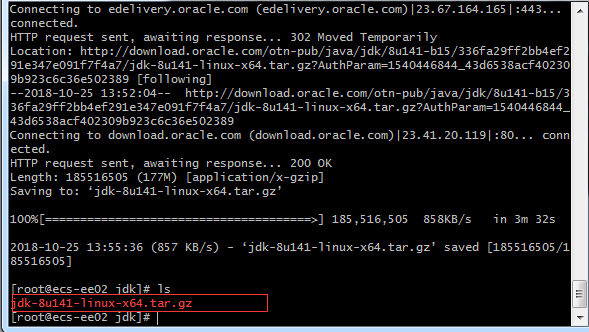
# 一 部署JDK

1 下载jdk 压缩包

下载时先进入压缩包想要下载的目录

下载路径：wget --no-cookies --no-check-certificate --header "Cookie: gpw\_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/8u141-b15/336fa29ff2bb4ef291e347e091f7f4a7/jdk-8u141-linux-x64.tar.gz"

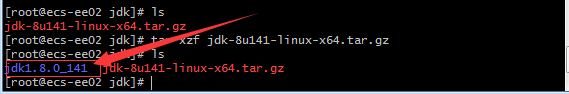




## 2 解压jdk压缩包

解压命令：tar xzf jdk-8u141-linux-x64.tar.gz

解压后可以看解压好的文件夹jdk1.8.0\_141



## 3 配置jdk环境变量

1 编辑/etc/profile 文件（系统环境配置） 文件在结尾添加环境配置

命令：1） vi /etc/profile 进入文件

1. i 进入编辑文件模式
2. Esc 退出编辑文件模式
3. Shift+：进入文件操作模式
4. Wq 回车 保存并退出
5. 退出连接，重新登录（不重新登录，可能安装成功，查看也不显示）

jdk配置内容添加完毕后，输入命令 " source /etc/profile " 重新加载配置文件

1. Java -version javac -version which java 查看jdk安装是否成功

环境参数： export 导入，JAVA\_HOME 变量名称 /usr/share/jdk1.6.0\_14

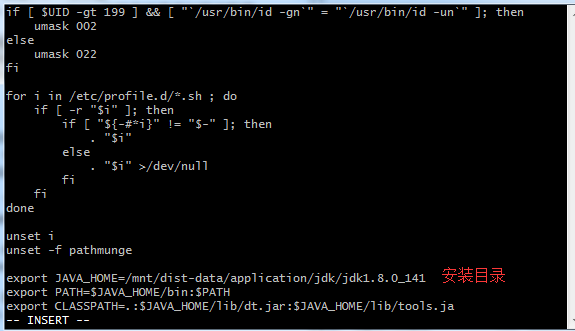
安装路径

8）文件在结尾添加环境配置

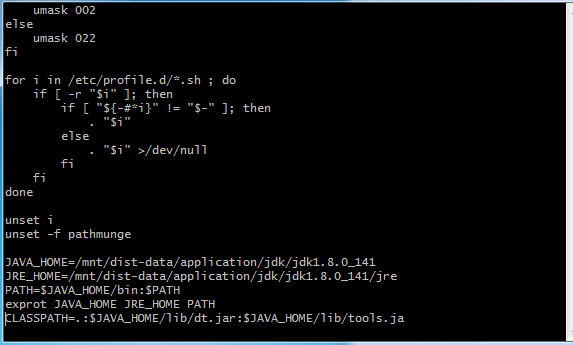
export JAVA\_HOME=/usr/share/jdk1.6.0\_14

export PATH=$JAVA\_HOME/bin:$PATH

export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.ja



下面这种方式导入环境变量也是可以的，先声明变量，然后导入声明



# 二 部署Tomcat

## 1 查库

1 去tomcat库查看想要下载的版本 （有些版本库里可能没有了，所有先看下确认下）

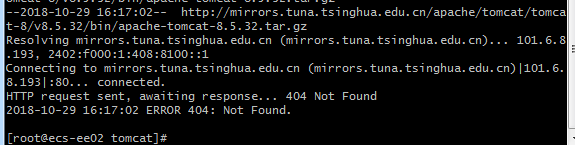
<https://mirrors.cnnic.cn/apache/tomcat/>

## 2 下载Tomcat

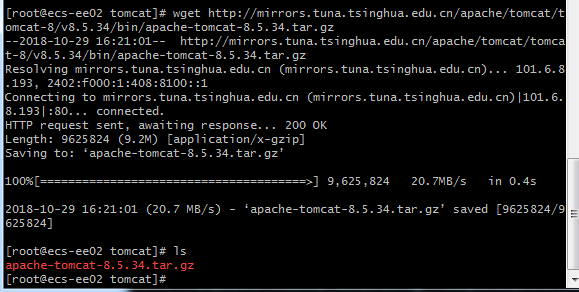
2 下载Tomcat（进入想要下载的目录）

wget [http://mirrors.tuna.tsinghua.edu.cn/apache/tomcat/tomcat-8/v8.5.34/bin/apache-tomcat-8.5.34.tar.gz](http://mirrors.tuna.tsinghua.edu.cn/apache/tomcat/tomcat-8/v8.5.32/bin/apache-tomcat-8.5.32.tar.gz)

如下图，8.5.32 版本的Tomcat库里已经没有了



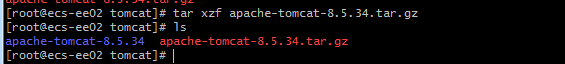
去库里可以看到版本8.5.34 有，下载就好



## 3 解压Tomcat包

3 解压安装Tomcat（当前压缩包目录）

tar xzf apache-tomcat-8.5.34.tar.gz

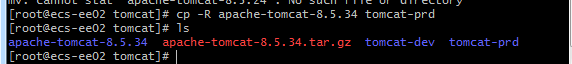


## 4 复制Tomcat

准备测试和生成环境

cp -R apache-tomcat-8.5.34 tomcat-dev（复制重命名为。。。）





## 5 添加Tomcat用户

有些文件操作权限需要区分下，例如上传文件，以Tomcat的用户身份上传的

1 groupadd tomcat 添加Tomcat分组

2 useradd -M -s /bin/nologin -g tomcat -d /mnt/disk-data/software/tomcat/tomcat-dev tomcat

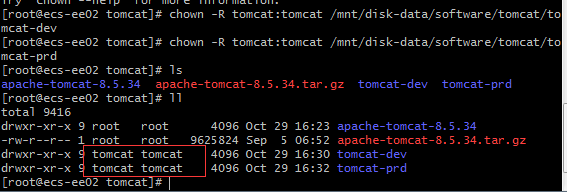
在Tomcat分组下添加用户



3 给目录赋予Tomcat用户管理

chown -R tomcat:tomcat /mnt/disk-data/software/tomcat/tomcat-dev

chown -R tomcat:tomcat /mnt/disk-data/software/tomcat/tomcat-prd



## 6 配置Tomcat服务（servcie）

### 1 命令配置

摘录：<https://blog.csdn.net/oaim_zh/article/details/78319863>

1 在Tomcat bin目录下创建setenv.sh文件

catalina.sh在执行的时候会调用同级路径下的setenv.sh来设置额外的环境变

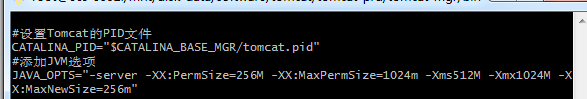
量，因此需要在/usr/local /tomcat/bin路径下创建setenv.sh文件。

