



链滴

[环境] 搭建 FastDFS 最小文件系统（一台机）

作者: [lin772662623](#)

原文链接: <https://ld246.com/article/1530185414500>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



文章写好有段时间了，如果有任何问题，欢迎留言，我会更正。

☺smile

1.什么是FastDFS

FastDFS是用c语言编写的一款开源的分布式文件系统。FastDFS为互联网量身定制，充分考虑了冗余备份、负载均衡、线性扩容等机制，并注重高可用、高性能等指标，使用FastDFS很容易搭建一套性能的文件服务器集群提供文件上传、下载等服务。

2.FastDFS架构

FastDFS架构包括 Tracker server和Storage server。客户端请求Tracker server进行文件上传、下载，通过Trackerserver调度最终由Storage server完成文件上传和下载。

Trackerserver作用是负载均衡和调度，通过Trackerserver在文件上传时可以根据一些策略找到Storage server提供文件上传服务。可以将tracker称为追踪服务器或调度服务器。

Storage server作用是文件存储，客户端上传的文件最终存储在Storage服务器上，Storage server没有实现自己的文件系统而是利用操作系统的文件系统来管理文件。可以将storage称为存储服务器。

2.1 Tracker 集群

FastDFS集群中的Tracker server可以有多台，Tracker server之间是相互平等关系同时提供服务Tracker server不存在单点故障。客户端请求Tracker server采用轮询方式，如果请求的tracker无法提供服务则换另一个tracker。

2.2 Storage集群

Storage集群采用了分组存储方式。storage集群由一个或多个组构成，集群存储总容量为集群中有组的存储容量之和。一个组由一台或多台存储服务器组成，组内的Storage server之间是平等关系，不同组的Storage server之间不会相互通信，同组内的Storage server之间会相互连接进行文件同步，而保证同组内每个storage上的文件完全一致的。一个组的存储容量为该组内存储服务器容量最小的一个，由此可见组内存储服务器的软硬件配置最好是一致的。

采用分组存储方式的好处是灵活、可控性较强。比如上传文件时，可以由客户端直接指定上传到组也可以由tracker进行调度选择。一个分组的存储服务器访问压力较大时，可以在该组增加存储服务来扩充服务能力（纵向扩容）。当系统容量不足时，可以增加组来扩充存储容量（横向扩容）。

2.3 Storage状态收集

Storage server会连接集群中所有的Tracker server，定时向他们报告自己的状态，包括磁盘剩余空间、文件同步状况、文件上传下载次数等统计信息。

3 开始搭建

3.1 FastDFS-tracker 安装

3.1.1 下载

tracker和storage使用相同的安装包，<https://github.com/happyfish100/FastDFS>。

下面用到与fastdfs相关的包，都可以在<https://github.com/happyfish100>找到

3.1.2 gcc

FastDFS是C语言开发，建议在linux上运行，本教程使用Centos6.4作为安装环境。

安装FastDFS需要编译，编译依赖于gcc环境，如果没有gcc环境，需要安装gcc:

```
sudo yum install gcc-c++
```

3.1.3 libevent

FastDFS依赖libevent库，需要安装:

```
sudo yum -y install libevent
```

3.1.4 libfastcommon

libfastcommon是FastDFS官方提供的，libfastcommon包含了FastDFS运行所需要的一些基础。

将libfastcommonV1.0.7.tar.gz 解压，执行以下命令

```
tar -zxvf libfastcommonV1.0.7.tar.gz
```

```
cd libfastcommon-1.0.7
```

```
./make.sh
```

```
./make.sh install /usr/local
```

注意：如果后面遇到问题，可以试下这里，如果没问题就不需要理会。

libfastcommon.so安装好后会自动将库文件拷贝至/usr/lib64下，由于FastDFS程序引用usr/lib目所以需要将/usr/lib64下的库文件拷贝至/usr/lib下。

3.1.5 tracker编译安装

```
tar -zxvf FastDFS_v5.05.tar.gz
```

```
cd FastDFS
```

```
./make.sh
```

```
./make.sh install /usr/local
```

