**开创啦系统安装配置手册**

文档修订

|  |  |  |  |
| --- | --- | --- | --- |
| **版本号** | **修订内容** | **修订日期** | **修订人** |
| 1.0 | 创建文档 | 2017/04/19 | 章传胜 |
| 2.0 | 加入nginx,redis | 2017/05/10 | 章传胜 |
|  |  |  |  |
|  |  |  |  |

# 文档目录

# 选择安装介质

# 概要说明

目前项目访问流程为：用户请求-->域名解析到203.130.41.41（本项目的互联网ip）-->nginx服务器10.1.83.129/10.159.191.130-->web服务器1--4 -->mysql，获得数据。

其中部分图片和文件采用ftp后台上传，前台展示，ftp服务器配置了2台，互相同步文件，每分钟同步一次。

Mysql采用双主机户为主从的配置，连接任意一台均可获得数据。每台mysql服务器均配置了nginx，启用3307端口来转发向2台mysql服务器的3306端口发起的请求。

前台web服务器4台，均是独立服务模式，采用nginx做负载均衡和静态缓存服务。

# 安装准备

在安装部署前，需要为开创啦服务创建执行用户。

以root用户登录，执行以下命令创建oseasy用户，通过passwd命令指定用户登录密码。

|  |
| --- |
| > adduser oseasy  > passwd oseasy |

最好设置一下系统的ulimit

以root用户登录 ，使用文本编辑器打开 /etc/security/limits.conf

|  |
| --- |
| > vi /etc/security/limits.conf  加入 \* - nofile 30000  修改后重启 （重启命令:reboot）  >ulimit -n  查看是否修改成功。（如果是30000，则修改成功） |

查看 linux版本

#cat /etc/redhat-release

CentOS Linux release 7.2.1511

# 查看内核/操作系统/CPU信息

uname –a

#ifconfig

如果不是命令

请安装ifconfig命令行

# yum install net-tools

完成后 #ifconfig

查看网卡是否是eth0 如果不是，本次版本需要修改 网卡为eth0

![C:\Users\victor\Documents\Tencent Files\3040652\Image\C2C\9{YTBUCY`](U}T@_5T`08R9.png](data:image/png;base64,)

# Jdk的安装

以root用户登录. 移除旧版jdk

|  |
| --- |
| find / -name javac -print & 查看是否装有jdk  rpm -qa | grep jdk 查看  rpm -qa | grep gcj 查看  yum -y remove java-1.8.0-\* 移除旧版jdk |

安装jdk

|  |
| --- |
| tar xvf jdk-8u121-linux-x64.tar.gz #解压  mv jdk1.8.0\_121 /usr/local  cd /usr/local/  vi + /etc/profile |

在文件最后插入

|  |
| --- |
| export JAVA\_HOME=/usr/local/jdk1.8.0\_121  export CLASSPATH=.:$JAVA\_HOME/jre/lib/rt.jar:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar  export PATH=$PATH:$JAVA\_HOME/bin |

保存后

source /etc/profile

查看jdk是否安装成功

java –version

|  |
| --- |
| java version "1.8.0\_121"  Java(TM) SE Runtime Environment (build 1.8.0\_121-b13)  Java HotSpot(TM) 64-Bit Server VM (build 25.121-b13, mixed mode) |

安装成功

# ftp安装(单机)

在线安装

## **卸载之前安装过的vsftd**

yum remove vsftpd

## **关掉iptables和selinux**

|  |
| --- |
| vi /etc/selinux/config  SELINUX=enforcing改为SELINUX=disabled |

**重启机器**查看selinux状态

/usr/sbin/sestatus –v

## **安装**vsftpd

**安装vsftpd服务器**： rpm -ivh vsftpd-\*.rpm 或从互联网寻找对应资源直接安装vsftpd：

yum -y install vsftpd

|  |
| --- |
| rpm -ivh vsftpd-3.0.2-10.el7.x86\_64.rpm 或  yum -y install vsftpd |