如果配置此文件创建pid，service启动失败

命令：vi setenv.sh

这里因为在/etc/profile文件里已配置了此Tomcat的环境变量，这里可以直接引用

[zm@zm-centos7 ~]$ su *# 切换至root用户* 密码： [root@zm-centos7 zm]*# vim /usr/local/tomcat/bin/setenv.sh # 创建setenv.sh脚本* [root@zm-centos7 zm]*# cat /usr/local/tomcat/bin/setenv.sh # 查看setenv.sh脚本内容* export CATALINA\_HOME=/usr/local/tomcat export CATALINA\_BASE=/usr/local/tomcat *#设置Tomcat的PID文件* CATALINA\_PID="$CATALINA\_BASE/tomcat.pid" *#添加JVM选项* JAVA\_OPTS="-server -XX:PermSize=256M -XX:MaxPermSize=1024m -Xms512M -Xmx1024M -XX:MaxNewSize=256m"



命令配置其实也就是创建一个service文件，里面配置执行命令，然后导入service



echo "[Unit]" > /etc/systemd/system/$SERVICE\_NAME.service

echo " Description=$SERVICE\_NAME" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " After=syslog.target network.target" >> /etc/systemd/system/$SERVICE\_NAME.service

echo "[Service]" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Type=forking" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " User=tomcat" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Group=tomcat" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment=JAVA\_HOME=$JAVA\_HOME" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment=CATALINA\_PID=$TOMCAT\_HOME/temp/tomcat.pid" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment=CATALINA\_HOME=$TOMCAT\_HOME" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment=CATALINA\_BASE=$TOMCAT\_HOME" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment='JAVA\_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/./urandom'" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " ExecStart=$TOMCAT\_HOME/bin/startup.sh" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " ExecStop=/bin/kill -15 $MAINPID" >> /etc/systemd/system/$SERVICE\_NAME.service

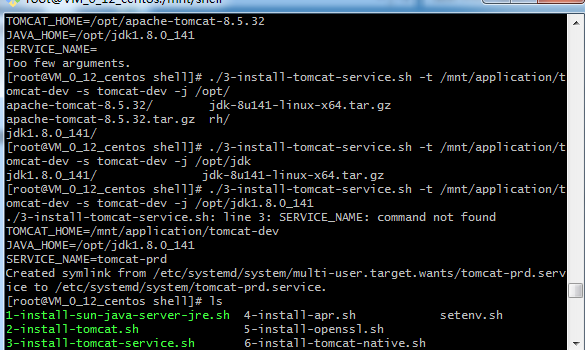
echo "[Install]" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " WantedBy=multi-user.target" >> /etc/systemd/system/$SERVICE\_NAME.service

`systemctl enable $SERVICE\_NAME`

### 2 脚本配置

下面是脚本内容



#!/bin/bash

while getopts "t:s:j:h" arg #选项后面的冒号表示该选项需要参数

do

case $arg in

t)

TOMCAT\_HOME=$OPTARG

;;

s)

SERVICE\_NAME=$OPTARG

;;

j)

JAVA\_HOME=$OPTARG

;;

h)

echo "usage:"

echo "./install-tomcat-service.sh -t TOMCAT\_HOME\_PATH -s SERVICE\_NAME -j JAVA\_HOME "

;;

?)

echo "usage:"

echo "./install-tomcat-service.sh -t TOMCAT\_HOME\_PATH -s SERVICE\_NAME -j JAVA\_HOME "

exit 1

;;

esac

done

echo "TOMCAT\_HOME=$TOMCAT\_HOME"

echo "JAVA\_HOME=$JAVA\_HOME"

echo "SERVICE\_NAME=$SERVICE\_NAME"

echo "[Unit]" > /etc/systemd/system/$SERVICE\_NAME.service

echo " Description=$SERVICE\_NAME" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " After=syslog.target network.target" >> /etc/systemd/system/$SERVICE\_NAME.service

echo "[Service]" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Type=forking" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " User=tomcat" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Group=tomcat" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment=JAVA\_HOME=$JAVA\_HOME" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment=CATALINA\_PID=$TOMCAT\_HOME/temp/tomcat.pid" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment=CATALINA\_HOME=$TOMCAT\_HOME" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment=CATALINA\_BASE=$TOMCAT\_HOME" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " Environment='JAVA\_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/./urandom'" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " ExecStart=$TOMCAT\_HOME/bin/startup.sh" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " ExecStop=/bin/kill -15 $MAINPID" >> /etc/systemd/system/$SERVICE\_NAME.service

echo "[Install]" >> /etc/systemd/system/$SERVICE\_NAME.service

echo " WantedBy=multi-user.target" >> /etc/systemd/system/$SERVICE\_NAME.service

`systemctl enable $SERVICE\_NAME`

### 3 系统服务配置

在之前的博文中已经对CentOS7下利用init.d目录下创建开机启动服务脚本，实现tomcat开机启动的介绍。但作为CentOS7来说，通过init.d建立启动脚本的机制已经不被推荐，推荐的方式是利用systemd机制实现开机启动，下面就该方式进行介绍。

摘录：https://blog.csdn.net/hello\_world\_qwp/article/details/78862277

1 将Tomcat注册为系统服务

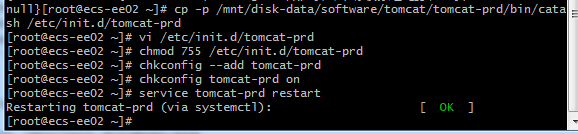
命令：cp -p /mnt/disk-data/software/tomcat/tomcat-dev/bin/catalina.sh /etc/init.d/tomcat



2 编辑启动文件

命令：vi /etc/init.d/tomcat-dev





# chkconfig: 112 63 37

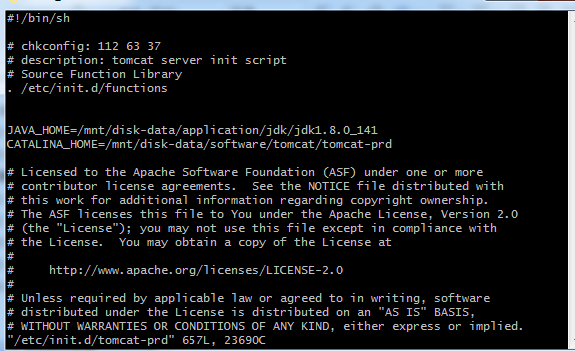
# description: tomcat server init script