安装成功将安装目录下的conf下的文件拷贝到/etc/fdfs/下。

```
sudo cp conf/* /etc/fdfs/
```

3.1.6 配置

安装成功后进入/etc/fdfs目录：

```
vi tracker.conf
```

修改其中内容：

```
base_path = /home/yuqing/FastDFS
```

改为：

```
base_path=/home/FastDFS
```

这里需要注意，没有 /home/FastDFS这个目录需要创建

```
sudo mkdir -p /home/FastDFS
```

3.1.7 启动

执行以下命令启动

```
fdfs_trackerd /etc/fdfs/tracker.conf restart
```

第一次启动并没有任何提示，可以进入/home/FastDFS/logs下查看日志

设置开机自动启动：

```
sudo vi /etc/rc.d/rc.local
```

添加以下内容：

```
fdfs_trackerd /etc/fdfs/tracker.conf restart
```

3.2 FastDFS-storage 安装

在同一台机子上继续装storage，编译方法同tracker，

3.2.1 配置

下面讲下配置

```
sudo vi /etc/fdfs/storage.conf
```

修改以下内容：

```
group_name=group1
```

```
base_path=/home/youqing/FastDFS 改为: base_path=/home/FastDFS
```

```
store_path0=/home/youqing/FastDFS 改为: store_path0=/home/FastDFS/fdfs_storage  
(这里需要注意，如果fdfs_storage没有创建，需要手工创建)
```

```
tracker_server=192.168.36.129:22122 #配置tracker服务器ip
```

3.2.2 启动

```
fdfs_storaged /etc/fdfs/storage.conf restart
```

设置开机自动启动

```
sudo vi /etc/rc.d/rc.local
```

添加以下内容：

```
fdfs_storaged /etc/fdfs/storage.conf restart
```

3.3 上传图片测试

```
sudo vi /etc/fdfs/client.conf
```

修改以下内容：

```
base_path=/home/FastDFS
```

```
tracker_server=192.168.36.129:22122
```

修改完后，使用以下命令上传图片

```
fdfs_test /etc/fdfs/client.conf upload /home/llq/Pictures/loading.gif
```

执行后，会打印出以下内容：

```
http://192.168.36.129/group1/M00/00/00/wKgkgVqzrQiAal71AAAqm9RPKXs429.gif
```

使用浏览器访问此url，能正常显示图片则为搭建成功。

3.4 FastDFS和Nginx整合

由于我们只有一台服务器，所以就在storage上安装nginx。

3.4.1 FastDFS-nginx-module

下载好包后，执行以下命令：

```
unzip fastdfs-nginx-module-master.zip
```

```
cd fastdfs-nginx-module-master/src
```

将FastDFS-nginx-module/src下的mod_fastdfs.conf拷贝至/etc/fdfs/下

```
cp mod_fastdfs.conf /etc/fdfs/
```

修改conf内容：

```
vi /etc/fdfs/mod_fastdfs.conf
```

修改以下内容：

```
base_path=/home/FastDFS
```

```
tracker_server=192.168.36.129:22122
```

```
url_have_group_name = true
```

```
store_path0=/home/FastDFS/fdfs_storage
```

将libfdfsclient.so拷贝至/usr/lib下

```
cp /usr/lib64/libfdfsclient.so /usr/lib/
```

创建nginx/client目录

```
sudo mkdir -p /var/temp/nginx/client
```

重新配置nginx，添加模块，然后编译

```
./configure --prefix=/usr/local/nginx --add-module=/home/llq/developer/fastdfs-nginx-module-master/src
```

```
make
```

```
make install
```

添加nginx相应的conf文件

```
cd /usr/local/nginx/conf
```

```
vi nginx.conf
```

添加以下内容：

```
include vhost/*.conf;
```

进入vhost，添加配置文件

```
cd vhost
```

```
vi nginx-fdfs.conf
```

```
server {  
    listen 80;  
    server_name 192.168.36.129;  
    location /group1/M00/ {  
        ngx_fastdfs_module;  
    }  
}
```

添加好之后，重启nginx

```
sudo /usr/local/nginx/sbin/nginx -s reload
```

出现以下字样则成功：

```
ngx_http_fastdfs_set pid=13534
```