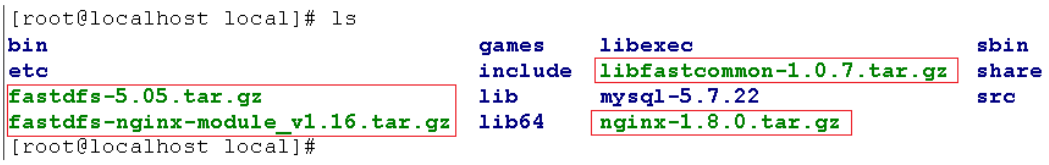
# Centos7安装FastDFS

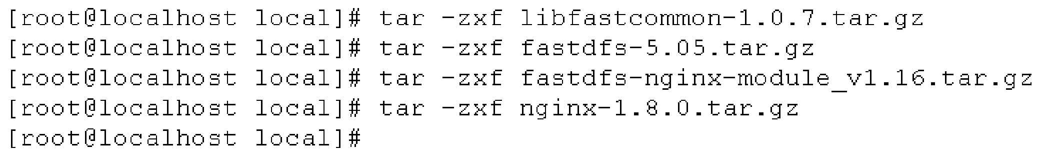
## 离线安装包准备：

将相关的安装包上传到 /usr/local 目录



```
[root@localhost local]# ls
bin                               games    libexec                      sbin
etc                               include  libfastcommon-1.0.7.tar.gz   share
fastdfs-5.05.tar.gz               lib      mysql-5.7.22                 src
fastdfs-nginx-module_v1.16.tar.gz lib64    nginx-1.8.0.tar.gz
[root@localhost local]#
```

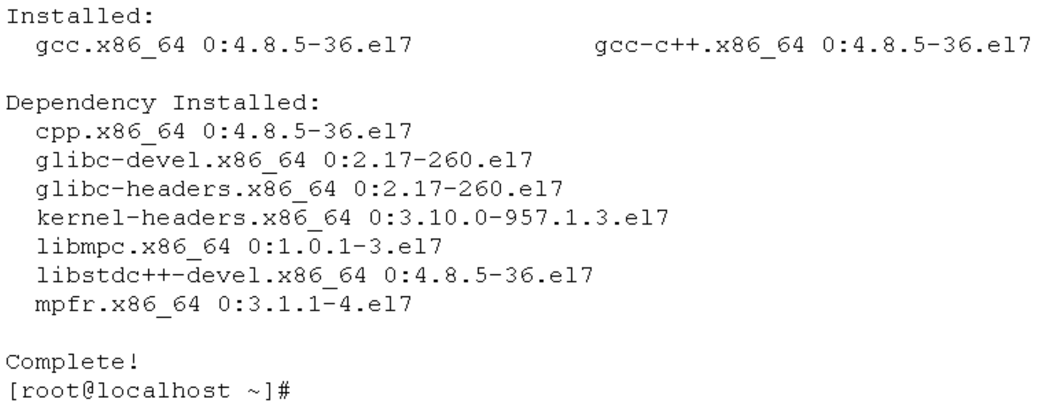
并解压到当前目录



```
[root@localhost local]# tar -zxf libfastcommon-1.0.7.tar.gz
[root@localhost local]# tar -zxf fastdfs-5.05.tar.gz
[root@localhost local]# tar -zxf fastdfs-nginx-module_v1.16.tar.gz
[root@localhost local]# tar -zxf nginx-1.8.0.tar.gz
[root@localhost local]#
```

