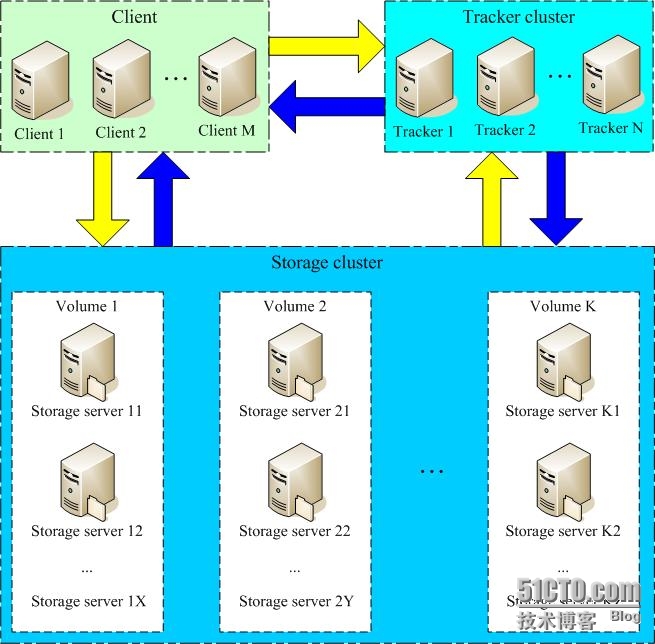
**一、背景描述**

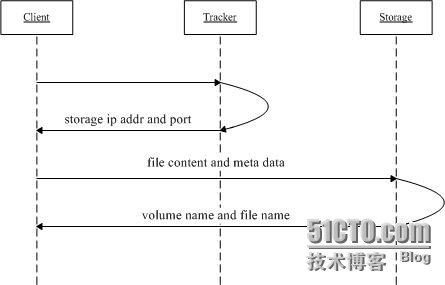
之前公司的图片服务器已经不能胜任了（直接使用tomcat），需要重新搭建图片服务器，这两天研究了下FastDFS，感觉挺好用记录下来以供日后参考。

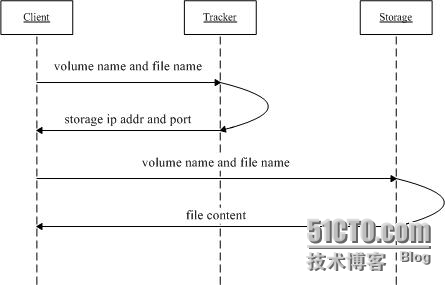
**二、FastDFS官方介绍**

FastDFS是一款类Google FS的开源分布式文件系统，它用纯C语言实现，支持Linux、FreeBSD、AIX等UNIX系统。它只能通过 专有API对文件进行存取访问，不支持POSIX接口方式，不能mount使用。准确地讲，Google FS以及FastDFS、mogileFS、 HDFS、TFS等类Google FS都不是系统级的分布式文件系统，而是应用级的分布式文件存储服务。

1、FastDFS架构图 

1. Tracker cluster中各个tracker server相互独立，不进行相互通信。
2. Storage cluster中各个storage组（Volume1,Volume2...）相互独立，不进行相互通信，也就是说各个组之间保存的数据是不相同的。但是各个组中的storage server之间是属于互相备份的关系，也就是说storage server之间保存相同的数据。
3. 每个storage server会启动一个单独的线程主动向Tracker cluster中每个tracker server报告其状态信息，包括磁盘使用情况，文件同步情况及文件上传下载次数统计等信息。

2、文件上传和下载的时序图 

1. Client通过Tracker server将文件上传到Storage server。
2. Tracker server向Client返回一台可用的Storage server的IP地址和端口号。
3. Client直接通过Tracker server返回的IP地址和端口与其中一台Storage server建立连接并进行文件上传。
4. 上传完成，Storage server返回Client一个文件ID，文件上传结束。 
5. Client通过Tracker server下载指定Storage组中某个Storage server上的某个文件（文件名包括Storage组名称）。
6. Tracker server向Client返回一台可用的Storage server的IP地址和端口号。
7. Client直接通过Tracker server返回的IP地址和端口与其中一台Storage server建立连接并进行文件下载。

**三、安装并配置FastDFS**

**1、运行环境及相关软件**

* CentOS 6.5
* FastDFS\_v5.08.tar.gz
* nginx-1.8.1.tar.gz
* fastdfs-nginx-module\_v1.16.tar.gz
* libfastcommon