装完后，配置文件/etc/vsftpd/vsftpd.conf ，去掉注释chroot\_local\_user=YES，让这条配置生效，固定用户在主目录下

参考提供的vsftpd.conf的配置

|  |
| --- |
| useradd -d /userRemoteFilePath ftponly //创建用户ftponly并指定用户主目录为/userRemoteFilePath  usermod -s /sbin/nologin ftponly //设置ftponly用户只能ftp，不能telnet  chown -R ftponly.ftponly /userRemoteFilePath //给目录及子目录授予用户ftponly  passwd ftponly //设置用户ftponly的密码  chmod a-w /userRemoteFilePath/tool/ |

以上完成，即可用ftp客户端测试

service vsftpd start 测试是否安装成功 (或systemctl start vsftpd)

设置开机启动

chkconfig vsftpd on

查看状态

chkconfig --list|grep vsftpd

httpd的安装

|  |
| --- |
| yum install httpd -y #安装  chkconfig httpd on #开机启动  vi /etc/httpd/conf/httpd.conf  httpd.conf 配置参考提供的配置  #设置 Apache的默认文档路径的位置是在/var/www/html  DocumentRoot "/userRemoteFilePath/tool"  service httpd start #开启apache |

访问下 <http://192.168.0.105>看能否访问

验证是否可以访问

echo ‘test’ > /userRemoteFilePath/tool/1.log

访问<http://192.168.0.105/1.log>

如果[You don't have permission to access / on this server.](http://blog.csdn.net/default7/article/details/40162443) 说明

httpd.conf 没设置正确或者selinux 没有关闭

httpd.conf的设置请参考给出的例子

上传ftp图片.zip至/ userRemoteFilePath/tool/oseasy 这个目录。解压

|  |
| --- |
| unzip –o ftp图片.zip –d /userRemoteFilePath/tool/oseasy  chown -R ftponly.ftponly /userRemoteFilePath/tool/oseasy |

如果连接包验证失败：

在配置文件中加入：

allow\_writeable\_chroot=YES

# Mysql安装(单机)

## **删除默认MySQL**

|  |
| --- |
| rpm -qa|grep mysql //查询MySQL是否安装  rpm -e mysql //普通删除模式  rpm -e --nodeps mysql //强制删除模式，不删除依赖包 |

## **安装mysql**

|  |
| --- |
| tar xvf mysql-5.7.17-linux-glibc2.5-x86\_64.tar.gz  mv mysql-5.7.17-linux-glibc2.5-x86\_64 /usr/local/  cd /usr/local/  ln -s mysql-5.7.17-linux-glibc2.5-x86\_64 mysql  cp my.cnf /etc/ #my.cnf 用已经编辑好的  mkdir -p /usr/local/mysql/data  mkdir -p /usr/local/mysql/log/  echo '' > /usr/local/mysql/log/error.log  echo '' >/usr/local/mysql/log/slow-query.log  groupadd oseasy  ####useradd -g oseasy oseasy  chown -R oseasy:oseasy /usr/local/mysql/ |

初始化数据库 这一步会产生一个随机密码

|  |
| --- |
| cd /usr/local/mysql  ./bin/mysqld --user=oseasy --basedir=/usr/local/mysql/ --datadir=/usr/local/mysql/data --initialize |

如果报

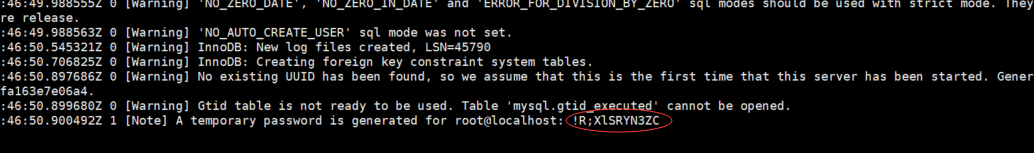
error while loading shared libraries: libaio.so

缺少安装包libaio和libaio-devel.