## 1.安装 gcc

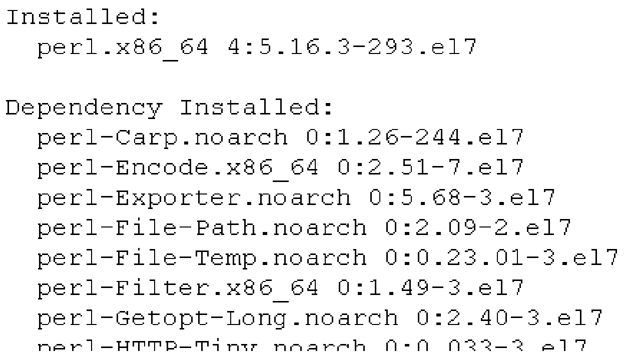
**yum install -y gcc gcc-c++**



```
Installed:
  gcc.x86_64 0:4.8.5-36.el7                  gcc-c++.x86_64 0:4.8.5-36.el7

Dependency Installed:
  cpp.x86_64 0:4.8.5-36.el7
  glibc-devel.x86_64 0:2.17-260.el7
  glibc-headers.x86_64 0:2.17-260.el7
  kernel-headers.x86_64 0:3.10.0-957.1.3.el7
  libmpc.x86_64 0:1.0.1-3.el7
  libstdc++-devel.x86_64 0:4.8.5-36.el7
  mpfr.x86_64 0:3.1.1-4.el7

Complete!
[root@localhost ~]#
```

## 2.安装 perl

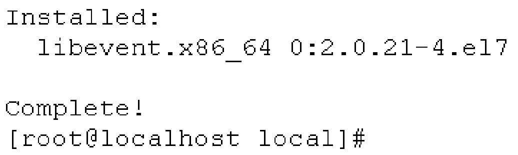
**yum install -y perl**



```
Installed:
  perl.x86_64 4:5.16.3-293.el7

Dependency Installed:
  perl-Carp.noarch 0:1.26-244.el7
  perl-Encode.x86_64 0:2.51-7.el7
  perl-Exporter.noarch 0:5.68-3.el7
  perl-File-Path.noarch 0:2.09-2.el7
  perl-File-Temp.noarch 0:0.23.01-3.el7
  perl-Filter.x86_64 0:1.49-3.el7
  perl-Getopt-Long.noarch 0:2.40-3.el7
  perl-HTTP-Tiny.noarch 0:0.033-3.el7
```

## 3.安装 libevent

yum install -y libevent



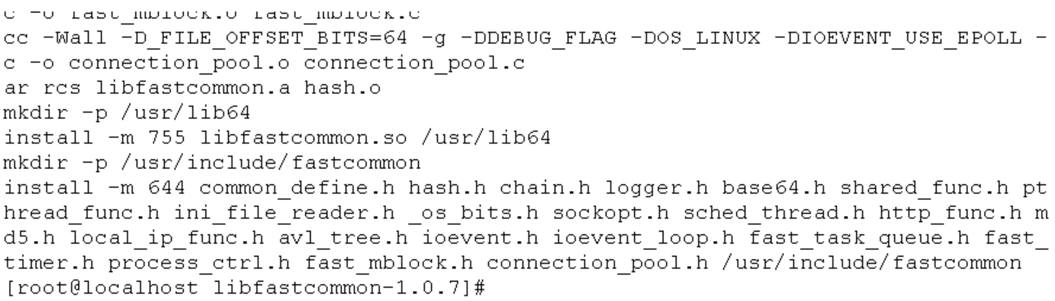
```
Installed:
  libevent.x86_64 0:2.0.21-4.el7

Complete!
[root@localhost local]#
```

