CentOS7+FastDFS+Nginx安装教程

#### FastDFS介绍

##### 图片服务器介绍

要实现图片上传功能，需要有一个图片服务器。图片服务器的特点：

1. 存储空间可扩展
2. 提供统一的访问方式
3. 访问效率高

原来使用Tomcat配置虚拟路径的方式不能满足需求。

企业中的图片服务器有两个方案：

1. 使用FastDFS分布式文件系统。

2. 使用Nginx服务器。

##### 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、横向扩展等机制，并注重高可用、高性能等指标，使用FastDFS很容易搭建一套高性能的文件服务器集群提供文件上传、下载等服务。

##### 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Tracker server调度最终由Storage server完成文件上传和下载。

Tracker server作用是负载均衡和调度，通过Tracker server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。

如下图：



Tracker 管理集群

收集信息，处理信息

为了保证高可用，可以搭建集群

Storage 保存文件

分为很多组，组和组之间的数据不一样

组内成员数据是一样的，保证数据的高可用

可以增加组，达到扩容的效果，

##### 文件上传流程

文件上传流程如下图（时序图）：



客户端上传文件后存储服务器将文件ID返回给客户端，此文件ID用于以后访问该文件的索引信息。文件索引信息包括：组名，虚拟磁盘路径，数据两级目录，文件名。



* 组名：文件上传后所在的storage组名称，在文件上传成功后有storage服务器返回，需要客户端自行保存。
* 虚拟磁盘路径：storage配置的虚拟路径，与磁盘选项store\_path\*对应。如果配置了store\_path0则是M00，如果配置了store\_path1则是M01，以此类推。
* 数据两级目录：storage服务器在每个虚拟磁盘路径下创建的两级目录，用于存储数据文件。两级目录的范围都是 00~FF
* 文件名：与文件上传时不同。是由存储服务器根据特定信息生成，文件名包含：源存储服务器IP地址、文件创建时间戳、文件大小、随机数和文件拓展名等信息。

##### 文件下载流程

文件下载流程如下图



tracker根据请求的文件路径即文件ID 来快速定义文件。

比如请求下边的文件：



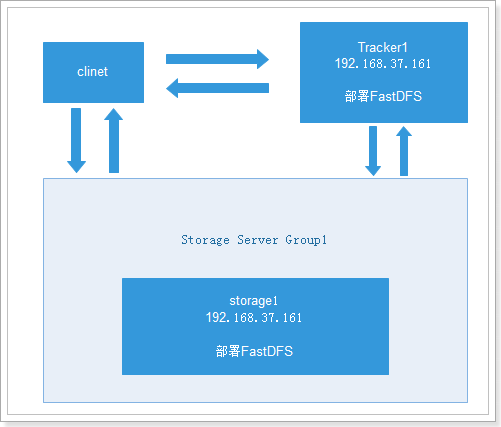
1.通过组名tracker能够很快的定位到客户端需要访问的存储服务器组是group1，并选择合适的存储服务器提供客户端访问。

2.存储服务器根据“文件存储虚拟磁盘路径”和“数据文件两级目录”可以很快定位到文件所在目录，并根据文件名找到客户端需要访问的文件。

#### 实现文件服务器

##### 架构图

本课程使用的FastDFS安装架构图如下：



**1 安装libfastcommon**

1.1 获取libfastcommon安装包：

wget https://github.com/happyfish100/libfastcommon/archive/V1.0.39.tar.gz

1.2 解压安装包：tar -zxvf V1.0.39.tar.gz

1.3 进入目录：cd libfastcommon-1.0.39

1.4 执行编译：./make.sh

1.5 安装：./make.sh install

|  |
| --- |
| 可能遇到的问题：  -bash: make: command not found  -bash: gcc: command not found  解决方案：  debian通过apt-get install gcc make安装  centos通过yum -y install gcc make安装 |

#### 2 安装FastDFS

2.1 获取fdfs安装包：

wget https://github.com/happyfish100/fastdfs/archive/V5.11.tar.gz

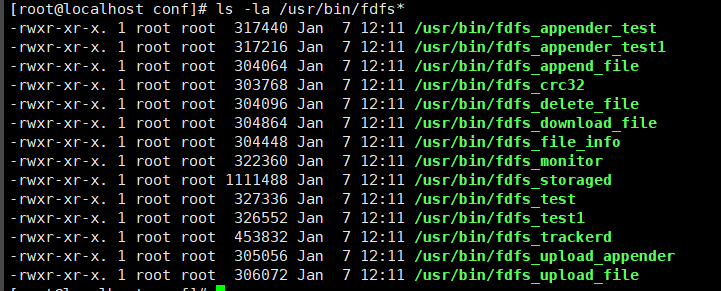
2.2 解压安装包：tar -zxvf V5.11.tar.gz

2.3 进入目录：cd fastdfs-5.11

2.4 执行编译：./make.sh

2.5 安装：./make.sh install

2.6查看可执行命令：ls -la /usr/bin/fdfs\*



#### 3 配置Tracker服务

3.1 进入/etc/fdfs目录，有四个.sample后缀的文件（自动生成的fdfs模板配置文件），通过cp命令拷贝tracker.conf.sample，删除.sample后缀作为正式文件：

