# 单机FastDFS安装部署

## 1、下载安装 libfastcommon

libfastcommon是从 FastDFS 和 FastDHT 中提取出来的公共 C 函数库，基础环境，安装即可 。

① 下载libfastcommon

# wget https://github.com/happyfish100/libfastcommon/archive/V1.0.7.tar.gz

② 解压

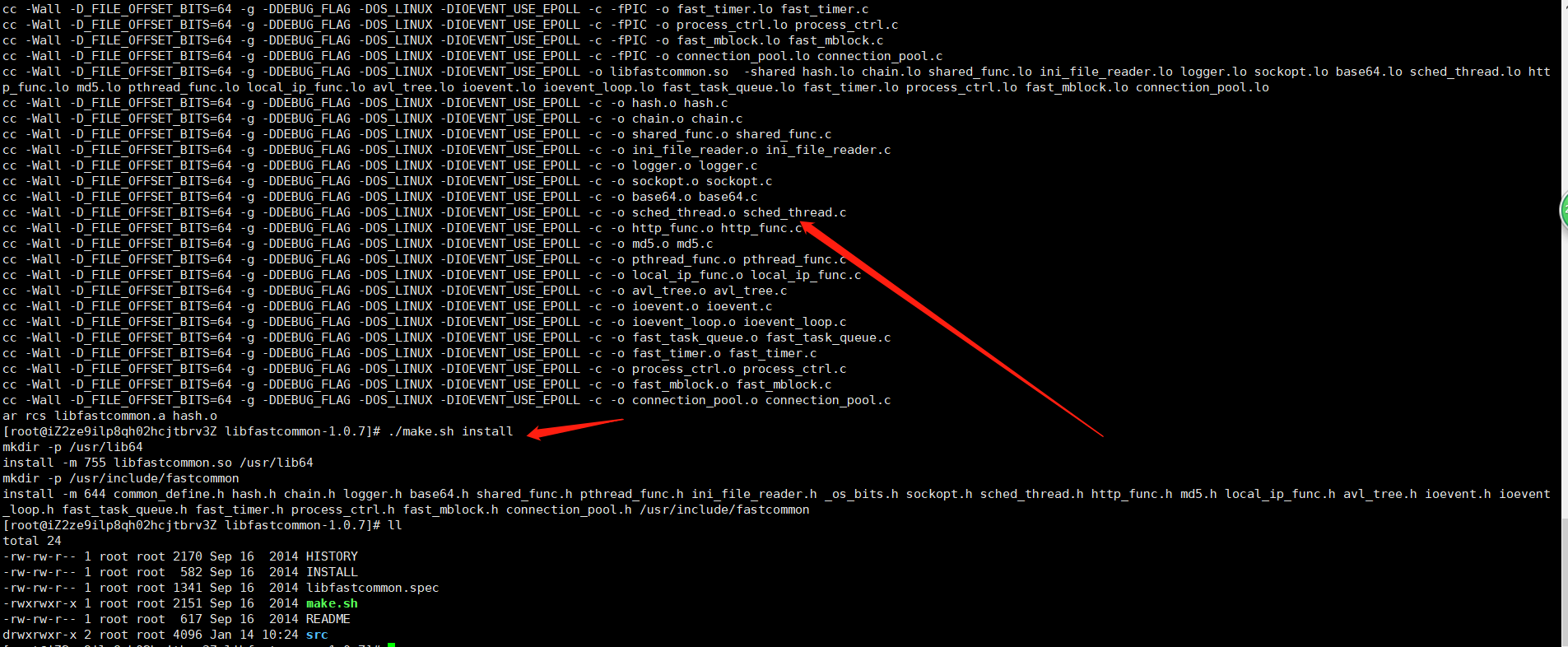
# tar -zxvf V1.0.7.tar.gz

# cd libfastcommon-1.0.7

③ 编译、安装

# ./make.sh

# ./make.sh install



④ libfastcommon.so 安装到了/usr/lib64/libfastcommon.so，但是FastDFS主程序设置的lib目录是/usr/local/lib，所以需要创建软链接。