## 4.安装 libfastcommon

libfastcommon 没有 yum 源，通过上传的安装包进行编译与安装

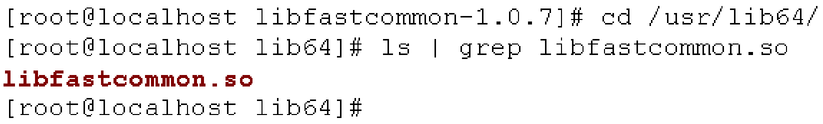
cd /usr/local/libfastcommon-1.0.7/

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



```
c -o fast_mblock.o fast_mblock.c
cc -Wall -D_FILE_OFFSET_BITS=64 -g -DDEBUG_FLAG -DOS_LINUX -DIOEVENT_USE_EPOLL -
c -o connection_pool.o connection_pool.c
ar rcs libfastcommon.a hash.o
mkdir -p /usr/lib64
install -m 755 libfastcommon.so /usr/lib64
mkdir -p /usr/include/fastcommon
install -m 644 common_define.h hash.h chain.h logger.h base64.h shared_func.h pt
hread_func.h ini_file_reader.h _os_bits.h sockopt.h sched_thread.h http_func.h m
d5.h local_ip_func.h avl_tree.h ioevent.h ioevent_loop.h fast_task_queue.h fast_
timer.h process_ctrl.h fast_mblock.h connection_pool.h /usr/include/fastcommon
[root@localhost libfastcommon-1.0.7]#
```

libfastcommon 安装好后会在 /usr/lib64 目录下生成 libfastcommon.so 库文件



```
[root@localhost libfastcommon-1.0.7]# cd /usr/lib64/
[root@localhost lib64]# ls | grep libfastcommon.so
libfastcommon.so
[root@localhost lib64]#
```

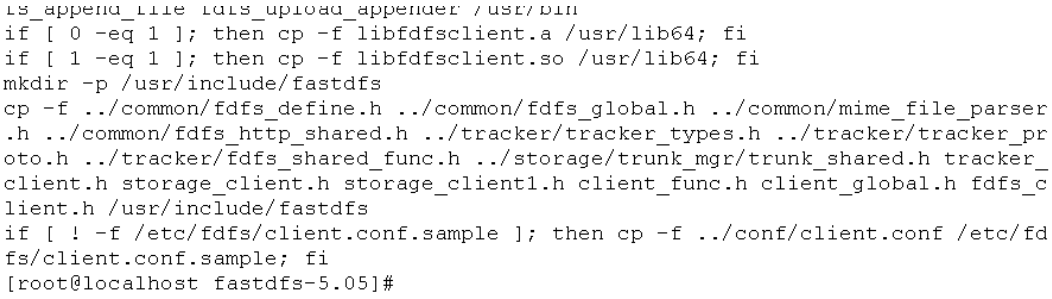
由于 FastDFS 程序引用 usr/lib 目录所以需要将 /usr/lib64 下的库文件拷贝至 /usr/lib 下

cp /usr/lib64/libfastcommon.so /usr/lib

## 5.安装 tracker

cd /usr/local/fastdfs-5.05/

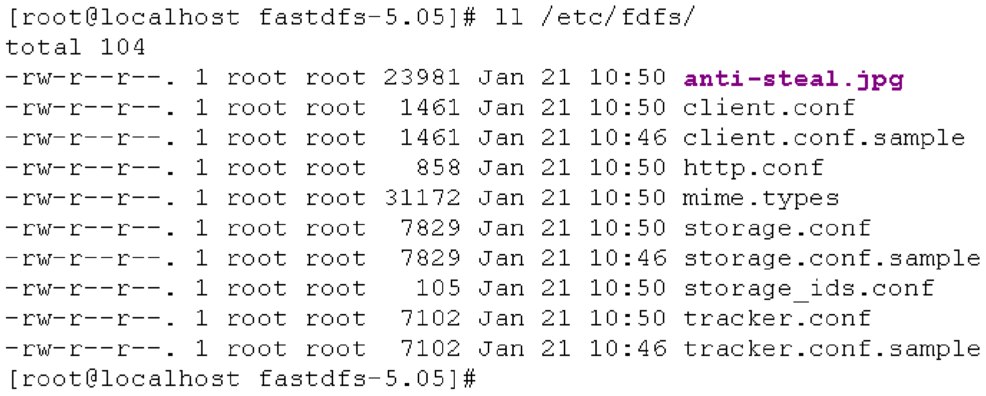
./make.sh && ./make.sh install



```
fs_append_file fdfs_upload_appender /usr/bin
if [ 0 -eq 1 ]; then cp -f libfdfsclient.a /usr/lib64; fi
if [ 1 -eq 1 ]; then cp -f libfdfsclient.so /usr/lib64; fi
mkdir -p /usr/include/fastdfs
cp -f ../common/fdfs_define.h ../common/fdfs_global.h ../common/mime_file_parser
.h ../common/fdfs_http_shared.h ../tracker/tracker_types.h ../tracker/tracker_pr
oto.h ../tracker/fdfs_shared_func.h ../storage/trunk_mgr/trunk_shared.h tracker_
client.h storage_client.h storage_client1.h client_func.h client_global.h fdfs_c
lient.h /usr/include/fastdfs
if [ ! -f /etc/fdfs/client.conf.sample ]; then cp -f ../conf/client.conf /etc/fd
fs/client.conf.sample; fi
[root@localhost fastdfs-5.05]#
```