3.1.1 cd /etc/fdfs/

3.1.2 cp tracker.conf.sample tracker.conf

3.1.3 编辑tracker.conf: vim tracker.conf

修改相关参数：

base\_path=/home/fastdfs/tracker #tracker存储data和log的跟路径

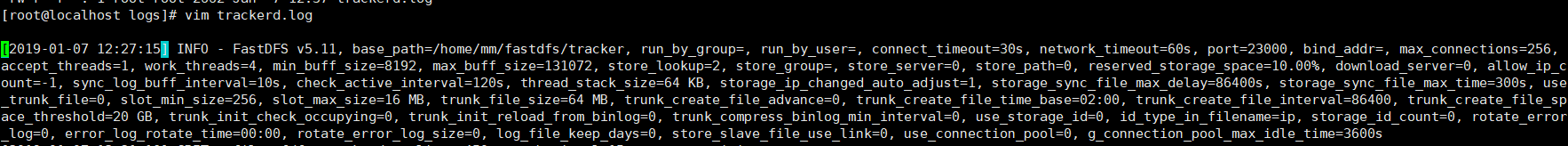
port=22122 #tracker默认22122

http.server\_port=80 #http端口，需要和nginx相同

3.2 启动tracker（支持start|stop|restart）  
 /usr/bin/fdfs\_trackerd /etc/fdfs/tracker.conf start

3.3 查看tracker启动日志：

进入刚刚指定的base\_path(/home/fastdfs/tracker)中有个logs目录，查看tracker.log文件



3.4 查看端口情况：netstat -apn|grep fdfs



|  |
| --- |
| 可能遇到的报错：  /usr/bin/fdfs\_trackerd: error while loading shared libraries: libfastcommon.so: cannot open shared object file: No such file or directory  解决方案：建立libfastcommon.so软链接  ln -s /usr/lib64/libfastcommon.so /usr/local/lib/libfastcommon.so  ln -s /usr/lib64/libfastcommon.so /usr/lib/libfastcommon.so |

#### 4 配置Storage服务

4.1 cd /etc/fdfs

进入/etc/fdfs目录，有cp命令拷贝storage.conf.sample，删除.sample后缀作为正式文件;

4.2 编辑storage.conf：vim storage.conf

修改相关参数：

base\_path=/home /fastdfs/storage #storage存储data和log的跟路径，必须提前创建好

port=23000 #storge默认23000，同一个组的storage端口号必须一致

group\_name=group1 #默认组名，根据实际情况修改

store\_path\_count=1 #存储路径个数，需要和store\_path个数匹配

store\_path0=/home/mm/fastdfs/storage #如果为空，则使用base\_path

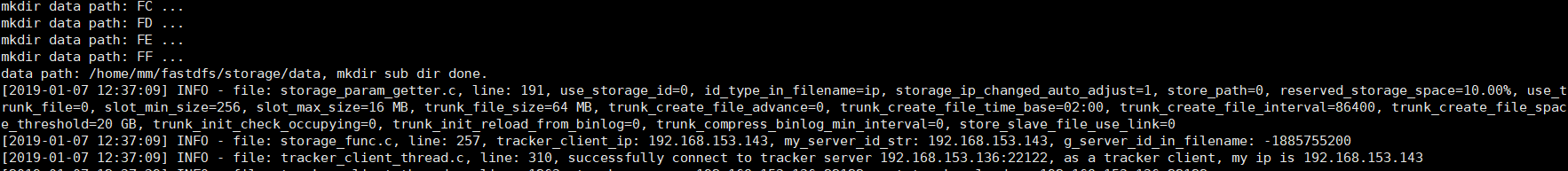
tracker\_server=10.122.149.211:22122 #配置该storage监听的tracker的ip和port

4.3 启动storage（支持start|stop|restart）：

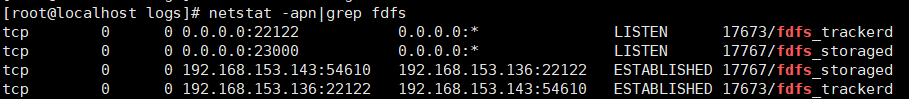
/usr/bin/fdfs\_storaged /etc/fdfs/storage.conf start

4.4 查看storage启动日志：进入刚刚指定的base\_path(/home/fastdfs/storage)中有个logs目录，查看storage.log文件

