

2. Instalación de Odoo en Debian¹

Llevaremos a cabo la instalación de Odoo (*OpenErp*) en un **clon enlazado** MV de Debian. Posteriormente intentaremos acceder a su entorno gráfico desde una máquina cliente Windows y luego Ubuntu.

Para ello realizamos **clones enlazados de las plantillas de MV de Debian, Ubuntu-Escenario1 y Windows 10-Escenario1**.

A. Empezamos actualizando el sistema:

```
#apt-get update  
#apt-get dist-upgrade
```

Si no estaba actualizado es conveniente reiniciar el servidor.

Necesitamos **un usuario de sistema (-system)**, que será el que ejecute como **propietario todo el servicio de Odoo**. También nos interesa que su **carpeta personal sea donde vamos a alojar todo el sistema Odoo (-home=/opt/odoo)** y agregamos el grupo Odoo (*-group odoo*).

Nota.- podemos elegir otro usuario: maria, juanito, admin, miempresa.....

```
#adduser --system --home=/opt/odoo --group odoo
```

B. Vamos a instalar y configurar el servidor de bases de datos en PostgreSQL que es un sistema de bases de datos en software libre. Para ello:

```
#apt-get install postgresql
```

Para configurarlo lo primero es loguearnos con el usuario especial "postgres".

```
#su - postgres
```

Y ahora ya desde el usuario especial creamos la base de datos a utilizar dándole los permisos y configuraciones requeridas por Odoo.

```
$createuser --createdb --username postgres --no-createrole --no-superuser --pwprompt  
odoo
```

Nos pedirá el password y será el mismo que utilizemos como administrador. Y ya podemos salir del usuario especial "postgres".

```
$exit
```

El comando es bastante intuitivo, no obstante haremos un resumen: (*createuser*) creamos usuario (*-createdb*) y creamos base de datos, (*-username*) que pertenecerá al usuario "postgres" y sin

¹<http://colibris.es/tutorial/como-instalar-odoo-8-para-debian/>

permisos de superusuario ni roles (*-no-createrole -no-superuser*) para utilizar con la ruta de sistema Odoo (*-pwprompt odoo*).

C. A continuación (paciencia) instalamos Python y algunas librerías necesarias para Odoo.

```
#apt-get install python-dateutil python-decorator python-docutils python-feedparser  
python-gdata python-gevent python-imaging python-jinja2 python-ldap python-libxslt1  
python-lxml python-mako python-mock python-openid python-passlib python-psutil  
python-psycopg2 python-pybabel python-pychart python-pydot python-pyparsing python-  
pypdf python-reportlab python-requests python-simplejson python-tz python-unittest2  
python-vatnumber python-vobject python-werkzeug python-xlwt python-yaml wkhtmltopdf  
node-less
```

Explicación de algunas:

Python: es un lenguaje de programación interpretado, y esto significa que se escribe muy cercano al lenguaje humano que este interpretará antes de llegar al procesador. Actualmente está presente en mucha aplicación importantes entre ellas Odoo que lo utiliza, ya que gran parte está escrito con él. Por eso requerimos varias de sus librerías.

Wkhtmltopdf: es una librería que utilizaremos para imprimir en PDF nuestros documentos generados con el ERP. Esencial para sacar informes, facturas y demás.

Del resto, la mayoría son simples plugins de Python que requeriremos según usos de módulos

D. Ahora instalaremos el **servidor de Odoo**.

Primero instalamos GIT.

```
#apt-get install git
```

Cambiamos al usuario *odoo* para que sea propietario, ya que será el que ejecute todo el sistema.

```
#su - odoo -s /bin/bash
```

```
#cd ..
```

Ahora con el usuario correcto vamos a descargarlos y clonar una rama del código fuente de Odoo que está en GIT. Para eso lo instalamos antes. Como estamos en la carpeta */opt/odoo*, **nos subimos un nivel para que el clon se quede dentro de */opt/odoo* y no en */opt/odoo/odoo***.

```
$git clone https://www.github.com/odoo/odoo --depth 1 --branch 9.0 --single-branch2  
$exit
```

Puede llevar un **cierto tiempo**.

² Si falla vigila los dobles guiones

GIT es el comando principal de **control de versiones** y **Github** es el mayor proveedor de alojamiento de proyectos que actualmente está siendo utilizado por la comunidad del software libre para crear nuevas aplicaciones o bien colaborar en las existentes.