yum install libaio\*

成功后查看

tail -200f /usr/local/mysql/log/error.log



会查看到随机密码，修改密码时需要用到

## **启动mysql 切换到mysql用户**

|  |
| --- |
| #su oseasy  ./mysqld\_safe --user=oseasy &  ps -ef|grep mysql 检查是否启动成功 |

连接客户端

|  |
| --- |
| ./mysql -uroot –p  输入刚才保存的临时密码  >set password=password('新密码'); |

## **设置远程访问**

|  |
| --- |
| cd /usr/local/mysql/bin  mysql –uroot –p |

如果报错：

ERROR 2002 (HY000): Can’t connect to local MySQL server through socket ‘/var/lib/mysql/mysql.sock

权限问题：

|  |
| --- |
| chown –R oseasy:oseasy /var/lib/mysql/  chmod 777 –R /var/lib/mysql/ |

设置用户 root 可以在任意 IP 下被访问：

grant all privileges on \*.\* to root@"%" identified by "新密码";

刷新权限使之生效

flush privileges;

exit;

此时使用远程机器进行访问

## **添加服务系统服务**

|  |
| --- |
| cp -a ./support-files/mysql.server /etc/init.d/mysqld  su oseasy  service mysqld start  ps –ef|grep mysql |

如果出现Starting MySQL...The server quit without updating PID file这样的问题，

用chown -R oseasy:oseasy /usr/local/mysql/

请切换oseasy用户，权限问题引起的

## **数据的导入导出**

方法一

cd /usr/local/mysql/bin

./mysql -uroot -p123456 #连接mysql

mysql>create database **pro\_creativecloud**;

mysql>use **pro\_creativecloud**;

mysql>source /home/abc/abc.sql;

方法二

mysql -u用户名 -p密码 数据库名 < 数据库名.sql

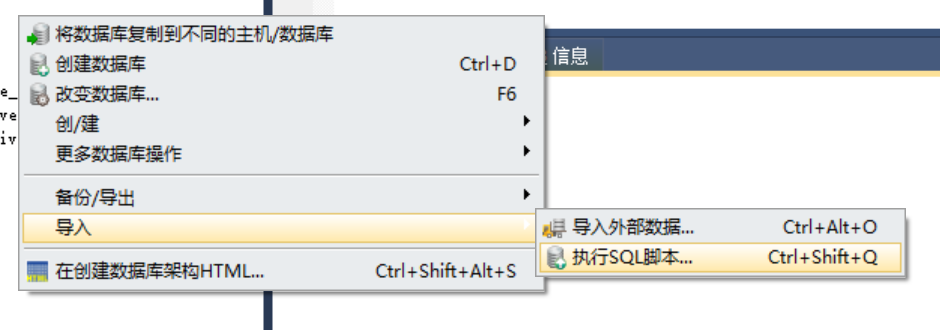
# mysqlimport -u root -p123456 < 20170423\_3.sql

方法三

用可视化工具SQLyog导入

安装sqlyong.连接上mysql





指定给出的sql文件导入。

# redis的安装（单机）

## **安装redis**

上传redis-3.0.7.tar.gz至服务器,解压至/usr/local/redis-3.0.7

|  |
| --- |
| tar zxvf redis-3.0.7.tar.gz /usr/local  ln –s redis-3.0.7.tar.gz redis  cd /usr/local/redis  make |

二进制文件是编译完成后在src目录下，通过下面的命令启动Redis服务

# nohup ./redis-server & ###启动redis

你可以使用内置的客户端命令redis-cli进行使用：

# src/redis-cli

redis> set foo bar

OK

redis> get foo

"bar"