4.5 此时再查看tracker日志：发现已经开始选举，并且作为唯一的一个tracker，被选举为leader

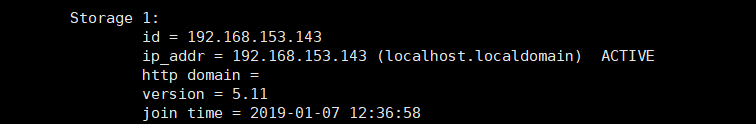


4.6 查看端口情况：netstat -apn|grep fdfs



4.8 通过monitor来查看storage是否成功绑定：

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



#### 5 安装Nginx和fastdfs-nginx-module模块

5.1 获取Nginx安装包

wget http://nginx.org/download/nginx-1.15.8.tar.gz

5.2 下载fastdfs-nginx-module安装包

wget https://github.com/happyfish100/fastdfs-nginx-module/archive/V1.20.tar.gz

5.3 解压nginx：tar -zxvf nginx-1.15.8.tar.gz

5.4 解压fastdfs-nginx-module：tar -xvf V1.20.tar.gz -C /usr/local/src

5.5 进入nginx目录：cd nginx-1.15.8安装依赖的库

yum -y install libpcre3 libpcre3-dev openssl libssl-dev libperl-dev

5.6 配置，并加载fastdfs-nginx-module模块：

./configure --prefix=/usr/local/nginx --add-module=/usr/local/src/fastdfs-nginx-module-1.20/src/

5.7 编译安装：

make

make install

|  |
| --- |
| 可能的报错：  /usr/include/fastdfs/fdfs\_define.h:15:27: fatal error: common\_define.h: No such file or directory  解决方案：修改fastdfs-nginx-module-1.20/src/config文件，然后重新第5.6步开始  ngx\_module\_incs="/usr/include/fastdfs /usr/include/fastcommon/"  CORE\_INCS="$CORE\_INCS /usr/include/fastdfs /usr/include/fastcommon/" |

5.8 查看安装路径：whereis nginx



5.9 启动、停止：

cd /usr/local/nginx/sbin/

./nginx #启动

./nginx -s stop #此方式相当于先查出nginx进程id再使用kill命令强制杀掉进程

./nginx -s quit #此方式停止步骤是待nginx进程处理任务完毕进行停止

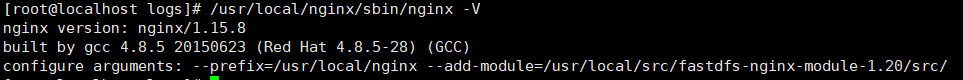
./nginx -s reload

5.10 验证启动状态：wget <http://本机IP地址>

|  |
| --- |
| 如果没有反应：  修改nginx.conf 配置文件： server {  listen 80;  server\_name 192.168.153.136;( 为当前机器ip ) |

5.11 查看此时的nginx版本：发现fastdfs模块已经安装好了

/usr/local/nginx/sbin/nginx -V



#### 6 配置Nginx和fastdfs-nginx-module模块

6.1 配置mod-fastdfs.conf，并拷贝到/etc/fdfs文件目录下

cd fastdfs-nginx-module-1.20/src/

cp mod\_fastdfs.conf /etc/fdfs

6.2 进入/etc/fdfs修改mod-fastdfs.conf：

base\_path=/home/fastdfs

tracker\_server=192.168.153.136:22122 #tracker的地址

url\_have\_group\_name=true #url是否包含group名称

storage\_server\_port=23000 #需要和storage配置的相同

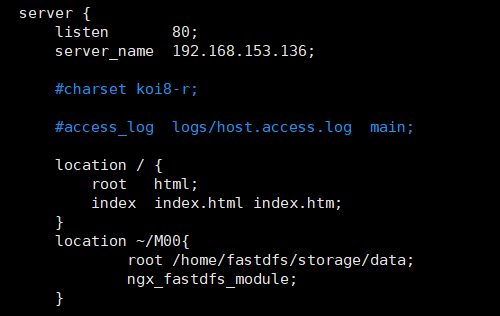
store\_path\_count=1 #存储路径个数，需要和store\_path个数匹配

store\_path0=/home /fastdfs/storage #文件存储的位置

6.3 配置nginx，80端口server增加location如图：

cd /usr/local/nginx/conf/

vi nginx.conf



6.4 最后需要拷贝fastdfs解压目录中的http.conf和mime.types：

cd /usr/local/src/fastdfs-5.11/conf

cp mime.types http.conf /etc/fdfs/

#### 7 FastDFS常用命令测试

7.1 上传文件

7.1.1 进入/etc/fdfs目录，用cp命令拷贝client.conf.sample，删除.sample后缀作为正式文件;