# Source Function Library

. /etc/init.d/functions

JAVA\_HOME=/mnt/disk-data/application/jdk/jdk1.8.0\_141

CATALINA\_HOME=/mnt/disk-data/software/tomcat/tomcat-prd



摘录：

特别声明：复制的是 " catalina.sh "，我记得上次一个小师弟找我帮他搞，搞了我两天，最后我问他这儿复制的是那个，他给我说的是 " startup.sh " ，还给我解释说网上加都是 cp 的 " startup.sh " ，我当时就火帽三丈，当时差点就把电脑都给他砸了。

（所以一定要注意了，网上很多都是复制粘贴的，误人子弟，我这儿有图有真相！）

在第二行加入如下内容：

# chkconfig: 112 63 37

# description: tomcat server init script

# Source Function Library

. /etc/init.d/functions

JAVA\_HOME=/usr/local/jdk/

CATALINA\_HOME=/usr/local/tomcat

如下图：

注意：一定要注意JDK的路径呀，这个弄错了，搞起来非常麻烦的（亲身经历！！！）

注册Tomcat服务，依次输入

授权：

chmod 755 /etc/init.d/tomcat

添加到服务：

chkconfig --add tomcat

开机启动：

chkconfig tomcat on

如下图：

启动 Tomcat 服务，输入命令 "service tomcat stop/service tomcat start " ，出现 OK 的提示，说明Tomcat服务停止/启动成功，如下图：

关于 Linux下Tomcat8.0安装与配置图文教程 就写到这儿了。

最后再次声明，一定要注意路径，还有就是 cp 时注意文件的名称及大小写。

在本篇博文中我强调的够多了，如果你还是要配错误，那我也没办法了。

一定要注意细节问题，细节决定成败！！！

如果期间还遇到了什么问题，欢迎留言。。。

## 7 环境变量

**配置tomcat环境变量,目前还不知道配置有什么用，可以不配，这里只说明配置方法**

vi /etc/profile 文件在结尾添加环境配置

TOMCAT\_HOME=/opt/apache-tomcat-8.5.32

PATH=$PATH:$TOMCAT\_HOME/bin

export TOMCAT\_HOME PATH

## 8 Tomcat上传权限

Tomcat上传文件的默认权限是不可查看的，

摘录：<https://blog.csdn.net/hac99901431/article/details/77677236>



## 9 多台Tomcat服务器配置

### 1 tomcat配置

需求：多台Tomcat在一个服务器上同时运行

1 3个端口必须全部改掉，不然就会出现端口绑定冲突，也就是每个服务都有自己的端口占用



### 2 linux 配置环境变量

1 配置环境变量 2 修改catalina.sh 里启动环境

如果不配置，执行第二个Tomcat的shutdown.sh start.sh 等操作还是执行，第一个Tomcat

可能是catalina.sh配置没有改（没有亲测，网上这么说）





### 4 摘录

摘录：<https://blog.csdn.net/weixin_33446857/article/details/80918723（负载均衡）>

摘录：<https://blog.csdn.net/cheetahlover/article/details/78845553（代理多个）>

## 10 关闭Tomcat

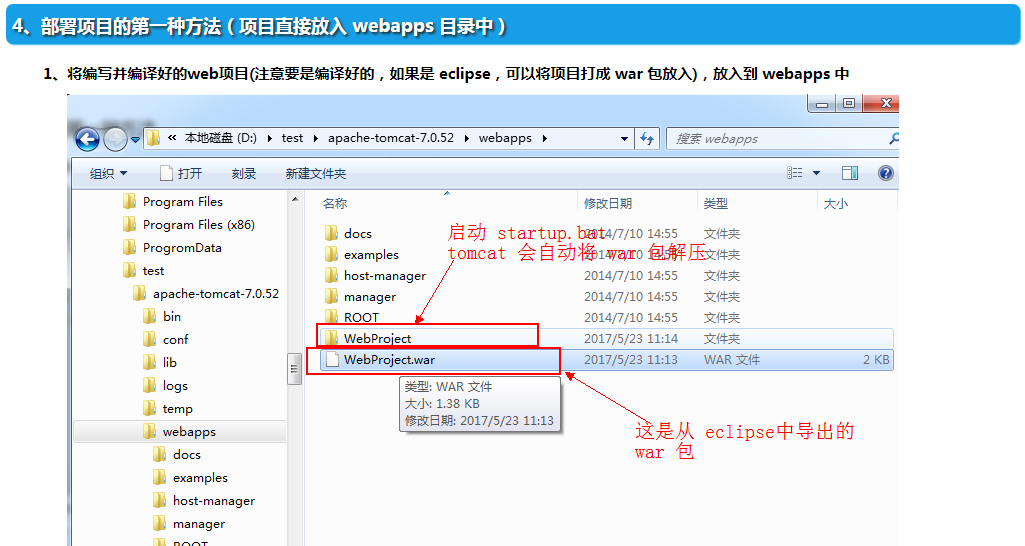
摘录：https://blog.csdn.net/wohiusdashi/article/details/81026900