安装成功将安装目录下的 conf 下的文件拷贝到 /etc/fdfs/ 下

**cp conf/* /etc/fdfs/**



```
[root@localhost fastdfs-5.05]# ll /etc/fdfs/
total 104
-rw-r--r--. 1 root root 23981 Jan 21 10:50 anti-steal.jpg
-rw-r--r--. 1 root root  1461 Jan 21 10:50 client.conf
-rw-r--r--. 1 root root  1461 Jan 21 10:46 client.conf.sample
-rw-r--r--. 1 root root   858 Jan 21 10:50 http.conf
-rw-r--r--. 1 root root 31172 Jan 21 10:50 mime.types
-rw-r--r--. 1 root root  7829 Jan 21 10:50 storage.conf
-rw-r--r--. 1 root root  7829 Jan 21 10:46 storage.conf.sample
-rw-r--r--. 1 root root   105 Jan 21 10:50 storage_ids.conf
-rw-r--r--. 1 root root  7102 Jan 21 10:50 tracker.conf
-rw-r--r--. 1 root root  7102 Jan 21 10:46 tracker.conf.sample
[root@localhost fastdfs-5.05]#
```

## 6.配置和启动 tracker

切换到 /etc/fdfs 目录下

**cd /etc/fdfs**

修改 tracker.conf

**vi tracker.conf**

base_path = /home/fastdfs
http.server_port = 80

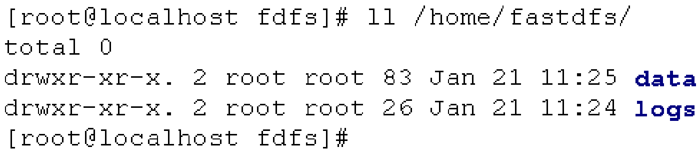
创建 /home/fastdfs 目录

**mkdir -p /home/fastdfs**

启动 tracker

**/usr/bin/fdfs_trackerd /etc/fdfs/tracker.conf restart**

PS：启动成功后，在 /home/fastdfs/ 目录下生成两个目录， 一个是数据，一个是日志



```
[root@localhost fdfs]# ll /home/fastdfs/
total 0
drwxr-xr-x. 2 root root 83 Jan 21 11:25 data
drwxr-xr-x. 2 root root 26 Jan 21 11:24 logs
[root@localhost fdfs]#
```

## 7.配置和启动 storage

切换到 /etc/fdfs 目录下

**cd /etc/fdfs**

修改 tracker.conf

**vi storage.conf**

base_path = /home/fastdfs
store_path0 = /home/fdfs_storage
tracker_server = x.x.x.x:22122 (根据实际情况改)
http.server_port = 88

