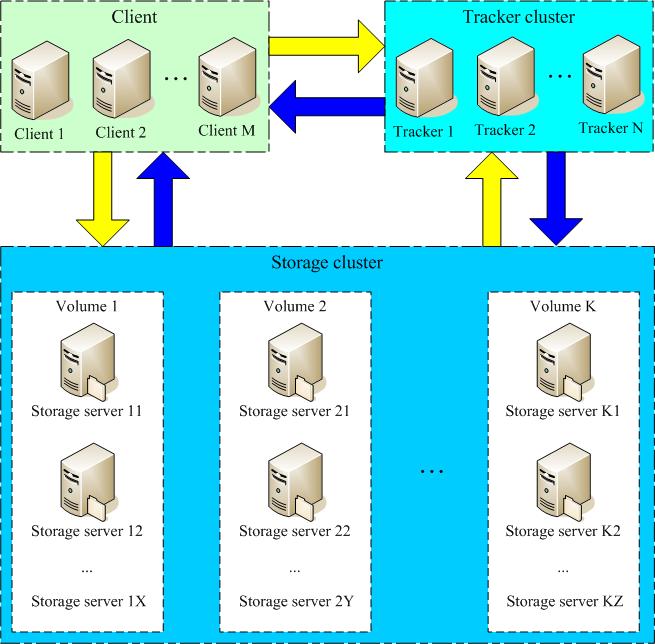
**FastDFS安装配置文档**

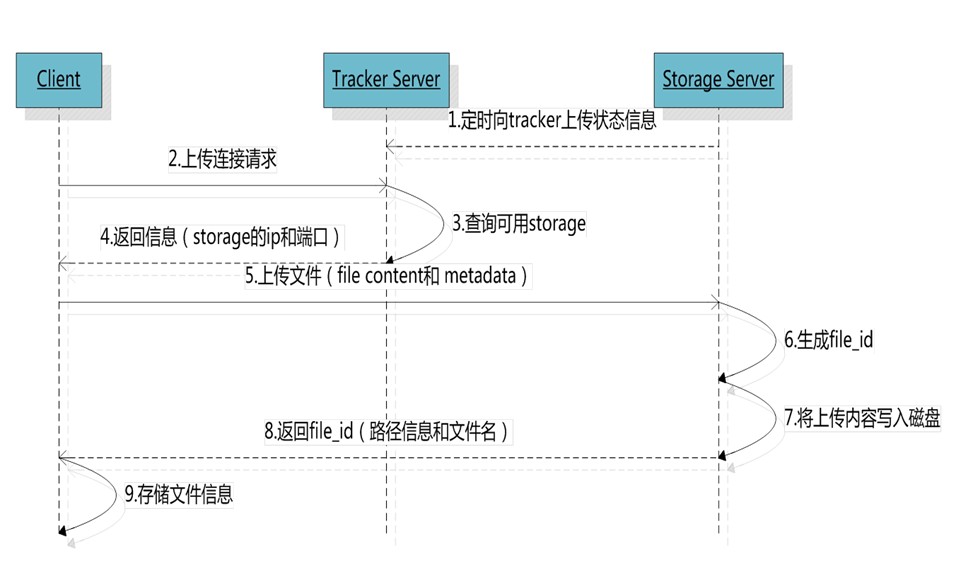
**V1.0**

**闫兴魁**

**2016-10-18**

**版权所有2016-2026**





## 一、安装libevent

在搭建分布式环境之前，需要确保已经安装了libevent，FastDFS安装时对libevent的版本有一定要求。注需要卸载重装系统自带的libevent。使用最新的**stable版本**。本示例中使用的版本为libevent-2.0.21-stable。注意，**此步骤需要在所有机器上都执行，**否则会导致FastDFS安装失败。

**1：判断是否安装了libevent**

>> rpm -qa|grep libevent

如果有输出，则说明已经安装了libevent

Fastdfs对libevent版本有要求，所以先删除系统自带的libevent,然后安装最新稳定版本

在本示例中，输出为libevent-1.4.13-1.el6.x86\_64

**2：卸载已有的libevent**

>> rpm -e libevent --nodeps

**3：安装最新的libevent**

本示例的libevent使用版本为libevent-2.0.21-stable.tar.gz

安装libevent

>> tar zvxf libevent-2.0.21-stable.tar.gz

>> cd libevent-2.0.21-stable

>> ./configure --prefix=/usr

>> make

>>make install

# **[linux make: \*\*\* No targets specified and no makefile found. Stop.](http://www.cnblogs.com/-lpf/p/5588208.html)**

 yum install gcc gcc-c++ autoconf automake

**4：检查是否安装成功**

>> ls –al /usr/lib |grep libevent（或者>> ls –al /usr/local/lib |grep libevent）若有结果，则说明安装正确