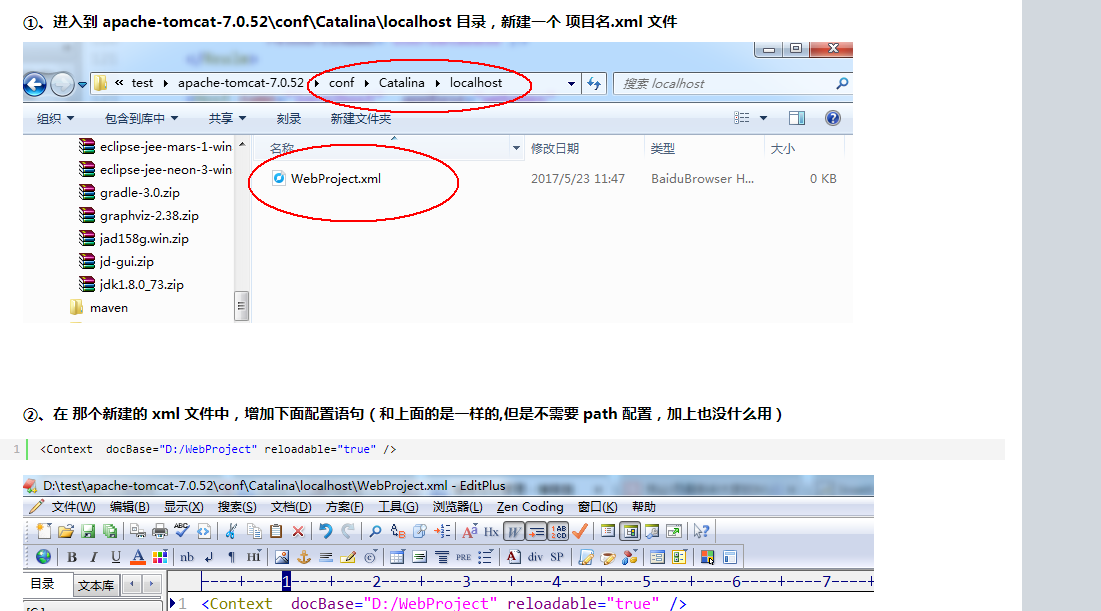
## 11 部署项目

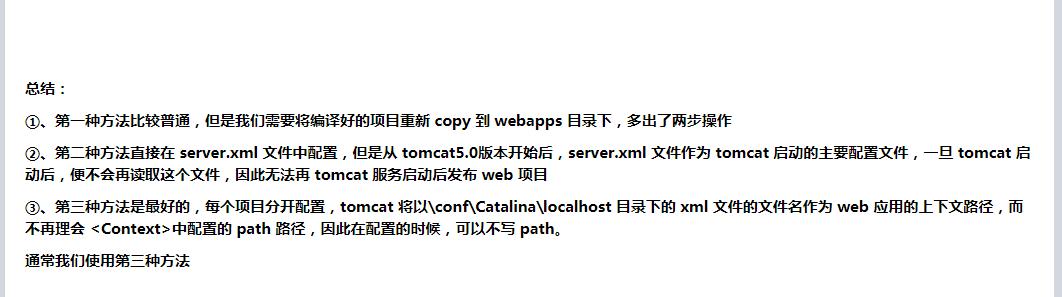
摘录：<https://www.cnblogs.com/ysocean/p/6893446.html>

其实就是配置一个扫描路径









## 12 Tomcat部署静态资源



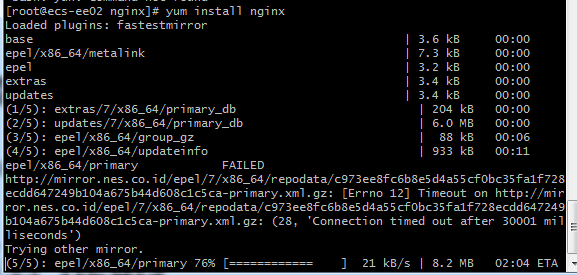
# 三 部署Nginx

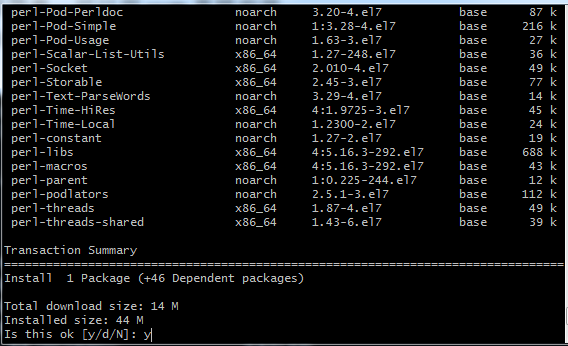
## 1 安装

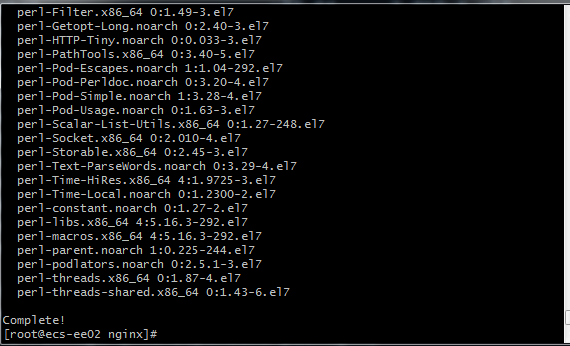
Nginx安装简单，可以用yum直接安装

yum install nginx

中间需要输入y 确认 回车



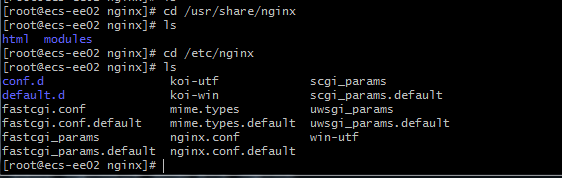




## 2 查看安装目录

whereis nginx

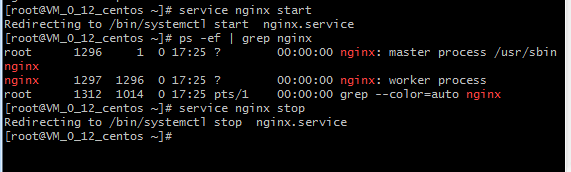




## 3 启动

1 服务启动（yum安装完后自动安装了服务）

service nginx start



2 配置启动

nginx -tc /etc/nginx/nginx.conf（直接启动）  
nginx -s reload -c /etc/nginx/nginx.conf （校验配置是否正确启动）

## 4 配置

vi nginx.conf 编辑配置解析

### 1 静态资源解析

注意：防火墙端口一定要开放

location /jiyin-admin/ {

root /mnt/disk-data/jiyin/www;

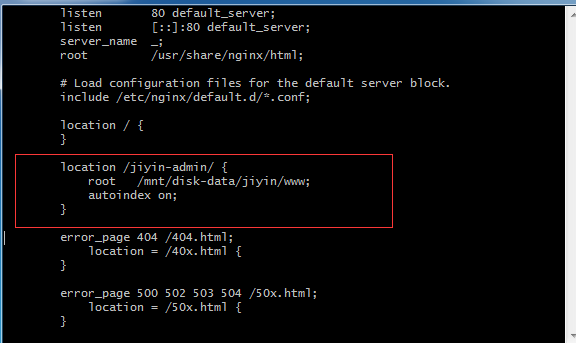
autoindex on;

}

访问路径：<http://119.3.19.208/jiyin-admin/index.html>

root：指向资源目录

jiyin-admin: 1 访问路径的根目录，2 资源目录的下一级（目录）



### 2 tomcat服务器跳转配置

Nginx 收到请求，会跳转访问本地Tomcat服务器器

location / {

proxy\_set\_header Host $host:$server\_port;

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

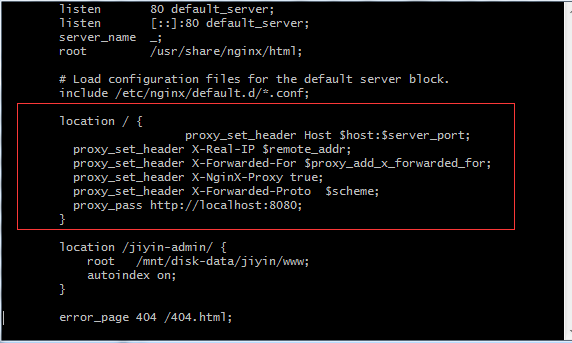
proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