# ln -s /usr/lib64/libfastcommon.so /usr/local/lib/libfastcommon.so

# ln -s /usr/lib64/libfastcommon.so /usr/lib/libfastcommon.so

# ln -s /usr/lib64/libfdfsclient.so /usr/local/lib/libfdfsclient.so

# ln -s /usr/lib64/libfdfsclient.so /usr/lib/libfdfsclient.so

## 2、下载安装FastDFS

① 下载FastDFS

# wget https://github.com/happyfish100/fastdfs/archive/V5.05.tar.gz

② 解压

# tar -zxvf V5.05.tar.gz

# cd fastdfs-5.05

③ 编译、安装

# ./make.sh

# ./make.sh install

④ 默认安装方式安装后的相应文件与目录  
　　A、服务脚本：

/etc/init.d/fdfs\_storaged

/etc/init.d/fdfs\_tracker

　　B、配置文件（这三个是作者给的样例配置文件） :

/etc/fdfs/client.conf.sample

/etc/fdfs/storage.conf.sample

/etc/fdfs/tracker.conf.sample

　　C、命令工具在 /usr/bin/ 目录下：

fdfs\_appender\_test

fdfs\_appender\_test1

fdfs\_append\_file

fdfs\_crc32

fdfs\_delete\_file

fdfs\_download\_file

fdfs\_file\_info

fdfs\_monitor

fdfs\_storaged

fdfs\_test

fdfs\_test1

fdfs\_trackerd

fdfs\_upload\_appender

fdfs\_upload\_file

stop.sh

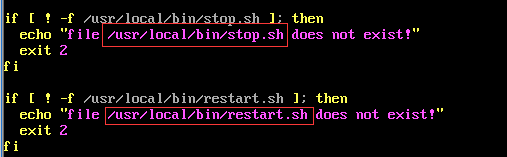
restart.sh

⑤ FastDFS 服务脚本设置的 bin 目录是 /usr/local/bin， 但实际命令安装在 /usr/bin/ 下。

　　两种方式：

　　》 一是修改FastDFS 服务脚本中相应的命令路径，也就是把 /etc/init.d/fdfs\_storaged 和 /etc/init.d/fdfs\_tracker 两个脚本中的 /usr/local/bin 修改成 /usr/bin。

 　　　　# vim fdfs\_trackerd  
　　　　使用查找替换命令进统一修改:%s+/usr/local/bin+/usr/bin  
　　　　# vim fdfs\_storaged  
　　　　使用查找替换命令进统一修改:%s+/usr/local/bin+/usr/bin



　　》 二是建立 /usr/bin 到 /usr/local/bin 的软链接，我是用这种方式。

# ln -s /usr/bin/fdfs\_trackerd /usr/local/bin

# ln -s /usr/bin/fdfs\_storaged /usr/local/bin

# ln -s /usr/bin/stop.sh /usr/local/bin

# ln -s /usr/bin/restart.sh /usr/local/bin

## 3、配置FastDFS跟踪器(Tracker)

配置文件详细说明参考：[FastDFS 配置文件详解](http://bbs.chinaunix.net/forum.php?mod=viewthread&tid=1941456&extra=page%3D1%26filter%3Ddigest%26digest%3D1)

① 进入 /etc/fdfs，复制 FastDFS 跟踪器样例配置文件 tracker.conf.sample，并重命名为 tracker.conf。

# cd /etc/fdfs

# cp tracker.conf.sample tracker.conf

# vim tracker.conf

② 编辑tracker.conf ，标红的需要修改下，其它的默认即可。

# 配置文件是否不生效，false 为生效

disabled=false

# 提供服务的端口

port=22122

# Tracker 数据和日志目录地址(根目录必须存在,子目录会自动创建)

base\_path=/ljzsg/fastdfs/tracker

# HTTP 服务端口

http.server\_port=80

③ 创建tracker基础数据目录，即base\_path对应的目录

# mkdir -p /ljzsg/fastdfs/tracker

④ 防火墙中打开跟踪端口（默认的22122）

# vim /etc/sysconfig/iptables

添加如下端口行：

-A INPUT -m state --state NEW -m tcp -p tcp --dport 22122 -j ACCEPT

重启防火墙：

# service iptables restart

⑤ 启动Tracker

初次成功启动，会在 /ljzsg/fdfsdfs/tracker/ (配置的base\_path)下创建 data、logs 两个目录。

可以用这种方式启动

# /etc/init.d/fdfs\_trackerd start

也可以用这种方式启动，前提是上面创建了软链接，后面都用这种方式

# service fdfs\_trackerd start

查看 FastDFS Tracker 是否已成功启动 ，22122端口正在被监听，则算是Tracker服务安装成功。

# netstat -unltp|grep fdfs

https://images2017.cnblogs.com/blog/856154/201710/856154-20171011121344184-1089101646.png

