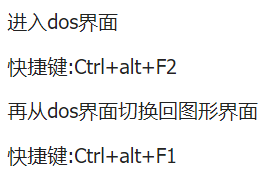
Cp 复制文件

Cp -r 复制目录

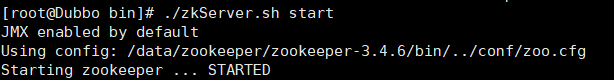
rm -rf 删除文件夹

mv 文件名 修改后的文件名 重命名文件



## 软件

ZK 



Dubbo-admin 

Redis 160 

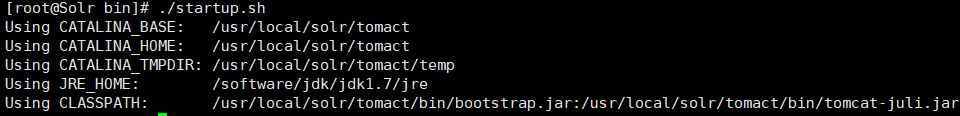
./redis-server redis.conf

./redis-cli shutdown

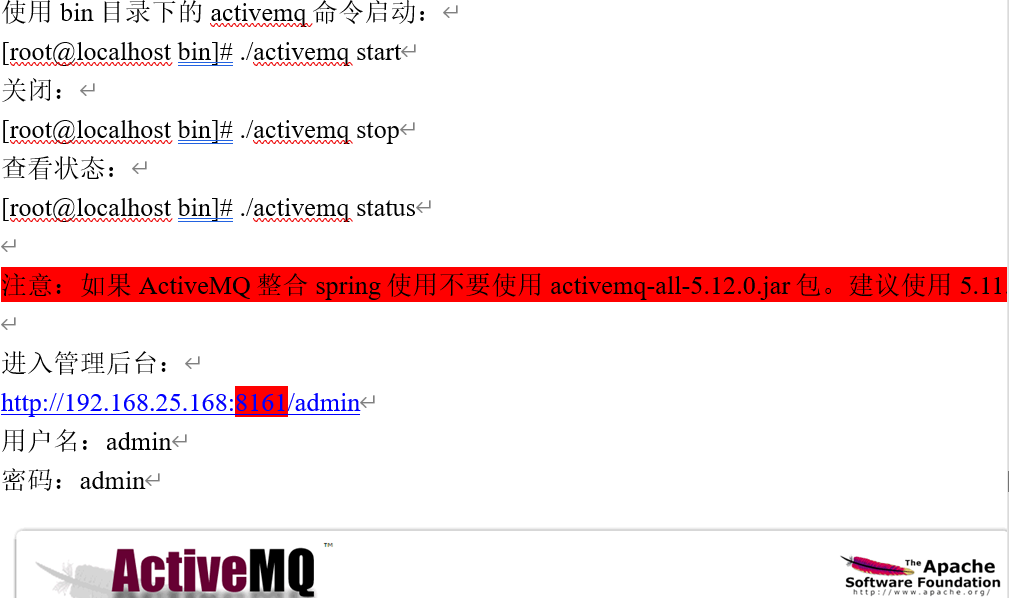
图片访问161，两个nginx

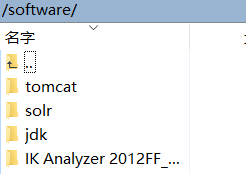


Solr /usr/local/solr/tomact/bin



Mq在162 





## ZK



要先启zk，再起应用。

/data/zookeeper/zookeeper-3.4.6/bin



Tomcat dubbo-admin：/data/tomcat/app-dubbo-admin/bin

访问http://192.168.181.160:8080/dubbo-admin/



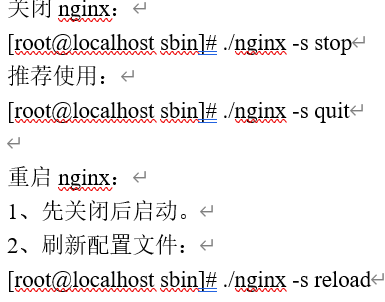
访问：http://192.168.181.160:8080/dubbo-admin/

## Nginx

位置







**Nginx配置反向代理：**

**场景：请求量激增，需要服务器扩容，并添加反向代理服务器。**

**此时假设有两个应用在不同的服务器对外提供服务（域名为sina和sohu），需要nginx做请求代理转发。**





**需要配置server节点中location子节点的proxy\_pass属性**

**需要配置upstream节点指向的应用服务器的地址。**

**注意：**

**这两个应用目前是在同一个机器上，所以要用不同的端口进行区分。一般是不同的ip。**

**Nginx的端口是80，tomcat的是8080**





1. **所有请求到nginx服务器。**
2. **Nginx做负载均衡，并将请求转发给不同的应用服务器**
3. **应用服务器承接请求并相应，结果返回给nginx，nginx再将结果相应给用户。**

