Installing Odoo 13.0

Install Odoo

$ export LC\_ALL=C

$ sudo apt-get update -y

$ sudo apt-get upgrade -y

$ sudo apt install postgresql -y

$ wget -O - https://nightly.odoo.com/odoo.key | apt-key add -

$ echo "deb http://nightly.odoo.com/13.0/nightly/deb/ ./" >> /etc/apt/sources.list.d/odoo.list

$ apt-get update && apt-get install odoo -y

$ apt-get upgrade -y

Install wkhtmltopdf

$ wget https://github.com/wkhtmltopdf/wkhtmltopdf/releases/download/0.12.5/wkhtmltox\_0.12.5-1.bionic\_amd64.deb

$ sudo apt install ./wkhtmltox\_0.12.5-1.bionic\_amd64.deb -y

$ sudo service odoo restart

OR

$ sudo apt install wkhtmltopdf

Install nginx

$ sudo apt-get install nginx -y

$ sudo vim /etc/nginx/conf.d/odoo.conf and replace the following

#odoo server

upstream odoo {

server 127.0.0.1:8069;

}

upstream odoochat {

server 127.0.0.1:8072;

}

server {

# listen 443;

# server\_name mekdam.humanplus.app;

listen 80 default\_server;

server\_name 213.219.39.246;

proxy\_read\_timeout 720s;

proxy\_connect\_timeout 720s;

proxy\_send\_timeout 720s;

client\_max\_body\_size 0;

# Add Headers for odoo proxy mode

proxy\_set\_header X-Forwarded-Host $host;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

proxy\_set\_header X-Real-IP $remote\_addr;

# log

access\_log /var/log/nginx/odoo.access.log;

error\_log /var/log/nginx/odoo.error.log;

# Redirect requests to odoo backend server

location / {

proxy\_redirect off;

proxy\_pass http://odoo;

}

location /longpolling {

proxy\_pass http://odoochat;

}

# common gzip

gzip\_types text/css text/less text/plain text/xml application/xml application/json application/javascript;

gzip on;

}

$ vim /etc/nginx/sites-enabled/default

Comment all lines in this file

$ nginx -t

$ sudo service nginx reload

Shoule show syntax is ok

$ vim /etc/odoo/odoo.conf

Add the following values if the server is 4 CPU and 8 GRam, this configuration will run 48 concurrent users perfectly:

addons\_path = /usr/lib/python2.7/dist-packages/odoo/addons,/var/lib/odoo/ihuman/addons,/var/lib/odoo/ihuman/third\_party\_modules  
workers = 8

proxy\_mode = True

limit\_memory\_hard = 1677721600

limit\_memory\_soft = 629145600

limit\_request = 8192

limit\_time\_cpu = 600

limit\_time\_real = 1200

max\_cron\_threads = 1

Add the following values if the server is 2 CPU and 4 GRam, this configuration will run about 20 concurrent users perfectly:

(\*\*Note) This need to be tested

addons\_path = /usr/lib/python2.7/dist-packages/odoo/addons,/var/lib/odoo/ihuman/addons,/var/lib/odoo/ihuman/third\_party\_modules

workers = 6

proxy\_mode = True

limit\_memory\_hard = 1677721600

limit\_memory\_soft = 629145600

limit\_request = 5461

limit\_time\_cpu = 600

limit\_time\_real = 1200

max\_cron\_threads = 1

$ sudo service odoo restart

After restarting nginx and odoo you will find problem in poll, do the following to fix poll issue

$ sudo apt-get install python3-pip -y

You may get error while using pip >> locale.Error: unsupported locale setting use the command export LC\_ALL=C to solve it

$ sudo pip3 install gevent

$ sudo pip3 install psycogreen

$ sudo service odoo restart

Certificate with certbot and certification hook

$ sudo apt-get update

$ sudo apt-get install software-properties-common -y

$ sudo add-apt-repository universe

$ sudo add-apt-repository ppa:certbot/certbot

sudo apt-add-repository -r ppa:certbot/certbot

$ sudo apt-get update

$ sudo apt-get install certbot python3-certbot-nginx -y

$ sudo certbot --nginx  
This will ask you to certify some domains, if you like to certify domain do this: sudo certbot --nginx -d sub.example.com

$ systemctl list-timers

If you want to certify wildcard, all subdomains from your domain

$ apt install certbot

$ certbot certonly --manual --preferred-challenges=dns --email hazem.abdalazeem@gmail.com --agree-tos -d \*.silverwings-tr.com

Auto Backup with rsnapshot

$ sudo apt-get install rsnapshot

$ vim /etc/rsnapshot.conf

Change the lines:

23 snapshot\_root /var/lib/odoo/backups/

93 retain daily 3

94 retain weekly 2

95 retain monthly 1

128 lockfile /var/lib/odoo/rsnapshot.pid

239 backup\_exec /usr/bin/curl -X POST -F "master\_pwd=Mekdam@12345" -F "name=Production" -F "backup\_format=zip" -o "/backups/Mekdam\_$(date +'%F').zip" "http://localh ost:8069/web/database/backup"

240 backup /backups/ .

241 backup\_exec /bin/rm -rf /backups/\*

$ mkdir /var/lib/odoo/backups/

$ chown -R odoo: backups/

$ cd

$ mkdir /backups

$ chown -R odoo: /backups

$ vim /etc/systemd/system/rsnapshot@.service

[Unit]

Description=rsnapshot (%I) backup

[Service]

Type=oneshot

Nice=19

IOSchedulingClass=idle

ExecStart=/usr/bin/rsnapshot %I

User=odoo

$ vim /etc/systemd/system/rsnapshot-daily.timer

[Unit]

Description=rsnapshot daily backup

[Timer]

# 01:00 is the clock time when to start it

OnCalendar=01:00

Persistent=true

Unit=rsnapshot@daily.service

[Install]

WantedBy=timers.target

$ vim /etc/systemd/system/rsnapshot-weekly.timer

[Unit]

Description=rsnapshot weekly backup

[Timer]

# Run once per week on Saturday at 2:00, after daily runs

OnCalendar=Saturday \*-\*-\* 02:00:00

Persistent=true

Unit=rsnapshot@weekly.service

[Install]

WantedBy=timers.target

$ vim /etc/systemd/system/rsnapshot-monthly.timer

[Unit]

Description=rsnapshot monthly backup

[Timer]

# Run once per month at 3:00 UTC, after daily and weekly runs

OnCalendar=\*-\*-1 03:00:00

Persistent=true

Unit=rsnapshot@monthly.service

[Install]

WantedBy=timers.target

$ systemctl daemon-reload

$ systemctl enable --now rsnapshot-{daily,weekly,monthly}.timer

$ systemctl status rsnapshot-monthly.timer

$ sudo -Hu odoo rsnapshot daily