7.1.2 修改client.conf相关配置：

base\_path=/home/fastdfs/tracker #tracker服务器文件路径

tracker\_server=192.168.153.136:22122 #tracker服务器IP地址和端口号

http.tracker\_server\_port=80 # tracker服务器的http端口号，必须和tracker的设置对应起来

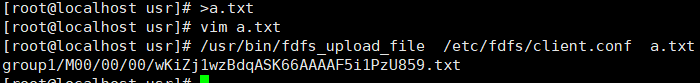
7.1.3 新建一个测试文档1.txt，内容为ssssssssssss

命令：

/usr/bin/fdfs\_upload\_file <config\_file> <local\_filename>

示例：

/usr/bin/fdfs\_upload\_file /etc/fdfs/client.conf a.txt



group1/M00/00/00/wKiZj1wzBdqASK66AAAAF5i1PzU859.txt

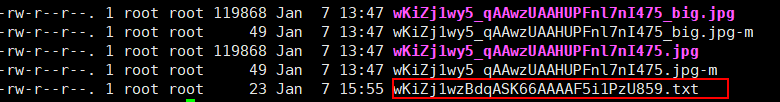
7.1.4 返回值分析

组名：group1

磁盘：M00

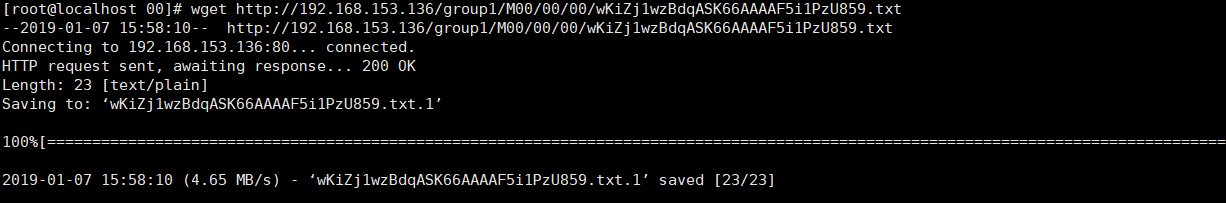
目录：00/00

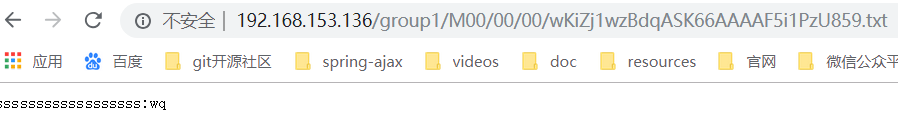
文件名称：wKiZj1wzBdqASK66AAAAF5i1PzU859.txt

7.1.5 查看结果，进入storage的data目录：  


7.2 通过wget和浏览器方式访问成功：

wget <http://192.168.153.136/group1/M00/00/00/wKiZj1wzBdqASK66AAAAF5i1PzU859.txt>





7.3 下载文件：

命令：

/usr/bin/fdfs\_download\_file <config\_file> <file\_id> [local\_filename]

示例：

/usr/bin/fdfs\_download\_file /etc/fdfs/client.conf group1/M00/00/00/CnqV01trmeyAbAN0AAAABLh3frE677.txt a.txt

7.4 删除文件：

命令：

/usr/bin/fdfs\_delete\_file <config\_file> <file\_id>

示例：

/usr/bin/fdfs\_delete\_file /etc/fdfs/client.conf group1/M00/00/00/CnqV01trmeyAbAN0AAAABLh3frE677.txt

查看结果，进入storage的data目录文件不存在，通过wget再次获取404：

#### 8确认服务是否可用

8.1 确认tracker是否启动

/usr/bin/fdfs\_trackerd /etc/fdfs/tracker.conf restart

如下图关闭线程再执行则表示成功



8.2 确认storage是否启动

/usr/bin/fdfs\_storaged /etc/fdfs/storage.conf restart

如下图关闭线程再执行则表示成功

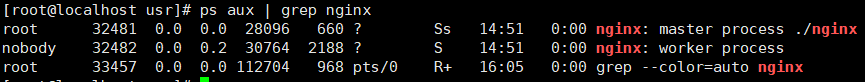


如果不成功，同样的命令再执行一次

8.3 确认Nginx是否可用

ps aux | grep nginx

查看线程是否存在



9 java代码客户端测试

保证开放 22122 和23000 端口，供java连接

9.1 添加依赖包

<dependency>

<groupId>org.csource</groupId>

<artifactId>fastdfs-client-java</artifactId>

<version>1.27-SNAPSHOT</version>

</dependency>

获取地址：关注微信公众号：浅醉JAVA

后台留言免费获取（公众号测试阶段，无法自动获取，留言联系）

附有简单API使用方法