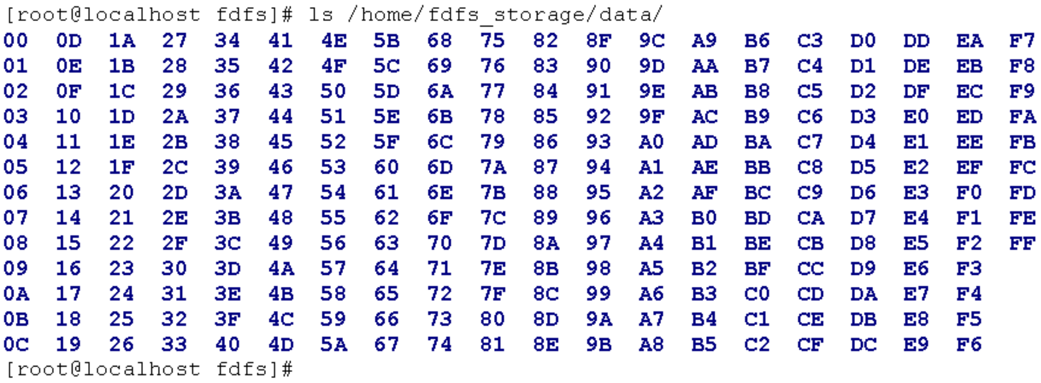
创建 /home/fdfs_storage 目录

**mkdir -p /home/fdfs_storage**

启动 storage

**/usr/bin/fdfs_storaged /etc/fdfs/storage.conf restart**

PS：启动成功后，在 /home/fdfs_storage/data 目录下生成以下显示目录



```
[root@localhost fdfs]# ls /home/fdfs_storage/data/
00  0D  1A  27  34  41  4E  5B  68  75  82  8F  9C  A9  B6  C3  D0  DD  EA  F7
01  0E  1B  28  35  42  4F  5C  69  76  83  90  9D  AA  B7  C4  D1  DE  EB  F8
02  0F  1C  29  36  43  50  5D  6A  77  84  91  9E  AB  B8  C5  D2  DF  EC  F9
03  10  1D  2A  37  44  51  5E  6B  78  85  92  9F  AC  B9  C6  D3  E0  ED  FA
04  11  1E  2B  38  45  52  5F  6C  79  86  93  A0  AD  BA  C7  D4  E1  EE  FB
05  12  1F  2C  39  46  53  60  6D  7A  87  94  A1  AE  BB  C8  D5  E2  EF  FC
06  13  20  2D  3A  47  54  61  6E  7B  88  95  A2  AF  BC  C9  D6  E3  F0  FD
07  14  21  2E  3B  48  55  62  6F  7C  89  96  A3  B0  BD  CA  D7  E4  F1  FE
08  15  22  2F  3C  49  56  63  70  7D  8A  97  A4  B1  BE  CB  D8  E5  F2  FF
09  16  23  30  3D  4A  57  64  71  7E  8B  98  A5  B2  BF  CC  D9  E6  F3
0A  17  24  31  3E  4B  58  65  72  7F  8C  99  A6  B3  C0  CD  DA  E7  F4
0B  18  25  32  3F  4C  59  66  73  80  8D  9A  A7  B4  C1  CE  DB  E8  F5
0C  19  26  33  40  4D  5A  67  74  81  8E  9B  A8  B5  C2  CF  DC  E9  F6
[root@localhost fdfs]#
```

## 8.测试FastDFS上传文件

切换到 /etc/fdfs 目录下

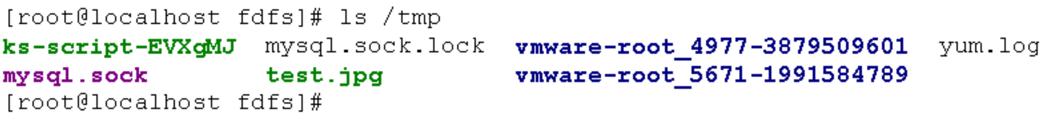
**cd /etc/fdfs**

修改 client.conf

**vi client.conf**

base_path = /home/fastdfs
tracker_server = x.x.x.x:22122 (根据实际情况改)