测试：

在dubbo上启动两个应用





## fastDFS

需要启动nginx，能够使用连接访问图片访问图片

**打开n2 /usr/local/nginx2**

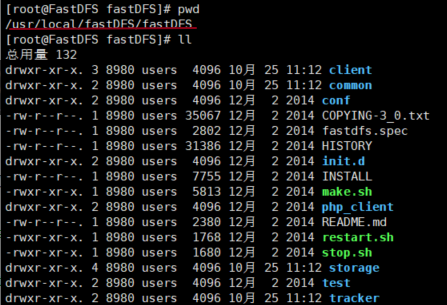
**图片 /opt/fastdfs\_storage\_data/data/00/00**

nginx 9999

nginx2 80

轻量级的分布式文件系统，由跟踪服务器Tracker server 存储服务器 storage ser 和client组成。主要解决海量数据存储问题，适合中小文件 大于4KB 小于500MB为载体的在线服务。

FastDFS的安装目录：



Fastdfs安装成功：



**fdfs放三部分的配置文件。 在etc中**

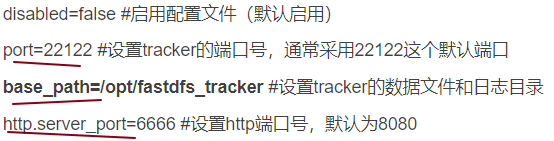
### Tracker

创建Tracker服务器的文件路径，用于存储Tracker的数据文件和日志文件。

/opt/fastdfs\_tracker

**在opt中**

配置/etc/fdfs目录下的tracker.conf配置文件



**启动命令：**

**service fdfs\_trackerd start**

启动命令执行成功，那么同时在刚才创建的tracker文件目录**/opt/fastdfs\_tracker**中就可以看到启动后新生成的data和logs目录。

**netstat命令查看一下端口监听情况：**

**netstat -unltp|grep fdfs**

将tracker设置为**开机启动**，打开**/etc/rc.d/rc.local**并在其中加入以下配置：

service fdfs\_trackerd start

检查一下rc.local是否具备可执行权限，若是无可执行权限则通过进行授权:

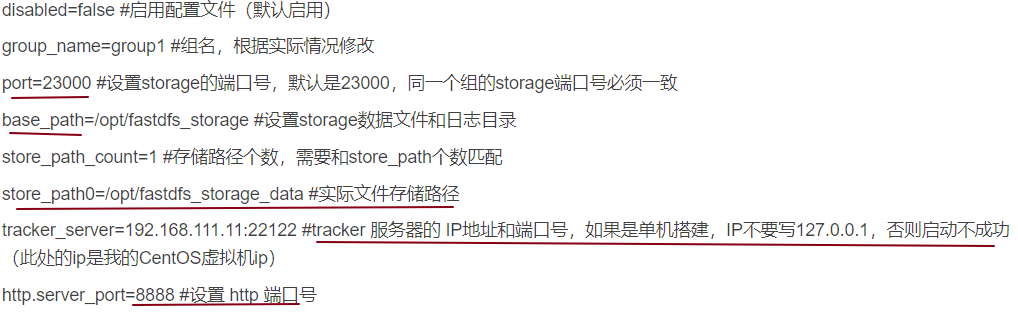
chmod +x /etc/rc.d/rc.local

### Storage

Storage服务器的文件目录



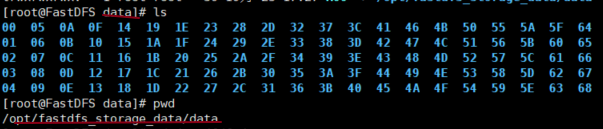
接下来修改/etc/fdfs目录下的storage.conf配置文件。



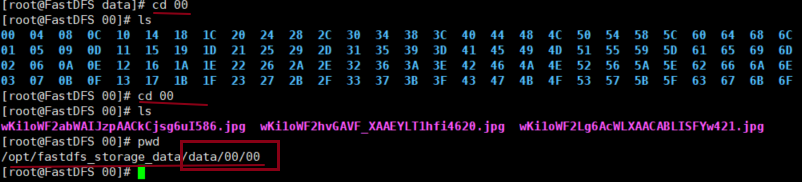
**启动Storage服务：**

**service fdfs\_storaged start**

如果启动成功，/opt/fastdfs\_storage中就可以看到启动后新生成的data和logs目录，端口23000也应被正常监听，还有一点就是**文件存储路径下会生成多级存储目录**。



data下有256个1级目录，每级目录下又有256个2级子目录，总共65536个文件，新写的文件会以hash的方式被路由到其中某个子目录下，然后将文件数据直接作为一个本地文件存储到该目录中。