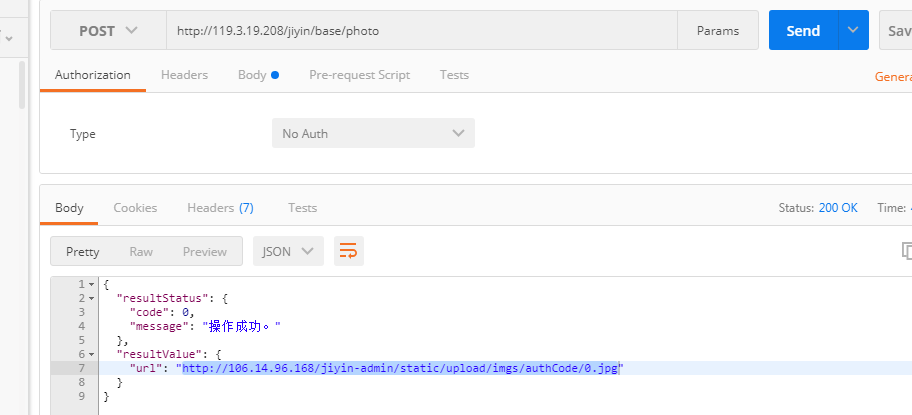
proxy\_set\_header X-NginX-Proxy true;

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

proxy\_pass http://localhost:8080;

}





### 3 ssl证书配置

配置完后，用域名就可以访问本服务器了

1 配置域名解析，就是在Nginx配置配置一个新server 服务

如图，第一个服务监听的是80端口，也就是正常的nginx访问

第二个服务监听的是443 ssl http2,等等这样域名的路径调用，就能被捕捉到了

2 DNS 域名解析，NDS服务器只负责IP和域名的对应，解析完成后，可以

直接通过域名访问服务器

3 SSL 证书：可以解析域名，并对传输进行加密，验证，ssl解析配置完成后

就可以通用https访问服务器，

4 ssl和DNS的联系：ssl之所以可以解析域名，是要通过dns服务器的认证，

在解析域名的时候，要先申请ssl证书，然后填写ssl证书的记录值，这样

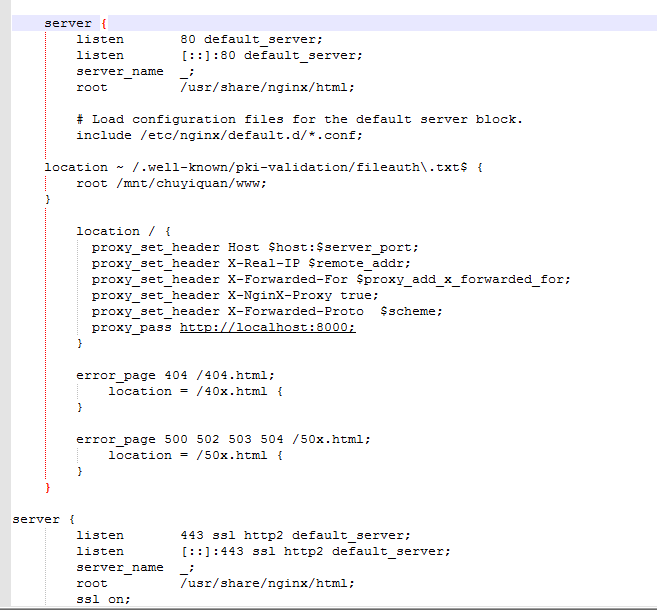
DNS在认证通过后，ssl就可以收到响应，并且生成最终的ssl证书了，

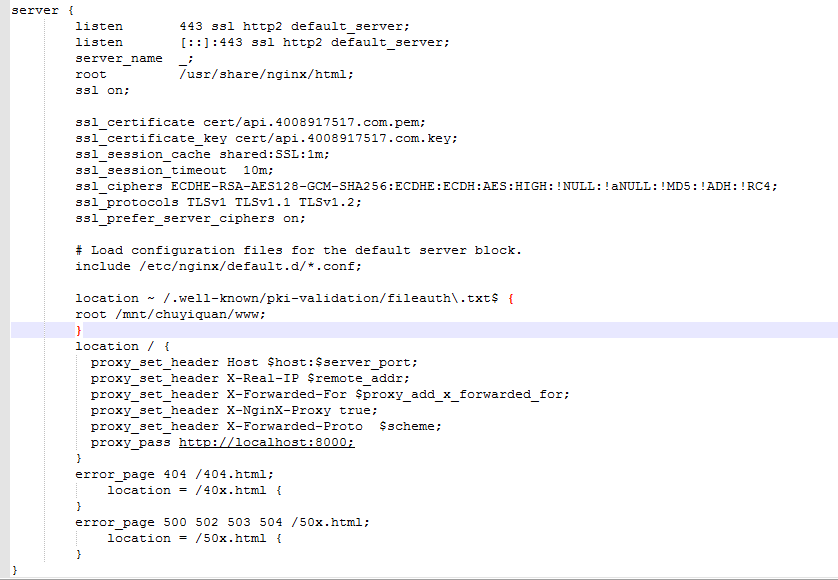
下载生成好的ssl证书，配置到服务器上，解析https请求，就可以通用https访问服务器了

摘录网址：

https://cloud.tencent.com/document/product/400/4142#1.-.E6.89.8B.E5.8A.A8dns.E9.AA.8C.E8.AF.81







server {

listen 443 ssl http2 default\_server;

listen [::]:443 ssl http2 default\_server;

server\_name \_;

root /usr/share/nginx/html;

ssl on;

ssl\_certificate cert/api.4008917517.com.pem;

ssl\_certificate\_key cert/api.4008917517.com.key;

ssl\_session\_cache shared:SSL:1m;

ssl\_session\_timeout 10m;

ssl\_ciphers ECDHE-RSA-AES128-GCM-SHA256:ECDHE:ECDH:AES:HIGH:!NULL:!aNULL:!MD5:!ADH:!RC4;

ssl\_protocols TLSv1 TLSv1.1 TLSv1.2;

ssl\_prefer\_server\_ciphers on;

# Load configuration files for the default server block.

include /etc/nginx/default.d/\*.conf;

location ~ /.well-known/pki-validation/fileauth\.txt$ {

root /mnt/chuyiquan/www;

}

location / {

proxy\_set\_header Host $host:$server\_port;

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

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

proxy\_set\_header X-NginX-Proxy true;

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

proxy\_pass http://localhost:8000;

}

error\_page 404 /404.html;

location = /40x.html {

}

error\_page 500 502 503 504 /50x.html;