注意：

1) 如果是在32为系统上安装，且libevent 的安装目录在/usr/local/lib下，则还需要建立libevent-2.0.so.5 到/usr/lib的软链接，这样其他程序运行时才可以找到libevent库

>> ln –s /usr/local/lib/libevent-2.0.so.5 /usr/lib/libevent-2.0.so.5

2) 如果是在64位系统，需要创建一个libevent-2.0.so.5 到/usr/lib64的软链接

>> ln -s /usr/lib/libevent-2.0.so.5 /usr/lib64/libevent-2.0.so.5 (假设libevent 的安装目录在/usr/lib下)

到此一步，安装libevent工作完毕。

## 二、安装FastDFS

**安装FastDFS**

**-bash: unzip: command not found**

 yum install -y unzip zip

**\*\*\* [../common/fdfs\_global.o] Error 1**

**unzip libfastcommon-master.zip**

 cd libfastcommon-master/

  #  ./make.sh

  # ./make.sh install

>> tar zvxf FastDFS\_v4.06.tar.gz

>> cd FastDFs

>> ./make.sh

>> ./make.sh install

注意上面两步，检查是否出错，如果出错，则说明上面的libevent没有安装好

安装成功后，FastDFS 安装在/usr/local/bin中。配置文件在/etc/fdfs中

至此，FastDFS安装完毕

## 三、 安装nginx以及fastdfs-nginx-module（只需在storage）

FastDFS通过HTTP服务器来提供HTTP服务。为了支持高并发的访问，以及提供负载均衡等高性能的服务，本示例使用nginx作为HTTP服务器，FastDFS为我们提供了Nginx上使用的FastDFS模块（fastdfs-nginx-module）。此步操作只需要在storage上执行。tracker

上可不执行。

**1：fastdfs-nginx-module安装：**

>> tar -zxvf fastdfs-nginx-module\_v1.11.tar.gz (解压后的路径为/home/chenfeic/fastdfs/fastdfs-nginx-module)

Nginx需要依赖第三方的库 nginx的gzip模块需要zlib库，rewrite模块需要pcre库，那么我们来依次安装它们。

**2：zlib库安装:**

>> tar -vxzf zlib-1.2.8.tar.gz （本示例中解压位置为/home/chenfeic/fastdfs/zlib-1.2.8）

>>cd zlib-1.2.8

>> ./configure --prefix=/usr/local/zlib （设置安装路径）

>> make

>> make install

**3：pcre库安装：**

>> tar -vxzf pcre-8.33.tar.gz （本示例中解压位置为--with-pcre= /home/chenfeic/ fastdfs / pcre-8.33 ）

>> cd prce-8.33

>> ./configure --prefix=/usr/local/pcre

--libdir=/usr/local/lib/pcre --includedir=/usr/local/include/pcre

>> make

>> make install

**4：nginx安装：**

>> tar -zxvf nginx-1.4.4.tar.gz

>> cd nginx-1.4.4

>>./configure --prefix=/usr/local/nginx --with-zlib=**/home/chenfeic/fastdfs/zlib-1.2.8**  --with-pcre=**/home/chenfeic/fastdfs/pcre-8.33** --sbin-path=/usr/local/nginx --add-module=/home/chenfeic/fastdfs/fastdfs-nginx-module/src

>> make

>> make install

注意：--with-pcre --with-zlib 后面的路径都是

至此，我们的nginx安装成功，nginx默认是80端口，为了防止与其他服务冲突，建议换成其他端口（后续章节有详细介绍）。

## 四、配置及启动FastDFS

**1：tracker配置(只需在跟踪器节点配置)**

修改tracker的配置文件tracker.conf。该文件位置在/etc/fdfs/tracker.conf。主要修改以下几处，如有其他调整，可参考FastDFS文档自行调整。