**2、服务器规划**

| **服务器名称** | **IP地址和端口** | **备注** |
| --- | --- | --- |
| fastdfs-tracker | 192.168.0.8:22122 | 跟踪服务器/调度服务器 |
| fastdfs-storage | 192.168.0.9:23000 | 存储服务器 |

**3、安装FastDFS**

1. 首先创建工具目录（非必须）

mkdir -p /root/tools

**cd** /root/tools/

1. 下载并安装FastDFS依赖包libfastcommon

wget https://codeload.github.com/happyfish100/libfastcommon/zip/**master**

**unzip** **master**

**cd** libfastcommon-master/

./make.sh && ./make.sh install

1. 下载并安装FastDFS

wget http://jaist.dl.sourceforge.net/project/fastdfs/FastDFS%20Server%20Source%20Code/FastDFS%20Server%20with%20PHP%20Extension%20Source%20Code%20V5.08/FastDFS\_v5.08.tar.gz

tar xf FastDFS\_v5.08.tar.gz

cd FastDFS

./make.sh && ./make.sh install

**如果安装出现 ./make.sh:行178: perl: 未找到命令 执行下边代码 然后从新执行**

yum -y install zlib zlib-devel pcre pcre-devel gcc gcc-c++ openssl openssl-devel libevent libevent-devel perl unzip net-tools wget

**采用默认方式安装后的文件及目录：**

1. 服务脚本：

[root@fastdfs-tracker ~]*# ll /etc/init.d/ |grep fdfs*

-rwxr-xr-x. 1 root root 918 4月 22 22:08 fdfs\_storaged

-rwxr-xr-x. 1 root root 920 4月 22 22:08 fdfs\_trackerd

1. 样例配置文件

[root@fastdfs-tracker ~]*# ll /etc/fdfs/*

总用量 20

-rw-r--r--. 1 root root 1461 4月 22 22:08 client.conf.sample

-rw-r--r--. 1 root root 7927 4月 22 22:08 storage.conf.sample

-rw-r--r--. 1 root root 7200 4月 22 22:08 tracker.conf.sample

1. 命令行工具

[root@fastdfs-tracker ~]*# ll /usr/bin|grep fdfs*

-rwxr-xr-x. 1 root root 252272 4月 22 22:08 fdfs\_appender\_test

-rwxr-xr-x. 1 root root 252225 4月 22 22:08 fdfs\_appender\_test1

-rwxr-xr-x. 1 root root 242449 4月 22 22:08 fdfs\_append\_file

-rwxr-xr-x. 1 root root 242013 4月 22 22:08 fdfs\_crc32

-rwxr-xr-x. 1 root root 242508 4月 22 22:08 fdfs\_delete\_file

-rwxr-xr-x. 1 root root 243627 4月 22 22:08 fdfs\_download\_file

-rwxr-xr-x. 1 root root 243369 4月 22 22:08 fdfs\_file\_info

-rwxr-xr-x. 1 root root 255657 4月 22 22:08 fdfs\_monitor

-rwxr-xr-x. 1 root root 863913 4月 22 22:08 fdfs\_storaged

-rwxr-xr-x. 1 root root 258712 4月 22 22:08 fdfs\_test

-rwxr-xr-x. 1 root root 257881 4月 22 22:08 fdfs\_test1

-rwxr-xr-x. 1 root root 365232 4月 22 22:08 fdfs\_trackerd

-rwxr-xr-x. 1 root root 243547 4月 22 22:08 fdfs\_upload\_appender

-rwxr-xr-x. 1 root root 244453 4月 22 22:08 fdfs\_upload\_file

注意：虽然FastDFS区分tracker和storage服务器，但是安装的软件及步骤均相同，只是不同的配置文件而已，因此以上安装适用tracker server和storage server

**4、配置跟踪服务器（tracker server）**

1. 拷贝tracker server和client端样例配置文件并重命名

[root@fastdfs-tracker ~]# cp /etc/fdfs/tracker.conf.sample /etc/fdfs/tracker.conf