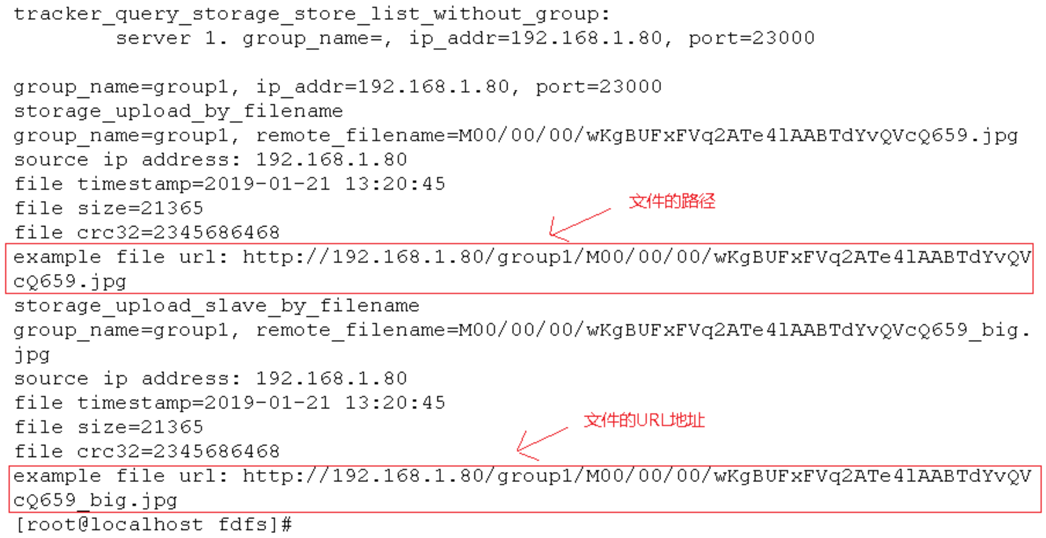
拷贝一张图片 test.jpg 到Centos服务器上的 /tmp 目录下



```
[root@localhost fdfs]# ls /tmp
ks-script-EVXgMJ  mysql.sock.lock  vmware-root_4977-3879509601  yum.log
mysql.sock        test.jpg         vmware-root_5671-1991584789
[root@localhost fdfs]#
```

进行上传测试

**/usr/bin/fdfs_test /etc/fdfs/client.conf upload /tmp/test.jpg**



```
tracker_query_storage_store_list_without_group:
        server 1. group_name=, ip_addr=192.168.1.80, port=23000

group_name=group1, ip_addr=192.168.1.80, port=23000
storage_upload_by_filename
group_name=group1, remote_filename=M00/00/00/wKgBUFxFVq2ATe41AABTdYvQVcQ659.jpg
source ip address: 192.168.1.80
file timestamp=2019-01-21 13:20:45
file size=21365
file crc32=2345686468
example file url: http://192.168.1.80/group1/M00/00/00/wKgBUFxFVq2ATe41AABTdYvQV
cQ659.jpg
storage_upload_slave_by_filename
group_name=group1, remote_filename=M00/00/00/wKgBUFxFVq2ATe41AABTdYvQVcQ659_big.
jpg
source ip address: 192.168.1.80
file timestamp=2019-01-21 13:20:45
file size=21365
file crc32=2345686468
example file url: http://192.168.1.80/group1/M00/00/00/wKgBUFxFVq2ATe41AABTdYvQV
cQ659_big.jpg
[root@localhost fdfs]#
```

PS：由于现在还没有和nginx整合无法使用http下载

## 9.FastDFS 和Nginx整合

切换到 /usr/local/fastdfs-nginx-module/src 目录下

```
cd /usr/local/fastdfs-nginx-module/src/
```

修改config文件

```
vi config
```

将文件中的所有 /usr/local/ 路径改为 /usr/

复制当前目录下的 mod_fastdfs.conf 文件到 /etc/fdfs/ 目录下

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

切换到 /etc/fdfs 目录，修改 mod_fastdfs.conf 文件

```
vi mod_fastdfs.conf
```

```
base_path=/home/fastdfs
tracker_server=x.x.x.x:22122(根据实际情况改)
url_have_group_name=true #url中包含group名称
store_path0=/home/fdfs_storage #指定文件存储路径（上面配置的store路径）
```