关闭Tracker命令：

# service fdfs\_trackerd stop

⑥ 设置Tracker开机启动

# chkconfig fdfs\_trackerd on  
  
或者：  
# vim /etc/rc.d/rc.local  
加入配置：  
/etc/init.d/fdfs\_trackerd start

⑦ tracker server 目录及文件结构

Tracker服务启动成功后，会在base\_path下创建data、logs两个目录。目录结构如下：

${base\_path}

|\_\_data

| |\_\_storage\_groups.dat：存储分组信息

| |\_\_storage\_servers.dat：存储服务器列表

|\_\_logs

| |\_\_trackerd.log： tracker server 日志文件

## 4、配置 FastDFS 存储 (Storage)

① 进入 /etc/fdfs 目录，复制 FastDFS 存储器样例配置文件 storage.conf.sample，并重命名为 storage.conf

# cd /etc/fdfs

# cp storage.conf.sample storage.conf  
# vim storage.conf

② 编辑storage.conf

标红的需要修改，其它的默认即可。

# 配置文件是否不生效，false 为生效

disabled=false

# 指定此 storage server 所在 组(卷)

group\_name=group1

# storage server 服务端口

port=23000

# 心跳间隔时间，单位为秒 (这里是指主动向 tracker server 发送心跳)

heart\_beat\_interval=30

# Storage 数据和日志目录地址(根目录必须存在，子目录会自动生成)

base\_path=/ljzsg/fastdfs/storage

# 存放文件时 storage server 支持多个路径。这里配置存放文件的基路径数目，通常只配一个目录。

store\_path\_count=1

# 逐一配置 store\_path\_count 个路径，索引号基于 0。

# 如果不配置 store\_path0，那它就和 base\_path 对应的路径一样。

#store\_path0=/ljzsg/fastdfs/file

# FastDFS 存储文件时，采用了两级目录。这里配置存放文件的目录个数。

# 如果本参数只为 N（如： 256），那么 storage server 在初次运行时，会在 store\_path 下自动创建 N \* N 个存放文件的子目录。

subdir\_count\_per\_path=256

# tracker\_server 的列表 ，会主动连接 tracker\_server

# 有多个 tracker server 时，每个 tracker server 写一行

tracker\_server=file.ljzsg.com:22122  
  
# 允许系统同步的时间段 (默认是全天) 。一般用于避免高峰同步产生一些问题而设定。  
sync\_start\_time=00:00  
sync\_end\_time=23:59

# 访问端口  
http.server\_port=80

③ 创建Storage基础数据目录，对应base\_path目录

# mkdir -p /ljzsg/fastdfs/storage

# 这是配置的store\_path0路径（我没有用这个）

# mkdir -p /ljzsg/fastdfs/file

④ 防火墙中打开存储器端口（默认的 23000）

# vim /etc/sysconfig/iptables

添加如下端口行：

-A INPUT -m state --state NEW -m tcp -p tcp --dport 23000 -j ACCEPT

重启防火墙：

# service iptables restart

https://images2017.cnblogs.com/blog/856154/201710/856154-20171011133233777-1903096242.png

⑤ 启动 Storage

启动Storage前确保Tracker是启动的。初次启动成功，会在 /ljzsg/fastdfs/storage 目录下创建 data、 logs 两个目录。

可以用这种方式启动

# /etc/init.d/fdfs\_storaged start

也可以用这种方式，后面都用这种

# service fdfs\_storaged start

查看 Storage 是否成功启动，23000 端口正在被监听，就算 Storage 启动成功。

# netstat -unltp|grep fdfs

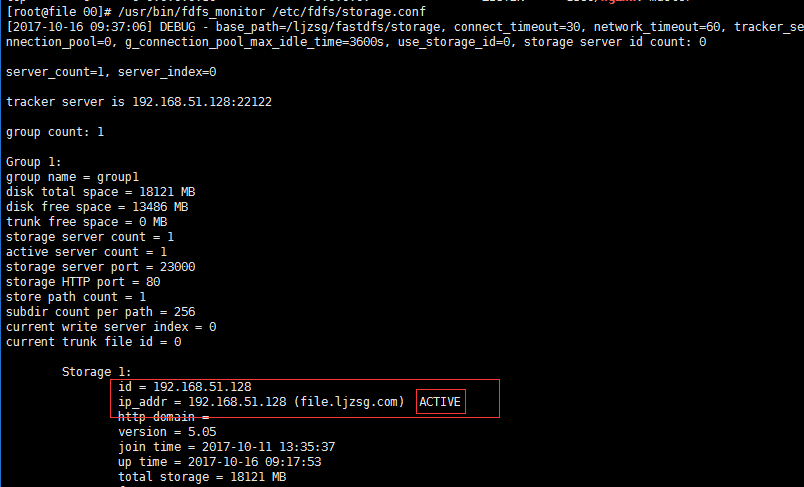
https://images2017.cnblogs.com/blog/856154/201710/856154-20171011134723387-604894314.png

