEnv.properties*.

```
cd /ServidorOXP
cp LibertyaEnvTemplate.properties LibertyaEnv.properties
```

Editar *LibertyaEnv.properties* seteando todos los parámetros manualmente (para que resulte mas sencillo se puede llevar un *LibertyaEnv.properties* ya configurado mediante la interfaz gráfica en otra implementación, y editar directamente ese en donde la mayoría de los parámetros no van a cambiar).

Ejecutar el configurador automático que toma los parámetros desde el archivo *LibertyaEnv.properties* (no requiere interfaz gráfica)

```
/ServidorOXP/ConfigurarAuto.sh
```

Para abrir los puertos configurados para el servidor Jboss de Libertya (por ejemplo 8080 y 1099) en el firewall y el puerto

```
firewall-cmd --zone=public --add-port=8080/tcp --permanent
firewall-cmd --zone=public --add-port=1099/tcp --permanent
firewall-cmd --reload
```

4.4 Instalación del Servidor de Libertya como un servicio al arranque

IMPORTANTE: A fin de mejorar la seguridad del sistema, se debe iniciar el servicio mediante un usuario regular y NO mediante el usuario **root**. Si bien la invocación al servicio se realiza mediante **root**, el script de inicio se encarga de delegar el owner del proceso a un usuario regular. A continuación se incluyen los pasos para iniciar el servicio mediante el usuario **libertya**.

Crear el usuario **libertya**



```
[root@/]# adduser libertya
```

Crear el directorio **/var/run/libertya**, donde se alojará el PID del servicio y asignar como owner de dicho directorio al usuario **libertya**

```
[root@/]# mkdir /var/run/libertya  
[root@/]# chown libertya /var/run/libertya
```

Cambiar el owner de los archivos en **/ServidorOXP** al usuario **libertya**. Toda operación sobre el directorio **ServidorOXP** debe ser realizada mediante el usuario **libertya** dado que de utilizar **root**, los archivos generados no tendran los permisos correctos.

```
[root@/ServidorOXP/]# chown libertya . -R
```

Configurar las variables de entorno **OXP_HOME** y **JAVA_HOME** para el nuevo usuario. Para ello editar el archivo **/home/libertya/.bashrc** y agregar las lineas:

```
export JAVA_HOME=/usr/java/jdk1.6.0_45  
export OXP_HOME=/ServidorOXP
```

Para aplicar los cambios en la sesión actual utilizar el siguiente comando:

```
[libertya@/]# source /home/libertya/.bashrc
```

En caso de ser necesario, reconfigurar los puertos 80 y 443 hacia el 8080 y 8443 (por defecto solo **root** puede gestionar puertos inferiores al 1024). Esta operación ya debe realizarse con el usuario **libertya**.

```
[libertya@/ServidorOXP/]# ./Configurar.sh
```

Iniciar el servidor manualmente con el usuario **libertya** para verificar que el mismo inicie correctamente con el nuevo usuario.

```
[libertya@/ServidorOXP/utis/]# ./IniciarServidor.sh  
[libertya@/ServidorOXP/utis/]# ./DetenerServidor.sh
```

Como **root**, copiar el script provisto **libertyad_fedora.sh** a **/etc/init.d/libertyad**

```
[root@/]# cp libertyad_fedora.sh /etc/init.d/libertyad
```

Asignar permisos de ejecución al script

```
[root@/]# chmod +x /etc/init.d/libertyad
```

Registrar el script como servicio del sistema



```
[root@/]# chkconfig --level 2345 libertyad on
```

Como root, iniciar libertya como servicio

```
[root@/]# service libertyad start
```

Verificar si se puede acceder al portal

```
[root@/]# wget localhost:8080
```

Para iniciar o detener el servidor se debe invocar al servicio con el usuario **root**

```
[root@/]# service libertyad start  
[root@/]# service libertyad stop
```