将 /usr/lib64 目录下的 libfdfsclient.so 文件拷贝至 /usr/lib 目录下

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

## 10.Nginx 的安装

安装nginx的依赖库：

```
yum install -y pcre pcre-devel
yum install -y zlib zlib-devel
yum install -y openssl openssl-devel
```

创建 /var/temp/nginx/client 目录

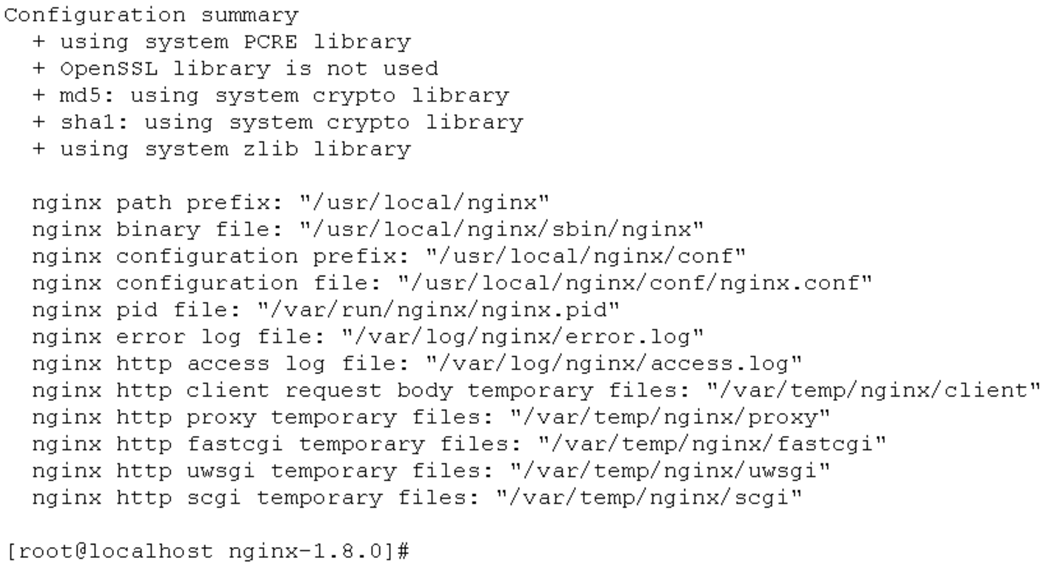
```
mkdir -p /var/temp/nginx/client
```

切换到 nginx 解压的目录

```
cd /usr/local/nginx-1.8.0/
```

执行以下配置命令：

```
./configure \
--prefix=/usr/local/nginx \
--pid-path=/var/run/nginx/nginx.pid \
--lock-path=/var/lock/nginx.lock \
--error-log-path=/var/log/nginx/error.log \
--http-log-path=/var/log/nginx/access.log \
--with-http_gzip_static_module \
--http-client-body-temp-path=/var/temp/nginx/client \
--http-proxy-temp-path=/var/temp/nginx/proxy \
--http-fastcgi-temp-path=/var/temp/nginx/fastcgi \
--http-uwsgi-temp-path=/var/temp/nginx/uwsgi \
--http-scgi-temp-path=/var/temp/nginx/scgi \
--add-module=/usr/local/fastdfs-nginx-module/src
```