[root@fastdfs-storage ~]# cp /etc/fdfs/client.conf.sample /etc/fdfs/client.conf

1. 编辑tracker server配置文件tracker.conf，需要修改内容如下：

disabled=false（默认为false，表示是否无效）

port=22122（默认为22122）

base\_path=/data/fastdfs/tracker

1. 编辑client端的配置文件client.conf，需要修改内容如下

base\_path=/data/fastdfs/tracker

tracker\_server=172.18.10.232:22122

1. 创建tracker server数据目录

**mkdir** -p /**data**/fastdfs/tracker

1. 测试启动tracker server，启动成功会自动在/data/fastdfs/tracker目录新建data和logs目录

[root@fastdfs-tracker ~]*# cd /data/fastdfs/tracker/*

[root@fastdfs-tracker tracker]*# ls*

[root@fastdfs-tracker tracker]*# /etc/init.d/fdfs\_trackerd start*

Starting FastDFS tracker server:

[root@fastdfs-tracker tracker]*# ss -lntup|grep 22122*

tcp LISTEN 0 128 \*:22122 \*:\* users:(("fdfs\_trackerd",3785,5))

[root@fastdfs-tracker tracker]*# ls*

data logs

1. 关闭tracker server

[root@fastdfs-tracker tracker]*# /etc/init.d/fdfs\_trackerd stop*

waiting **for** pid [3785] **exit** ...

pid [3785] **exit**.

**5、配置存储服务器（storage server）**

1. 拷贝storage server样例配置文件并重命名

[root@fastdfs-storage ~]# cp /etc/fdfs/storage.conf.sample /etc/fdfs/storage.conf

1. 编辑storage server配置文件storage.conf，需要修改内容如下：

disabled=false（默认为false，表示是否无效）

port=23000（默认为23000）

base\_path=/data/fastdfs/storage

tracker\_server=172.18.10.232:22122

store\_path0=/data/fastdfs/storage

http.server\_port=8888（默认为8888，nginx中配置的监听端口那之一致）

1. 创建storage server数据目录

[root@fastdfs-storage ~]**# mkdir -p /data/fastdfs/storage**

1. 测试启动storage server，启动成功会自动在/data/fastdfs/tracker目录新建data和logs目录（启动storage server的前提是tracker server必须事先已启动）

[root@fastdfs-storage ~]*# cd /data/fastdfs/storage/*

[root@fastdfs-storage storage]*# ls*

[root@fastdfs-storage storage]*# /etc/init.d/fdfs\_storaged start*

Starting FastDFS storage server:

[root@fastdfs-storage storage]*# ss -lntup|grep 23000*

tcp LISTEN 0 128 \*:23000 \*:\* users:(("fdfs\_storaged",3786,5))

[root@fastdfs-storage storage]*# ls*

data logs

**6. 文件上传测试**

执行如下文件上传命令：

[root@fastdfs-tracker ~]# /usr/bin/fdfs\_upload\_file /etc/fdfs/client.conf /home/oldcat/imgs/test.jpg

group1/M00/00/00/rBIK6VcaP0aARXXvAAHrUgHEviQ394.jpg

返回文件ID即说明文件已经上传成功

**四、存储服务器（storage server）安装并配置nginx**

1. 下载并安装fastdfs-nginx-module模块

注：FastDFS通过Tracker服务器,将文件放在Storage服务器存储，但是同组存储服务器之间需要进入文件复制，有同步延迟的问题。假设Tracker服务器将文件上传到了192.168.4.125，上传成功后文件ID已经返回给客户端。此时FastDFS存储集群机制会将这个文件同步到同组存储192.168.4.126，在文件还没有复制完成的情况下，客户端如果用这个文件ID在192.168.4.126上取文件,就会出现文件无法访问的错误。而fastdfs-nginx-module可以重定向文件连接到源服务器取文件,避免客户端由于复制延迟导致的文件无法访问错误。

[root@fastdfs-storage tools]*# wget http://nchc.dl.sourceforge.net/project/fastdfs/FastDFS%20Nginx%20Module%20Source%20Code/fastdfs-nginx-module\_v1.16.tar.gz*

[root@fastdfs-storage tools]*# tar xf fastdfs-nginx-module\_v1.16.tar.gz*

[root@fastdfs-storage tools]*# cd fastdfs-nginx-module/src/*

[root@fastdfs-storage src]*# vim config*

编辑config文件，执行如下命令进行批量替换并保存退出

:%s+/usr/local/+/usr/+g

1. 拷贝fastdfs-nginx-module模块中配置文件到/etc/fdfs目录中并编辑

[root@fastdfs-storage ~]*# cp /home/oldcat/tools/fastdfs-nginx-module/src/mod\_fastdfs.conf /etc/fdfs/*