El **comando (git clone) descarga** y clona de la web que pongas después, en nuestro caso, (<https://www.github.com/odoo/odoo>) con la profundidad de carpetas (`-depth 1`) y la rama elegida (`-branch 9.0 -single-branch`).

E. Configurando Odoo.

Tenemos una configuración por defecto en (`/opt/odoo/debian/openerp-server.conf`), por lo que la usaremos de base y la copiaremos en `/etc` donde debe estar con los permisos que correspondan.

```
#cp /opt/odoo/debian/openerp-server.conf /etc/odoo-server.conf
```

```
#chown odoo: /etc/odoo-server.conf
```

```
#chmod 640 /etc/odoo-server.conf
```

NOTA.- si bajamos la versión 10 el fichero de configuración pasa a llamarse **odoo.conf**

Y ahora nos toca modificarlo para dejarlo a nuestro gusto y personalización adecuadas.

- **db_password = False.** Aquí tenemos que **cambiar (False) por la contraseña que creamos al crear la base de datos en el paso B.**
- **addons_path = /...** La ruta adecuada es la que acabamos de descargar de GIT, por lo que **cambiamos su contenido** por este (`/opt/odoo/addons`).

También podemos añadir con control de Logs, añadiendo al final del documento:

- **logfile = /var/log/odoo/odoo.log**

Y creando su carpeta:

```
#mkdir /var/log/odoo
```

```
#chown odoo:root /var/log/odoo/
```

Y aquí **finalizaría la instalación**

F. Probemos el servidor.

Ya tenemos todo listo para probar el sistema.

```
#su - odoo -s /bin/bash
```

`$/opt/odoo/openerp-server` (en la versión 10 la orden es `/opt/odoo/odoo-bin`)

Y nos vamos al navegador desde otra máquina y ponemos en la ruta la IP de nuestro server:8069

Ejemplo: 10.0.0.1:8069

Si vemos que no se conecta o da error una de las soluciones pasa por instalar el servidor web en nuestra máquina Debian, es decir, Apache2.

`#apt-get install apache2`

Si todo va bien nos tiene que salir el selector de bases de datos para crear **una nueva base de datos** Odoo. Esto se realiza por que Odoo le permite tener varias aplicaciones en un mismo servidor.

Aquí ya damos nombre a la base que queremos crear y las contraseñas para protegerla. Y ya tenemos nuestro Odoo funcionando.

Conexión desde una máquina MV cliente.

La MV de Ubuntu la configuramos en modo Red Interna y en DHCP para que tome una ip de la máquina Debian. Nos conectamos al servidor de la siguiente forma:

<http://10.0.0.100:8069>.