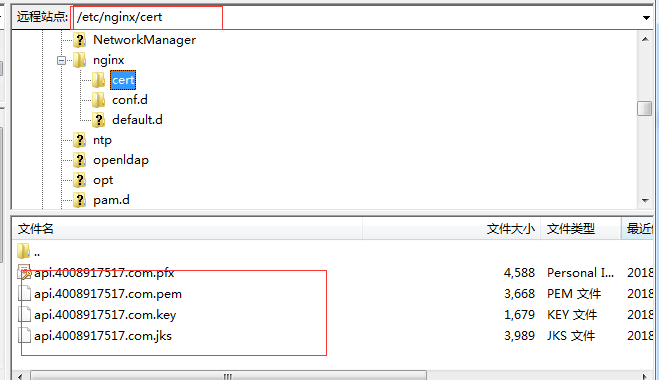
location = /50x.html {

}

}

2 证书存放路径

这个好像是nginx默认的配置路径下级，它就可以找到

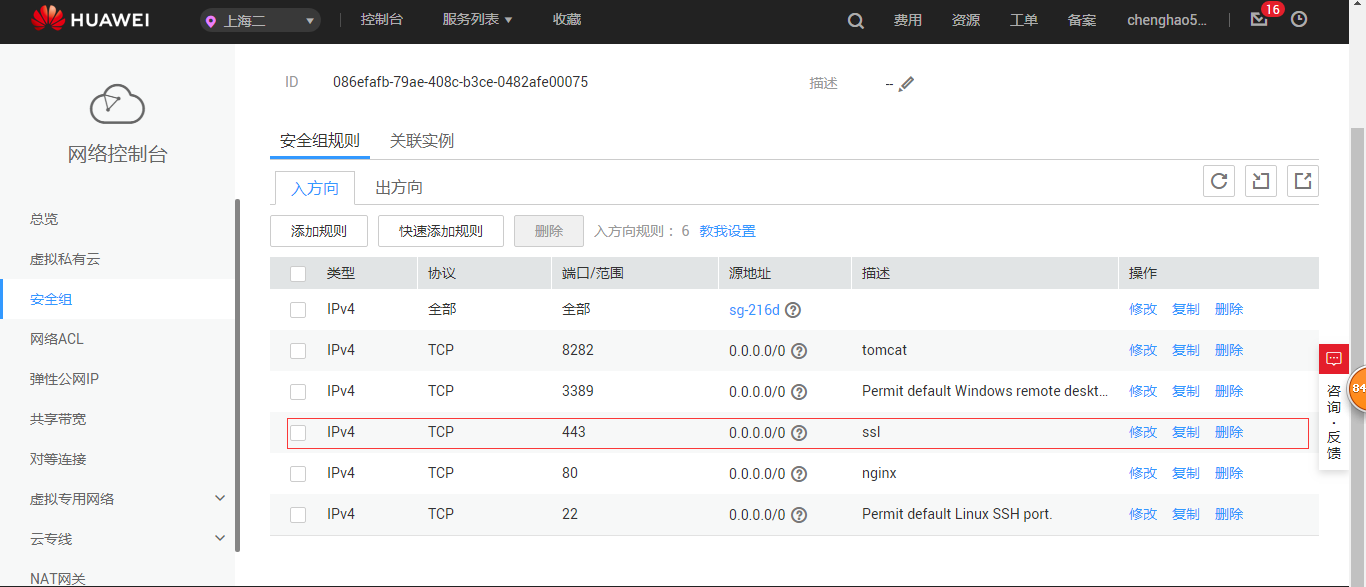


### 4 服务器安全组配置

华为云上这边443端口没有放行，ssl请求一直不通，配置改了半天，坑死我了

如果服务器防火墙开了，那么服务器也要开放433端口

不然nginx监听的433端口是访问不了的

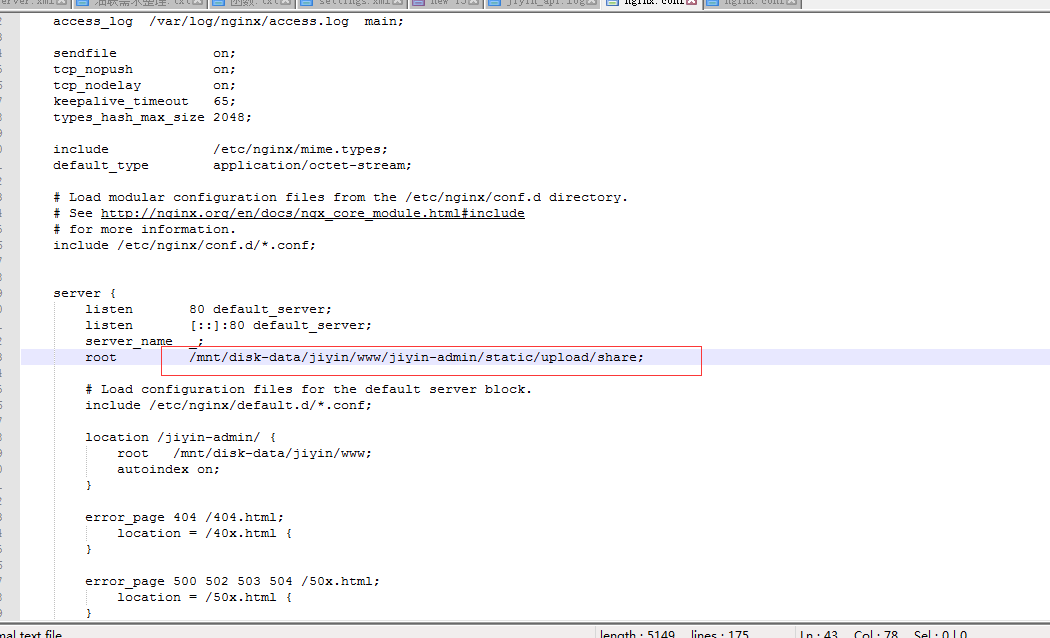


### 5 摘录

<http://www.xinnet.com/xinzhi/65/130548.html>



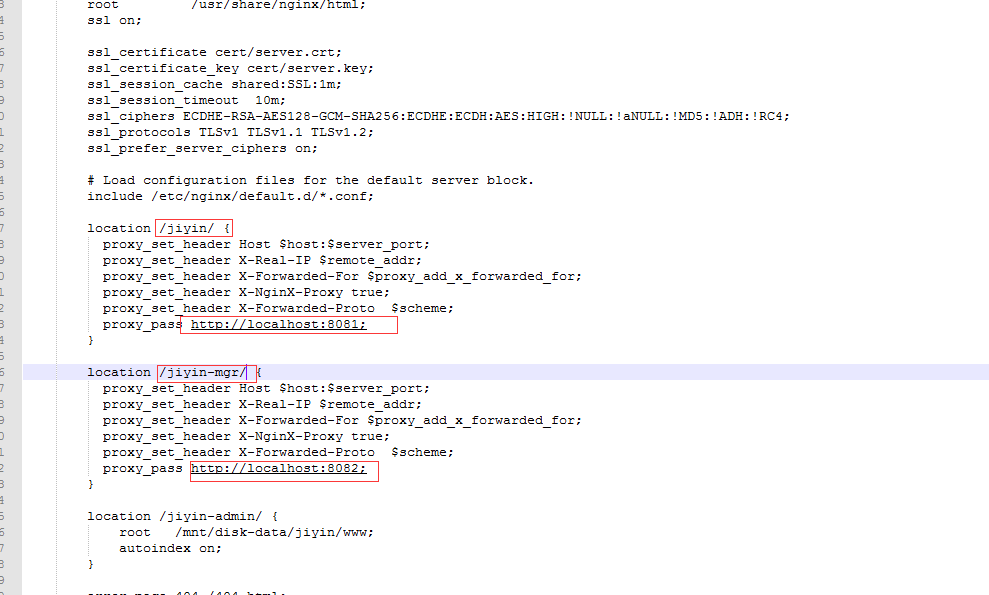
### 6 跳转官方主页



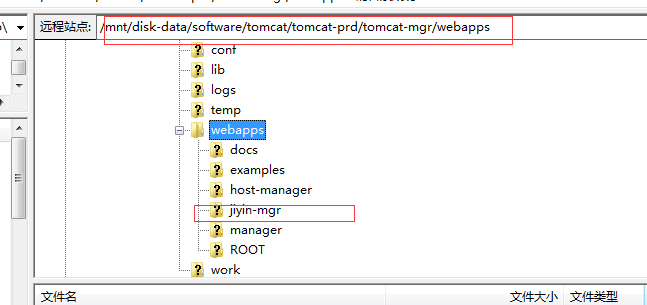
## 5 配置多个服务

### 1 多个Tomcat同时运行，用相同的域名

解析同一个域名，可以配置多个不同的IP跳转







### 2 多个Tomcat同时运行，并且用不同的域名

配置2个server解析不同HTTPS，不能加上2个默认服务器



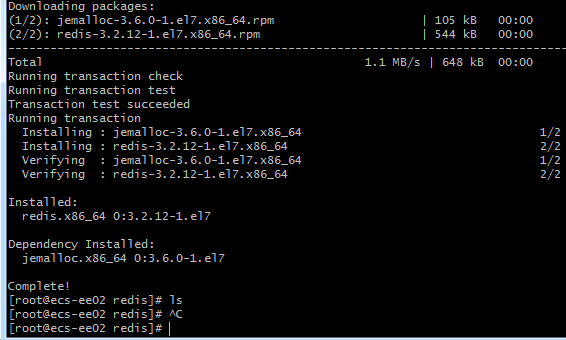


# 四 部署Redis

## 1 安装

这里用的是yum直接安装，直接装

1 yum install redis



## 2 查看Redis状态

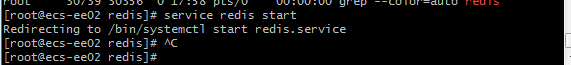
ps -ef | grep redis



## 3 启动Redis

service redis start （service 启动，yum 安装时会安装service服务）

systemctl restart redis (另一种启动方式)



## 4 网上教程

### ****1、yum install redis      --查看是否有redis   yum 源****

**2、yum install epel-release    --下载fedora的epel仓库**

**3、 yum install redis    -- 安装redis数据库**

**4、service redis start  Redirecting to /bin/systemctl start redis.service   --开启redis服务**

**redis-server /etc/redis.conf   --开启方式二**

**5、ps -ef | grep redis   -- 查看redis是否开启**

**6、redis-cli       -- 进入redis服务**

**7、redis-cli  shutdown      --关闭服务**

**8、开放端口6379、6380的防火墙**

**/sbin/iptables -I INPUT -p tcp --dport 6379  -j ACCEPT   开启6379**

**/sbin/iptables -I INPUT -p tcp --dport 6380 -j ACCEPT  开启6380**

**/etc/rc.d/init.d/iptables save                           保存**

**9、使用redis  desktop manager连接redis**

# 五 防火墙配置

## 1 查看防火墙状态

查看防火墙状态 systemctl status firewalld