[root@fastdfs-storage ~]*# vim /etc/fdfs/mod\_fastdfs.conf*

修改内容如下：

connect\_timeout=10

base\_path=/tmp（默认为/tmp）

tracker\_server=192.168.0.8:22122

storage\_server\_port=23000（默认配置为23000）

url\_have\_group\_name = true

store\_path0=/data/fastdfs/storage

group\_name=group1（默认配置为group1）

1. 安装nginx依赖库

[root@fastdfs-storage nginx-1.8.1]**# yum install -y pcre-devel zlib-devel**

1. 下载并安装nginx

[root@fastdfs-storage tools]**# wget http://nginx.org/download/nginx-1.8.1.tar.gz**

[root@fastdfs-storage tools]**# tar xf nginx-1.8.1.tar.gz**

[root@fastdfs-storage tools]**# cd nginx-1.8.1**

[root@fastdfs-storage nginx-1.8.1]**# ./configure --prefix=/application/nginx/ --add-module=../fastdfs-nginx-module/src/**

[root@fastdfs-storage nginx-1.8.1]**# make && make install**

1. 拷贝FastDFS中的部分配置文件到/etc/fdfs目录中

[root@fastdfs-storage ~]# **cp** /home/oldcat/tools/FastDFS/**conf**/http.**conf** /etc/fdfs/

[root@fastdfs-storage ~]# **cp** /home/oldcat/tools/FastDFS/**conf**/mime.types /etc/fdfs/

1. 配置nginx，如下所示：

[root@fastdfs-storage ~]# vim /application/nginx/conf/nginx.conf

user root;

worker\_processes 1;

events {

worker\_connections 1024;

}

http {

include mime.types;

default\_type application/octet-stream;

sendfile on;

keepalive\_timeout 65;

server {

listen 8888;

server\_name localhost;

location ~/group[0-9]/ {

ngx\_fastdfs\_module;

}

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

location = /50x.html {

root html;

}

}

}

说明：

* "user root"是解决下载操作时报404的问题
* 8888端口号与/etc/fdfs/storage.conf中的http.server\_port=8888相对应
* storage对应有多个group的情况下，访问路径带group名称，例如：/group1/M00/00/00/\*\*，对应nginx配置：

**location** **~/group**[0-9]/ {

ngx\_fastdfs\_module;

1. 拷贝nginx服务到/etc/init.d/目录下并启动

[root@fastdfs-storage ~]*# cp /application/nginx/sbin/nginx /etc/init.d/*

[root@fastdfs-storage ~]*# /etc/init.d/nginx*

[root@fastdfs-storage ~]*# ss -lntup|grep 8888*

tcp LISTEN 0 128 \*:8888 \*:\* users:(("nginx",7308,6),("nginx",7309,6))

8.启动nginx

[root@localhost ~]# /application/nginx/sbin/nginx

ngx\_http\_fastdfs\_set pid=6431

nginx: [emerg] bind() **to** 0.0.0.0:8888 failed (98: Address already **in** use)

nginx: [emerg] bind() **to** 0.0.0.0:8888 failed (98: Address already **in** use)

nginx: [emerg] bind() **to** 0.0.0.0:8888 failed (98: Address already **in** use)

nginx: [emerg] bind() **to** 0.0.0.0:8888 failed (98: Address already **in** use)

nginx: [emerg] bind() **to** 0.0.0.0:8888 failed (98: Address already **in** use)

1. 通过浏览器访问之前已经上传的文件<http://192.168.0.9:8888/group1/M00/00/00/wKgACVldyzKABZGhAACvN4b7WrA208.jpg>

**最后**

1. 以上是单示例，实际可以建立多个追踪和存储服务器集群
2. 以上配置没有考虑防火墙，端口自行放行
3. nginx需要采用root身份运行
4. nginx编译采用简单编译，缺少大量其他功能扩展模块，注意区分。

**参考文档：**

* <http://oldcat1981.blog.51cto.com/10670523/1766810>
* <http://blog.csdn.net/xyang81/article/details/52928230> #这篇写的不错
* FastDFS环境配置及php FastDFS扩展安装
* <http://blog.csdn.net/ssoul_liu/article/details/51059139>

**配置**

添加php扩展模块

fastdfs-tracker上

cd /root/tools/FastDFS/php\_client/

/usr/local/php/bin/phpize

./configure --with-php-config=/usr/local/php/bin/php-config

make && make install

cat /root/tools/FastDFS/php\_client/fastdfs\_client.ini >> /usr/local/php/etc/php.ini

lnmp restart