|  |
| --- |
| [ERP系统部署文档] |
| 内部培训资料 |
|  |
| **软件部-陈向明** |
| **2017-5-4** |

|  |
| --- |
| 视频系统部署文档，用于提高内部IT人员对此系统环境的快速搭建，提高工作效率。属内部培训资料，谢绝外部转载！ |

目 录

[第1章 环境准备 1](#_Toc481678013)

[第2章 安装与部署 1](#_Toc481678014)

[2.1 数据库安装 1](#_Toc481678015)

[2.2 安装odoo 2](#_Toc481678016)

[2.3 Nginx安装 3](#_Toc481678017)

[第3章 日常运维要点 5](#_Toc481678018)

[3.1 排错思路 5](#_Toc481678019)

[3.2 故障报告 5](#_Toc481678020)

[3.3 远程支援 5](#_Toc481678021)

[3.4 小结 5](#_Toc481678022)

# 环境准备

* 系统初始化
* 主机与目录规划
* 系统与软件版本管理

# 安装与部署

## 数据库安装

# 安装pg

apt-get install postgresql

# su到postgres用户

su - postgres

# 进入pg控制台

psql

# 创建数据库用户

CREATE USER odoo WITH PASSWORD 'lty8547@' CREATEDB ; #用户名密码自己设置此处只做示范

# 退出pg控制台

\q

# 退出postgres用户

exit

mkdir .pip

touch .pip/pip.conf

apt install postgresql-server-dev-all

# 把下面这段配置写到pip.conf里面

[global]

index-url=https://pypi.douban.com/simple

# 安装python pip

apt install python-pip

apt install python-dev

apt-get install libxml2 libxml2-dev libxslt1-dev libjpeg-dev libldap2-dev libsasl2-dev

# 安装nodejs

apt install nodejs

# 升级nodejs

#nodejs升级不是必备过程 如果默认nodejs版本低于6.0 那么就升级

npm cache clean -f

npm install -g n

n stable

ln -sf /usr/local/n/versions/node/7.8.0/bin/node /usr/bin/node

#软连接目标路径根据实际升级的版本不同而不同 视具体情况而定

# 安装 npm

apt-get install npm

# 安装 less

npm install -g less

## 安装odoo

#上传odoo代码

cd 代码文件夹

#安装python包

pip install -r requirements.txt

# odoo.conf 文件配置信息 配置项的值识具体情况而定

[options]

; This is the password that allows database operations:

; admin\_passwd = admin

db\_host = 127.0.0.1 #根据实际数据库部署信息设置

db\_port = 5432 #根据实际数据库部署信息设置

db\_user = odoo #根据实际数据库部署信息设置

db\_password = lty8547@ #根据实际数据库部署信息设置

#addons\_path 根据实际的odoo源码部署路径来设置（/path/to 为源代码实际目录）

addons\_path = /path/to/addons,/path/to/extend\_addons,/path/to/odoo/addons

max\_cron\_threads = 1 #定时任务线程数 根据系统负载设置

workers = 8 #worker数 根据系统负载和服务器cpu数量配置

proxy\_mode = True

limit\_time\_real=600

# xmlrpc\_interface 表示odoo服务器监听ip

xmlrpc\_interface = 127.0.0.1

# 启动odoo服务器

python odoo-bin -c odoo.conf

## Nginx安装

apt install nginx

# 创建文件

/etc/nginx/sites-enabled/ltyodootest #文件名自由选择 此处仅做示范

# 以下是配置文件内容，配置文件里面的端口号根据实际情况定

upstream odoo {

server 127.0.0.1:8069; #ip为odoo服务器运行所在主机的监听IP 即上面配置 的xmlrpc\_interface 的值

}

upstream odoochat {

server 127.0.0.1:8072; #ip为odoo服务器运行所在主机的监听IP 即上面配置 的xmlrpc\_interface 的值

}

# http -> https

server {

listen 80;

}

server {

#listen 443;

#server\_name 可以是一个可解析的域名

server\_name ltyodootest.com, [www.ltyodootest.com;](http://www.ltyodootest.com;)

proxy\_read\_timeout 720s;

proxy\_connect\_timeout 720s;

proxy\_send\_timeout 720s;

# 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;

# SSL parameters

#ssl on;

#ssl\_certificate /etc/ssl/nginx/server.crt;

#ssl\_certificate\_key /etc/ssl/nginx/server.key;

#ssl\_session\_timeout 30m;

#ssl\_protocols TLSv1 TLSv1.1 TLSv1.2;

#ssl\_ciphers 'ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-AES128-GCM-SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-AES128-SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-SHA:DHE-RSA-AES256-SHA:AES128-GCM-SHA256:AES256-GCM-SHA384:AES128-SHA256:AES256-SHA256:AES128-SHA:AES256-SHA:AES:CAMELLIA:DES-CBC3-SHA:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!PSK:!aECDH:!EDH-DSS-DES-CBC3-SHA:!EDH-RSA-DES-CBC3-SHA:!KRB5-DES-CBC3-SHA';

#ssl\_prefer\_server\_ciphers on;

# log

access\_log /var/log/nginx/odoo.access.log; #nginx访问该服务的访问日志文件

error\_log /var/log/nginx/odoo.error.log; #nginx访问该服务的错误日志文件

# 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;

}

#重启服务器

service nginx restart

# 日常运维要点

## 排错思路

* 分析日志
* 根据日志错误，解决故障

## 故障报告

* 如非运维人员，报故障时不要急，首先需冷静
* 简明扼要的报告故障现象
* 发日志，或将错误截图发给运维排查

## 远程支援

* 此项针对非运维人员
* 电话沟通
* QQ，termview等远程工具寻求技术支援

## 小结