>> /etc/fdfs/tracker.conf

base\_path=/home/chenfeic/fastdfs/tracker ====>放置data和log的目录。**这个路径必须存在**。

##include http.conf 修改为#include http.conf ====> **这里一定要注意！是 #include，不是include!!! 如果没有此配置项，则不需要处理**

rotate\_error\_log=true

启动tracker

>> /usr/local/bin/fdfs\_tracker /etc/fdfs/tracker.conf

检查是否正常启动

>> netstat –anp|grep –w fdfs

看是否有输出，且22122端口已启动

**2：storage配置（只需在存储节点配置，同一group节点的配置是一样的）**

**1）修改nginx端口（可选）**

nginx 默认端口号为80，为了防止冲突，可以设置为其他端口，本示例中将其修改为8080。

修改/usr/local/nginx/conf/nginx.conf

>> vim /usr/local/nginx/conf/nginx.conf

Server {  
  **listen 8080(从80改为8080)**

}

**2）支持FastDFS模块**

在nginx的**server配置段中**增加 /group1/M00的location声明

storage $> vim /usr/local/nginx/conf/nginx.conf

location **/group1 /M00** { //**注意：此处如果是在group1就写成group1/M00。如果是group2节点就写group2/M00。依次类推。**

root /home/chenfeic/ fastdfs /storage/data; **//必须保证路径存在，若不存在就创建**

ngx\_fastdfs\_module;

}（其中/home/chenfeic/ fastdfs /storage/为storage中配置的store\_path0路径，**data目录一定要加上**）

**给 storage 的存储目录做一个软连接**

> ln -s /home/chenfeic/ fastdfs /storage/data /home/chenfeic/ fastdfs /storage /data /M00

**3）修改storage配置**

修改storage的配置文件storage.conf。该文件位置在/etc/fdfs/storage.conf。主要修改以下几如有其他调整，可参考FastDFS文档自行调整

group\_name=group1 ====> 此台storage server所属的服务器组名。**此处如果是在group1就写成group1。如果是group2节点就写group2。依次类推。**

base\_path=/home/chenfeic/ fastdfs /storage ====> 放置data和log的目录。**此路径必须保证存在。**

store\_path0=/home/chenfeic/ fastdfs /storage ====> 放置文件的目录，**此路径必须保证存在,建议跟base\_path保持一致（**/home/chenfeic/ fastdfs /storage与nginx server段中的设置保持一致,**）**

rotate\_error\_log=true

http.disabled=true ====> 关闭内置的web server，如果有此配置项就设置，否则就不用设置。

http.server\_port=8080 ====> web server的端口改成8080（与nginx 端口一致）

**4）修改fastdfs-nginx-module配置**

拷贝mod\_fastdfs.conf 到/etc/fdfs目录下

>>cp /home/chenfeic/fastdfs/fastdfs-nginx-module/src/mod\_fastdfs.conf /etc/fdfs/

**5）修改mod\_fastdfs.conf配置文件**

>> vim /etc/fdfs/mod\_fastdfs.conf

base\_path=/home/chenfeic/ fastdfs /storage ====> 放置log的目录,与storage中的配置保持一致。

url\_have\_group\_name = true ====> 在URL中包含group名称。**一定要设置true**

store\_path0=/home/chenfeic/ fastdfs /storage ====> 放置文件的目录，与storage中保持一致

response\_mode=proxy ====> 对文件同步延迟的处理方式，通过redirect跳转和proxy代理两种方式解决

到此，配置已经完成，启动nginx和相关storage

>>/usr/local/nginx/nginx

>> /usr/local/bin/fdfs\_storage /etc/fdfs/ storage.conf

检查是否正常启动

>> netstat –anp|grep –w 8080 看8080端口是否启动

**3：测试是否正常安装**

**此步骤在tracker或者storage都可以测试，本示例中在storage上测试，修改client.conf配置文件。**

**测试**：

>vim hello.txt (创建一个测试的文件)

> /usr/local/bin/fdfs\_test / etc/fdfs/client.conf upload hello.txt

查看结果，看文件是否正确上传

**可以在任何storage机器上执行如下命令，查看fastdfs集群状态，如果各个storage状态都为ACTIVE，则搭建成功。**

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

**关闭storage:**

**/usr/local/bin/stop.sh /usr/local/bin/fdfs\_ storaged /etc/fdfs/storage.conf**

**或者killall fdfs\_storaged**

**重启storage：**

**/usr/local/bin/restart.sh /usr/local/bin/fdfs\_storaged /etc/fdfs/storage.conf**