如果make时出现zmalloc.h:50:31: 错误：jemalloc/jemalloc.h：没有那个文件或目录

请

make MALLOC=libc

# tomcat服务器安装（单机）

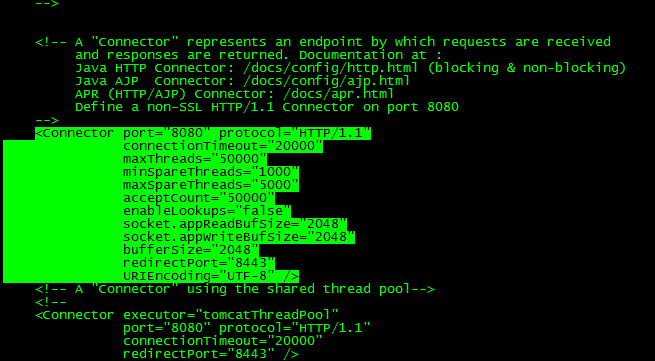
## **安装tomcat**

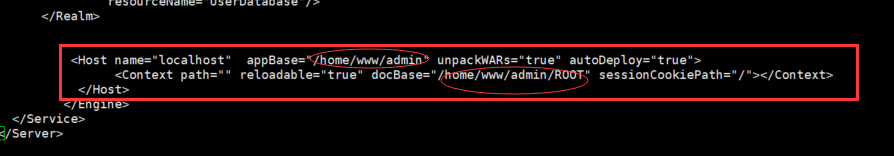
上传apache-tomcat-8.5.11.tar.gz 至服务器。解压至/usr/local/ apache-tomcat-8.5.11

|  |
| --- |
| tar zxvf apache-tomcat-8.5.11.tar.gz /usr/local  ln –s apache-tomcat-8.5.11 tomcat |

## **Tomcat配置**

修改/usr/local/tomcat/conf/server.xml

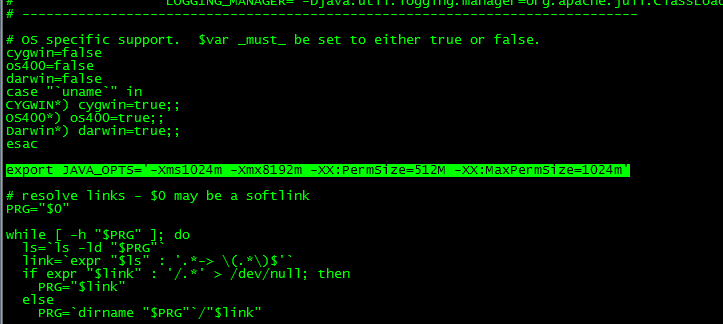




详细参考：server.xml 文件，已在目录包中提供

以上为当前配置值。

Jvm配置：/usr/local/tomcat/bin/catalina.sh



## **启动和关闭**

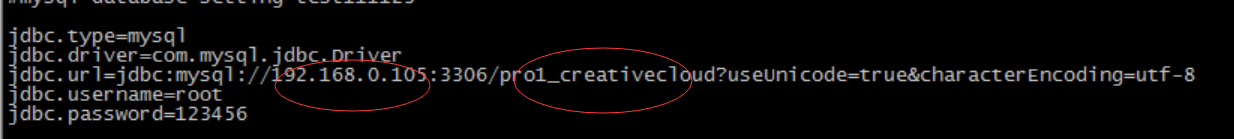
|  |
| --- |
| cd /usr/local/tomcat/bin  sh startup.sh start; #启动  ps –ef |grep tomcat  sh shutdown.sh ; #关闭  ps –ef |grep tomcat |

将程序的安装包ROOT.zip 至/home/www/admin/ 解压、最后的目录应该 /home/www/admin/ROOT

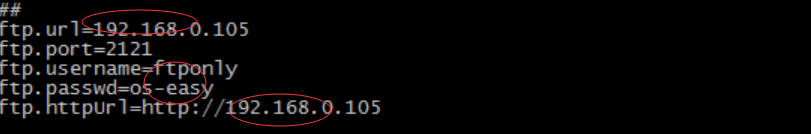
解压unzip –o ROOT.zip –d /home/www/admin/ROOT