确定了storage服务器启动成功后，还有一项工作就是看看storage服务器是否已经登记到 tracker服务器（也可以理解为tracker与storage是否整合成功），运行以下命令：

**/usr/bin/fdfs\_monitor /etc/fdfs/storage.conf**

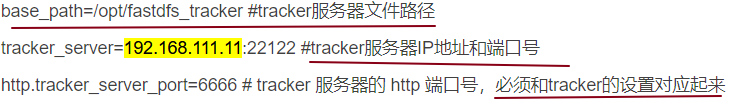


添加**开机启动**，打开**/etc/rc.d/rc.local**并将如下配置**追加**到文件中：

service fdfs\_storaged start

### client

前两部分安装完成，用client测试。编辑/etc/fdfs目录下的**client.conf 文件**：



配置完模拟上传图片：

**/usr/bin/fdfs\_upload\_file** **/etc/fdfs/client.conf** /opt/BLIZZARD**.jpg**

成功后返回路径：

这里写图片描述



**用浏览器发送HTTP请求访问一下文件**：



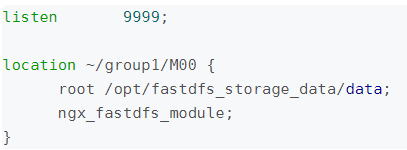
看自己的端口配置。

### 整合nginx

**要先开n1，才能访问n2**

**Storage的nginx**

修改nginx的配置文件：不用原来的80



FastDFS的安装目录下的conf目录，将http.conf和mime.types拷贝到/etc/fdfs目录下：

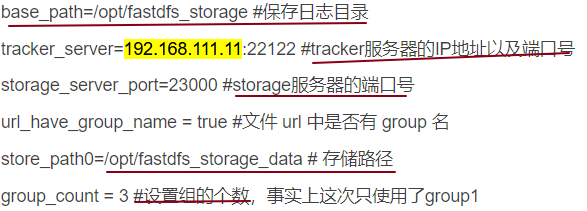
**cp -r** /usr/local/fastdfs-5.05/conf/http.conf /etc/fdfs/

cp -r /usr/local/fastdfs-5.05/conf/mime.types /etc/fdfs/

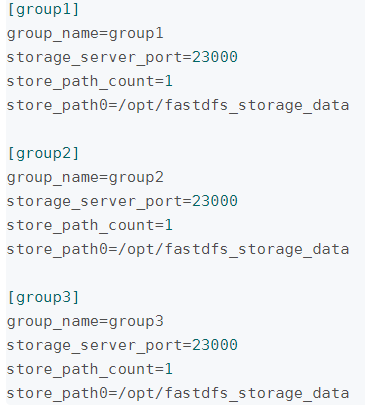
把fastdfs-nginx-module中 src目录下的mod\_fastdfs.conf拷贝到/etc/fdfs目录下：

**cp -r** /usr/local/fastdfs-nginx-module-master/src/mod\_fastdfs.conf /etc/fdfs/

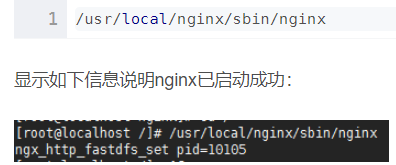
编辑mod\_fastdfs.conf：



设置了group\_count = 3，接下来就需要**在文件尾部追加这3个group setting**：



启动nginx：



storage服务器的nginx就已经安装完毕，接下来看一下tracker服务器的nginx安装。

最后要把nginx2开开，访问上传好的图片

怎么都传到了

？不能随机上传吗？

系统安装



sed -i "s|enabled=1|enabled=0|g" /etc/yum/pluginconf.d/fastestmirror.conf

mv /etc/yum.repos.d/CentOS-Base.repo /etc/yum.repos.d/CentOS-Base.repo.backup

curl -o /etc/yum.repos.d/CentOS-Base.repo https://www.xmpan.com/Centos-6-Vault-Aliyun.repo

yum clean all

yum makecache