关闭Storage命令：

# service fdfs\_storaged stop

查看Storage和Tracker是否在通信：

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



⑥ 设置 Storage 开机启动

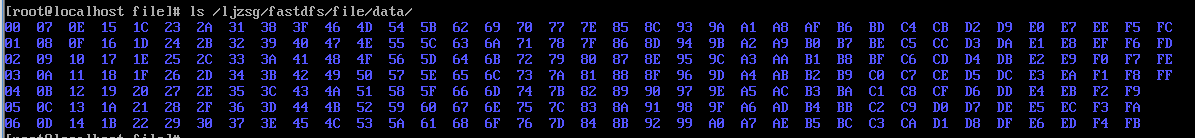
# chkconfig fdfs\_storaged on

或者：  
# vim /etc/rc.d/rc.local  
加入配置：  
/etc/init.d/fdfs\_storaged start

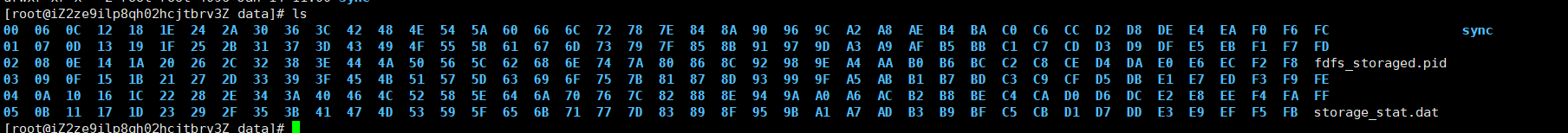
⑦ Storage 目录

同 Tracker，Storage 启动成功后，在base\_path 下创建了data、logs目录，记录着 Storage Server 的信息。

在 store\_path0 目录下，创建了N\*N个子目录：



没有配置store\_path0，在base\_path/data目录下创建了N\*N个子目录



## 5、文件上传测试

① 修改 Tracker 服务器中的客户端配置文件

# cd /etc/fdfs

# cp client.conf.sample client.conf

# vim client.conf

修改如下配置即可，其它默认。

# Client 的数据和日志目录

base\_path=/ljzsg/fastdfs/client

# Tracker端口

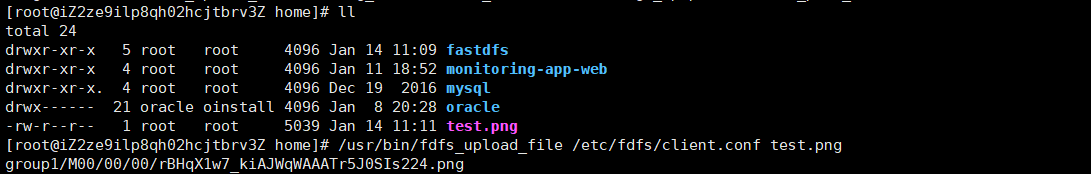
tracker\_server=file.ljzsg.com:22122

② 上传测试

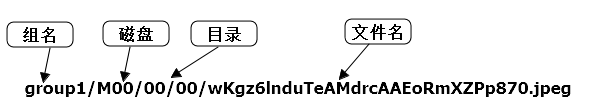
 在linux内部执行如下命令上传 namei.jpeg 图片

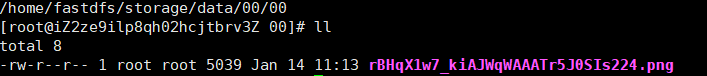
# /usr/bin/fdfs\_upload\_file /etc/fdfs/client.conf test.png

上传成功后返回文件ID号：group1/M00/00/00/rBHqX1w7\_kiAJWqWAAATr5J0SIs224.png



返回的文件ID由group、存储目录、两级子目录、fileid、文件后缀名（由客户端指定，主要用于区分文件类型）拼接而成。





# Nginx配置

location /fastdfs/ {

#图片存储路径

alias /home/logdata/fastdfs/storage/data/;

autoindex on;

}