cd /home/www/admin/ROOT/WEB-INF/classes #修改数据库 ftp连接地址

vi initiate.properties

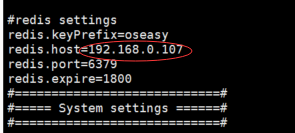


修改为对应的ip和pro\_creativecloud



修改ftp用户名密码和ip地址

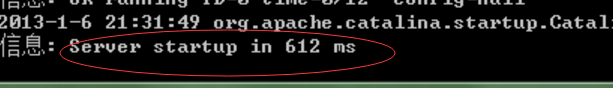
修改redis的地址



启动tomcat,查看日志

tail -200f logs/catalina.out

看到如下图的内容说明tomcat启动成功。



访问

<http://ip:port/a> 后台

<http://ip:port/f> 前台

# 导入许可

启动后访问后台<http://ip:port/a> 按要求导入授权，授权成功后。在次访问<http://ip:port/a> 即可。

# Tomcat集群配置

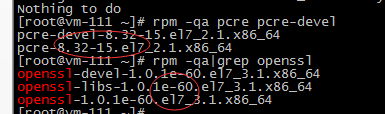
## **nginx安装**

1. 安装之前首先确认系统中是否已安装gcc、openssl-devel、pcre-devel、zlib-devel

|  |
| --- |
| rpm -qa pcre pcre-devel    rpm -qa|grep openssl |

安装必备工具

|  |
| --- |
| yum -y install gcc gcc-c++ autoconf automake  yum -y install zlib zlib-devel openssl openssl-devel pcre-devel |

安装成功。

说明：

pcre: 用来作地址重写的功能。

zlib：nginx 的gzip模块,传输数据打包，省流量（但消耗资源）。

openssl：提供ssl加密协议。

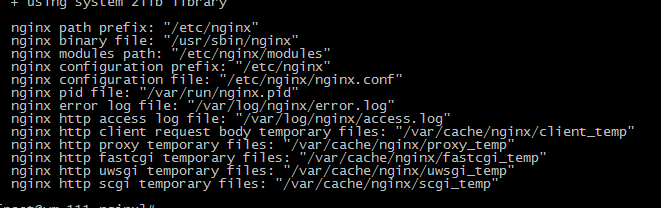
2）新建一个系统级用户组和匿名用户，以及下面编译时使用

|  |
| --- |
| groupadd -r nginx  useradd -s /sbin/nologin -g nginx -r nginx-user |

* 1. 编译安装Nginx

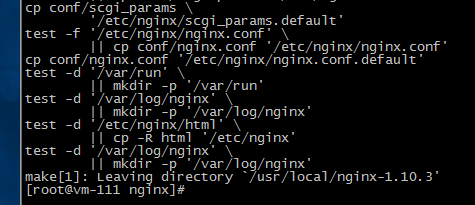
解压nginx包到/usr/local目录ll

|  |
| --- |
| make clean  ./configure \  --prefix=/etc/nginx \  --sbin-path=/usr/sbin/nginx \  --conf-path=/etc/nginx/nginx.conf \  --error-log-path=/var/log/nginx/error.log \  --http-log-path=/var/log/nginx/access.log \  --pid-path=/var/run/nginx.pid \  --lock-path=/var/run/nginx.lock \  --http-client-body-temp-path=/var/cache/nginx/client\_temp \  --http-proxy-temp-path=/var/cache/nginx/proxy\_temp \  --http-fastcgi-temp-path=/var/cache/nginx/fastcgi\_temp \  --http-uwsgi-temp-path=/var/cache/nginx/uwsgi\_temp \  --http-scgi-temp-path=/var/cache/nginx/scgi\_temp \  --user=nginx \  --group=nginx \  --with-http\_ssl\_module \  --with-http\_realip\_module \  --with-http\_addition\_module \  --with-http\_sub\_module \  --with-http\_dav\_module \  --with-http\_flv\_module \  --with-http\_mp4\_module \  --with-http\_gunzip\_module \  --with-http\_gzip\_static\_module \  --with-http\_random\_index\_module \  --with-http\_secure\_link\_module \  --with-http\_stub\_status\_module \  --with-http\_auth\_request\_module \  --with-mail \  --with-mail\_ssl\_module \  --with-file-aio \  --with-ipv6 \  --with-cc-opt='-O2 -g -pipe -Wp,-D\_FORTIFY\_SOURCE=2 -fexceptions -fstack-protector --param=ssp-buffer-size=4 -m64 -mtune=generic' |