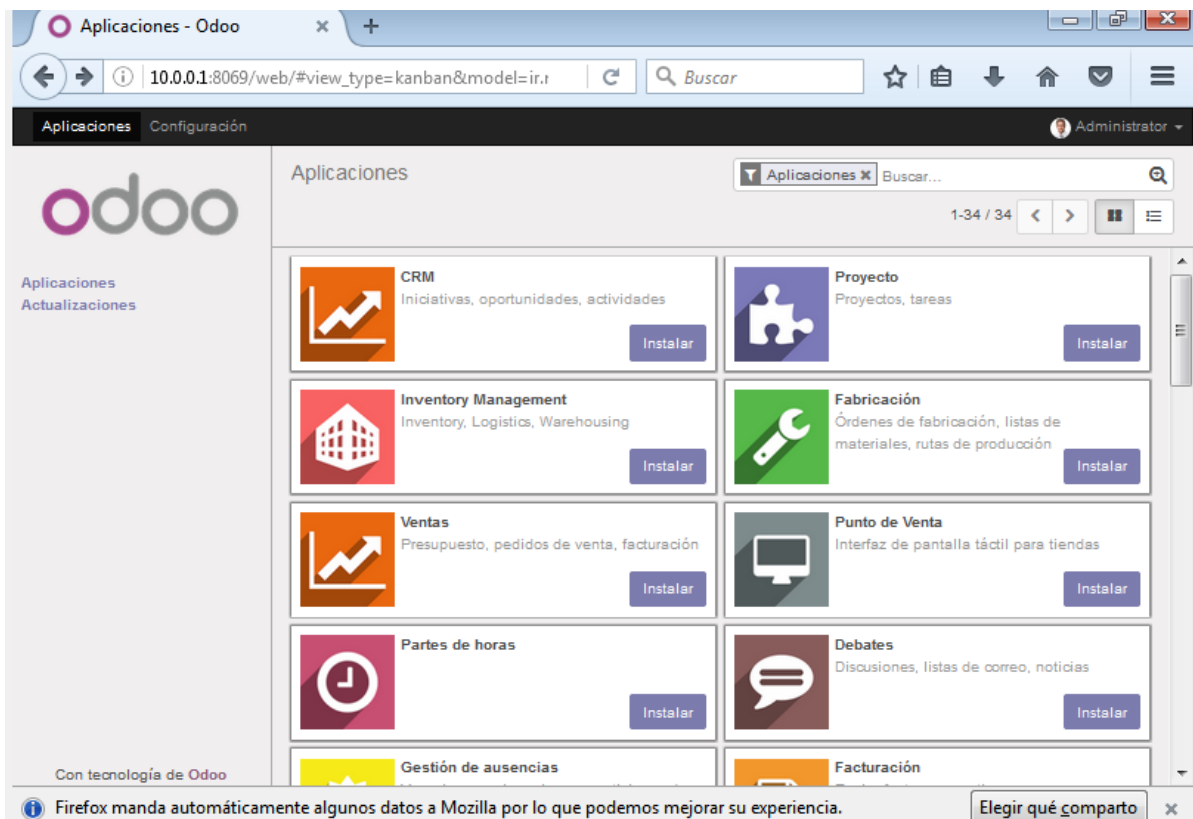
G. Creando BD de aplicación Odoo

Si marcamos la casilla **“Load demonstration data”** este cargara datos de prueba para visualizar el funcionamiento de Odoo y de los próximos módulos que habilite.

Esta pantalla aparecerá la primera vez que instalamos Odoo o luego que decida crear una nueva base de datos para una nueva aplicación Odoo. Para posterior acceso al este modulo consulte la URL: <http://10.0.0.100:8069/web/database/manager>.

H. Habilitando módulos disponibles de Odoo

Podemos habilitar los módulos disponibles para la aplicación Odoo tal como muestra la pantalla



De esta forma ya tiene instalado y configurado mínimamente el servidor Odoo y una aplicación web de Odoo, solo quedara realizar configuraciones básicas de la aplicación, creación la empresa, sus empleados, sus usuarios y demás procedimientos propios de cada modulo habilitado en esta aplicación para poder empezar a usar Odoo.

I. Detener servidor Odoo

Para detener el servidor Odoo, presionar la combinación de teclas **Ctrl + C** para enviar una segunda señal para forzar el cierre del proceso del servicio del servidor.

J. Algunas mejoras post-instalación

Podemos **dejar permanente y ejecutar de inicio el servidor Odoo**. Creamos el archivo, le damos permisos y actualizamos los índices de arranque.

```
#nano /etc/init.d/odoo-server  
  
#chmod 755 /etc/init.d/odoo-server  
  
#chown root: /etc/init.d/odoo-server  
  
#update-rc odoo-server defaults
```

Modificamos las rutas para utilizar nuestra configuración, con el siguiente código:

```
#nano /etc/init.d/odoo-server  
  
#!/bin/sh  
PATH=/bin:/sbin:/usr/bin  
DAEMON=/opt/odoo/openerp-server  
NAME=odoo-server  
DESC=odoo-server  
# Nombre del usuario administrador de odoo  
USER=odoo  
# Localización del fichero de configuración  
CONFIGFILE="/etc/odoo-server.conf"  
# obtenemos un identificador de proceso  
PIDFILE=/var/run/$NAME.pid  
# Opciones adicionales.  
DAEMON_OPTS="-c $CONFIGFILE"  
  
[-x $DAEMON ] || exit 0  
[-f $CONFIGFILE ] || exit 0  
  
echo -n "Lanzando ${DESC}: "  
start --quiet --pidfile ${PIDFILE} --chuid ${USER} --background --make-pidfile --exec ${DAEMON} -- $  
{DAEMON_OPTS}  
echo "${NAME}."
```

Otra opción cambiar el puerto de visión de Odoo para ello utilizamos IPTABLES

```
#nano /etc/rc.local
```

```
iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8069
```