开启防火墙 systemctl start firewalld

关闭防火墙 systemctl stop firewalld

开启防火墙 service firewalld start

若遇到无法开启

先用：systemctl unmask firewalld.service

然后：systemctl start firewalld.service

**2 查看**对外开放的端口状态 查询已开放的端口

netstat -anp 查询指定端口是否已开

firewall-cmd --query-port=666/tcp 提示 yes，表示开启；no表示未开启。

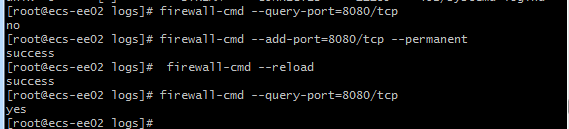


## 3添加端口

添加指定需要开放的端口：firewall-cmd --add-port=123/tcp --permanent

重载入添加的端口： firewall-cmd --reload

查询指定端口是否开启成功： firewall-cmd --query-port=123/tcp



# 六 MYSQL 操作

## 1 安装mysql-cli

安装mysql客户端，这样才能使用mysql命令

## 2 连接

命令 ：mysql -h 192.168.1.226 -u prd-jiyin --default-character-set=utf8 -p -P 6063

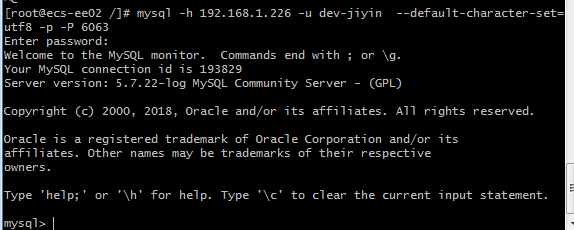
-h:IP

-u:用户名

--default-character-set=utf8:连接数据库编码格式（解决乱码）

-p:密码（小写的p）

-P：端口号(大写的P,不是默认端口必须加)



## 3 操作

**所有的操作命令要注分号“；”结尾**

show databases; 显示数据库

create database name; 创建数据库

source /mnt/disk-data/jiyin/DB/jiyin.sql; 导入SQL（文件）

show tables; 显示所有表

# 七 安装apr 和 Tomcat native

## 1 APR

echo "install apr-1.6.3 ......"

cd /usr/local/src

wget http://mirrors.shuosc.org/apache//apr/apr-1.6.3.tar.gz

tar -xzvf apr-1.6.3.tar.gz

cd apr-1.6.3

./configure --prefix=/usr/local/apr

make

make install

echo "install expat-devel for apr-util...."

yum -y install expat-devel

echo "install apr-util-1.6.1 ......"

cd /usr/local/src

wget http://mirrors.shuosc.org/apache/apr/apr-util-1.6.1.tar.gz

tar -xzvf apr-util-1.6.1.tar.gz

cd apr-util-1.6.1

./configure --with-apr=/usr/local/apr --with-jdk=/usr/java/jdk-1.8.0\_151

make

make install

## 2 Tomcat native

echo "install tomcat-native..."