完成后

make && make install

安装成功后没报错，启动nginx

|  |
| --- |
| nginx -c /etc/nginx/nginx.conf |

如果报错

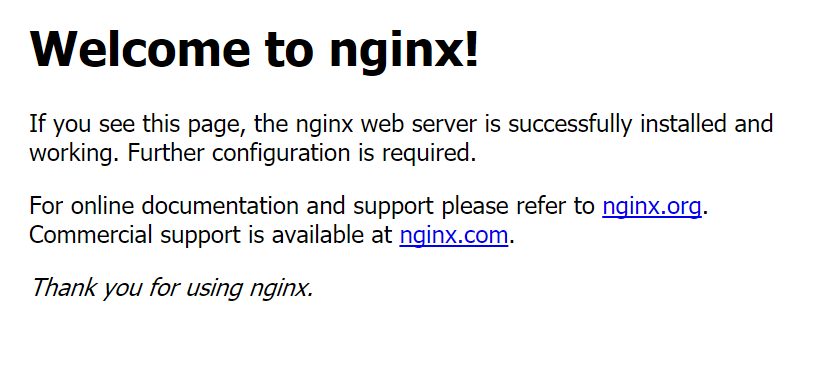
nginx: [emerg] getpwnam("nginx") faile

在nginx.conf中 把user nobody的注释去掉既可，如果提示

nginx: [emerg] mkdir() "/var/tmp/nginx/client\_temp" failed (2: No such file or directory)

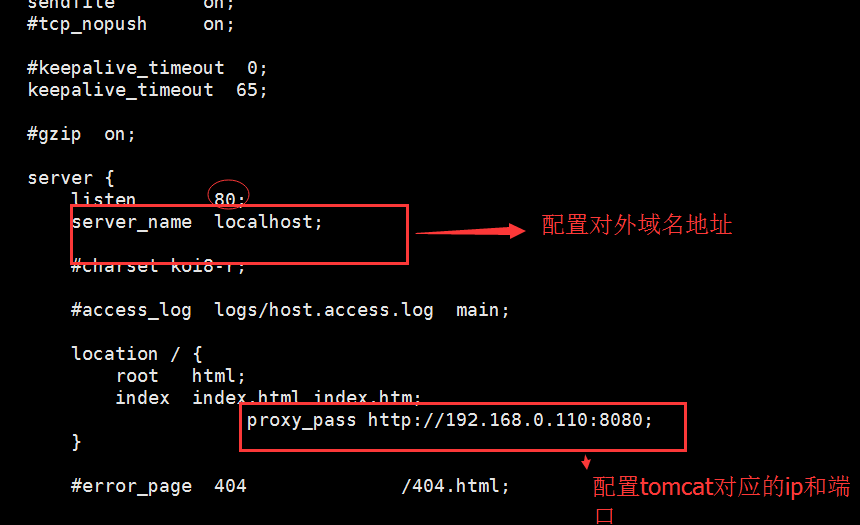
则用管理员去mkdir 创建文件夹后再启动，如果安装成功可以访问<http://ip>

mkdir –p /var/tmp/nginx/client\_temp

安装成功了。

## **nginx的配置**

配置参考nginx.conf



nginx -c /etc/nginx/nginx.conf -s reload

nginx –t 配置文件是否正确

可能用到的命令：

firewall-cmd --state 查看防火墙是否开启

firewall-cmd --permanent --add-service=ftp

firewall-cmd --reload

systemctl start vsftpd

systemctl enable vsftpd

firewall-cmd --permanent --add-service=httpd

firewall-cmd --permanent --add-port=3306/tcp

systemctl stop firewalld