# Instalando Odoo 11 on Fedora 25

$ sudo dnf upgrade -y

$  sudo dnf install postgresql-server postgresql-contrib -y

Una vez que termina la instalación (I) de PostgreSQL Server, procedemos a la configuración (S-setup), (“- -“) antes de initdb.

$ sudo postgresql-setup –initdb

Aunque la configuración por omisión del servicio permite se ejecute correctamente, es recomendable modificar unas líneas del archivo de configuración de /var/lib/pgsql/data/postgresql.conf:

Especificar la dirección (es), desde se puede acceder al servicio, para nuestro caso el servicio sólo atenderá conexiones de manera local.

listen\_addresses = ‘localhost’

Y en el puerto 5432, por lo que dichas líneas no les deberá preceder el símbolo de # y quedar como se muestra.

port = 5432

También realizaremos cambios en /var/lib/pgsql/data/pg\_hba.conf, busqué dentro del archivo y modifique para que las líneas queden de la siguiente manera.

# TYPE  DATABASE        USER            ADDRESS                 METHOD

# “local” is for Unix domain socket connections only  
local   all             postgres                                peer  
# IPv4 local connections:  
host    all             all             127.0.0.1/32            md5  
# IPv6 local connections:  
host    all             all             ::1/128                 md5.

Ahora habilitamos (E-enable) el servicio.

$ sudo systemctl start postgresql ; systemctl enable postgresql.

Es momento de iniciar sesión en postgresql para saber si todo está bien, probar (T- test).

$ sudo -s

# su – postgres

$ psql

psql (9.5.6)  
Digite «help» para obtener ayuda.

postgres=#

Ahora vamos a generar una base de datos.

#  CREATE ROLE miusuario WITH LOGIN CREATEDB PASSWORD ‘m1passw0rd’;

# CREATE DATABASE midb owner miusuario;

#\q

Esto nos regresa al prompt de postgres $.

$ psql -U miusuario -h localhost -d midb -W  
Contraseña para usuario miusuario:  
psql (9.5.6)  
Digite «help» para obtener ayuda.  
midb=>

Esto es prueba de que nuestro PostgreSQL Server se encuentra saludable y funcionando.

Ahora es momento de instalar Odoo, agregando el repositorio a nuestro server, (“- -“) antes de add-repor.

$ sudo dnf config-manager  –add-repo=https://nightly.odoo.com/10.0/nightly/rpm/odoo.repo

$ sudo dnf install -y odoo wkhtmltopdf python2-xlrd.noarch

Al terminar la instalación y previo a iniciar a Odoo cómo servicio, debemos configurar los parámetros correspondientes, en /etc/odoo/odoo.conf. Cambie los valores con los datos que hemos generado de ROL en PostgreSQL en los pasos anteriores.  
[options]  
; This is the password that allows database operations:  
; admin\_passwd = admin  
db\_host = localhost <– La BD se encuentra en el mismo servidor.  
db\_port = 5432 <– Puerto de PostgreSQL.  
db\_user = miusuario <–Rol que creamos dentro de PostgreSQL.  
db\_password =  m1passw0rd <– Contraseña asiganda al rol que generamos en los pasos anteriores para el rol.  
addons\_path = /usr/lib/python2.7/site-packages/odoo/addons

Es momento de iniciar Odoo.

$ sudo systemctl start odoo.service  
Consultemos si el servicio está activo.

$ sudo systemctl status odoo.service  
● odoo.service – Odoo Open Source ERP and CRM  
Loaded: loaded (/usr/lib/systemd/system/odoo.service; disabled; vendor preset: disabled)  
Active: active (running) since dom 2017-03-12 18:53:08 CST; 5s ago <– active (running) indica que el servicio esta OK.  
Main PID: 17050 (odoo)  
Tasks: 4 (limit: 4915)  
CGroup: /system.slice/odoo.service  
└─17050 /usr/bin/python /usr/bin/odoo –config /etc/odoo/odoo.conf –logfile /var/log/odoo/odoo-server.log

mar 12 18:53:08 camila.larenata.com systemd[1]: Started Odoo Open Source ERP and CRM.

Consultemos que los puertos estén activos.

$ sudo ss -atun | grep 8069

tcp    LISTEN     0      128       \*:8069                  \*:\*  
Nos interesa encontrar el puerto  8069/tcp que es el Odoo.

Ahora agregamos el puerto a nuestro firewall.

$ sudo firewall-cmd –add-port=8069/tcp  
success

$ sudo firewall-cmd –add-port=8069/tcp –permanent  
success

En otra terminal puede supervisar la actividad de Odoo,

$ tailf /var/log/odoo/odoo-server.log