echo $1

`cd $1/bin`

tar -xzvf tomcat-native.tar.gz

cd tomcat-native-\*

cd native

./configure --with-apr=/usr/local/apr --with-ssl=/usr/local/openssl

make

make install

# 八 云盘挂载

## 1 步骤

1 购买磁盘，云服务器管理，先进行磁盘挂载

2 服务器管理平台挂载好后，在服务器上进行磁盘分区，格式化

Fidsk -l 查看磁盘，disk磁盘下，如果没有device boot 分区，说明这个磁盘可以被挂载

这个时候就是对这个disk磁盘进行分区，

3 这里只分一个区所有都是默认

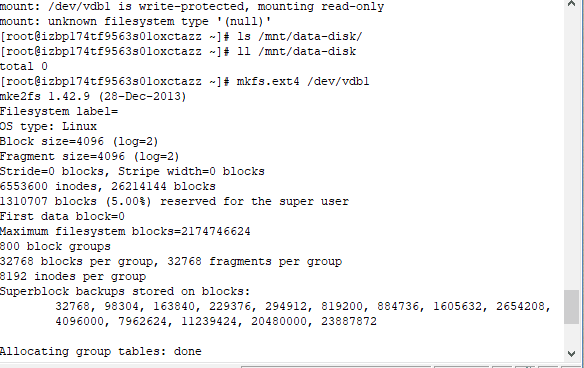
4 分好区后进行格式化

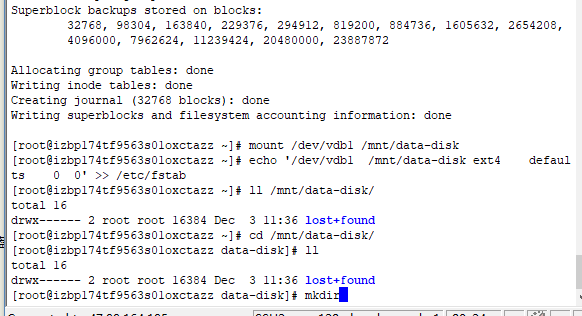
5 格式化好后进行挂载，其实就是路径映射

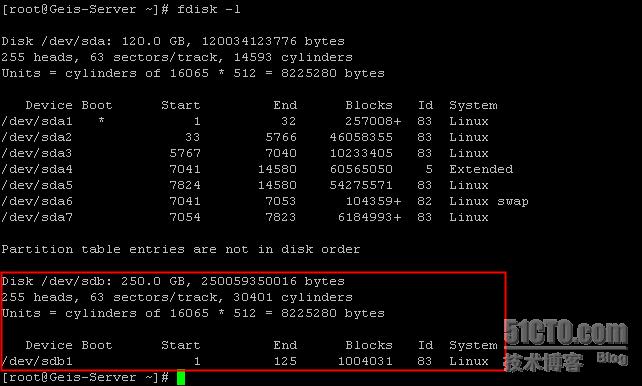
6.配置开机自动挂载

修改/etc/fstab文件，文件末尾添加：

/dev/vdb   /opt ext4    defaults    0  0

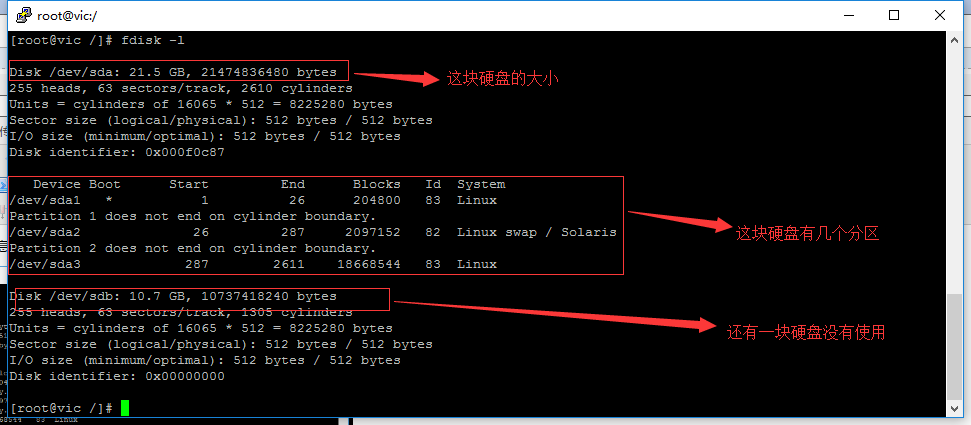


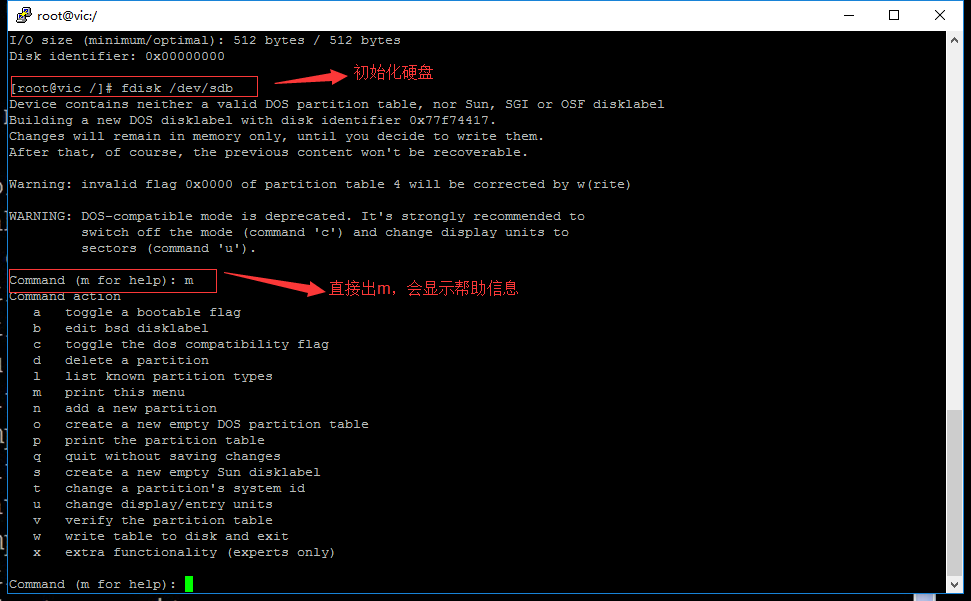




## 2 摘录

https://www.cnblogs.com/myvic/p/6816924.html





1 分析：各个参数的解析

　                   1. 输入 m 显示所有命令列示。

　　　　　　　　2. 输入 p 显示硬盘分割情形，打印分区表。

　　　　　　　　3. 输入 a 设定硬盘启动区。

　　　　　　　　4. 输入 n 设定新的硬盘分割区。

　　　　　　　　　4.1. 输入 e 硬盘为[延伸]分割区(extend)。

　　　　　　　　　4.2. 输入 p 硬盘为[主要]分割区(primary)。

　　　　　　　　5. 输入 t 改变硬盘分割区属性。

　　　　　　　　　　t:分区系统id号

　　　　　　　　　　　　L:82:linux swap

　　　　　　　　　　　　83:linux

　　　　　　　　　　　　86：NTFS window分区

　　　　　　　　6. 输入 d 删除硬盘分割区属性。

　　　　　　　　7. 输入 q 结束不存入硬盘分割区属性。

　　　　　　　　8. 输入 w 结束并写入硬盘分割区属性