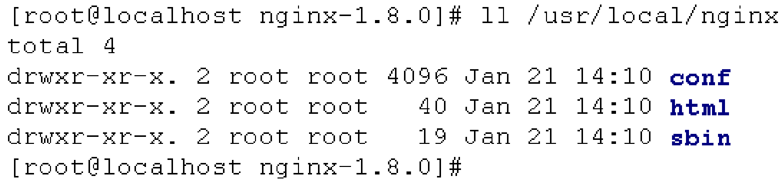


```
Configuration summary
  + using system PCRE library
  + OpenSSL library is not used
  + md5: using system crypto library
  + sha1: using system crypto library
  + using system zlib library

  nginx path prefix: "/usr/local/nginx"
  nginx binary file: "/usr/local/nginx/sbin/nginx"
  nginx configuration prefix: "/usr/local/nginx/conf"
  nginx configuration file: "/usr/local/nginx/conf/nginx.conf"
  nginx pid file: "/var/run/nginx/nginx.pid"
  nginx error log file: "/var/log/nginx/error.log"
  nginx http access log file: "/var/log/nginx/access.log"
  nginx http client request body temporary files: "/var/temp/nginx/client"
  nginx http proxy temporary files: "/var/temp/nginx/proxy"
  nginx http fastcgi temporary files: "/var/temp/nginx/fastcgi"
  nginx http uwsgi temporary files: "/var/temp/nginx/uwsgi"
  nginx http scgi temporary files: "/var/temp/nginx/scgi"

[root@localhost nginx-1.8.0]#
```

编译并安装

**make && make install**

PS:安装成功后查看生成的目录



```
[root@localhost nginx-1.8.0]# ll /usr/local/nginx
total 4
drwxr-xr-x. 2 root root 4096 Jan 21 14:10 conf
drwxr-xr-x. 2 root root   40 Jan 21 14:10 html
drwxr-xr-x. 2 root root   19 Jan 21 14:10 sbin
[root@localhost nginx-1.8.0]#
```

创建 /usr/local/nginx/logs 目录

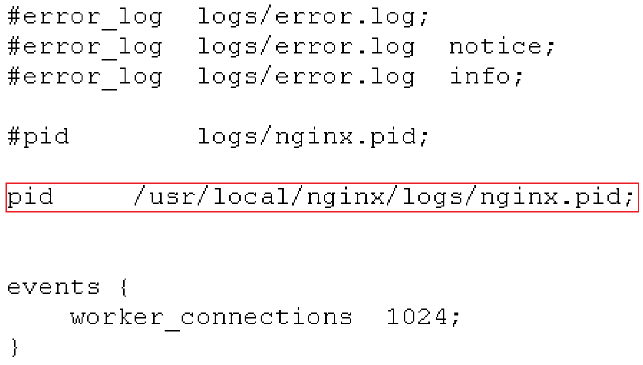
**mkdir /usr/local/nginx/logs**

切换到 /usr/local/nginx/conf/ 目录

**cd /usr/local/nginx/conf/**

修改 nginx.conf 文件

**vi nginx.conf**



```
#error_log  logs/error.log;
#error_log  logs/error.log  notice;
#error_log  logs/error.log  info;

#pid        logs/nginx.pid;

pid     /usr/local/nginx/logs/nginx.pid;


events {
    worker_connections  1024;
}
```

```
  server {
      listen       80;
      server_name  192.168.1.80;

      #charset koi8-r;

      #access_log  logs/host.access.log  main;

      location ~/group([0-9]) {
          root /home/fdfs_storage/data;
          ngx_fastdfs_module;
      }

      location / {
          root   html;
          index  index.html index.htm;
      }

      #error_page  404              /404.html;
```

切换到 /usr/local/nginx/sbin/ 目录

**cd /usr/local/nginx/sbin/**

启动 nginx

**./nginx**

```
[root@localhost sbin]# ./nginx
ngx_http_fastdfs_set pid=13057
[root@localhost sbin]#
```

查看 nginx 进程

```
[root@localhost sbin]# ps -ef | grep nginx
root      13058      1  0 14:35 ?        00:00:00 nginx: master process ./nginx
nobody    13059  13058  0 14:35 ?        00:00:00 nginx: worker process
root      13061   7803  0 14:36 pts/2    00:00:00 grep --color=auto nginx
[root@localhost sbin]#
```

放通防火墙80(nginx)、22122(tracker)、23000(storage)端口

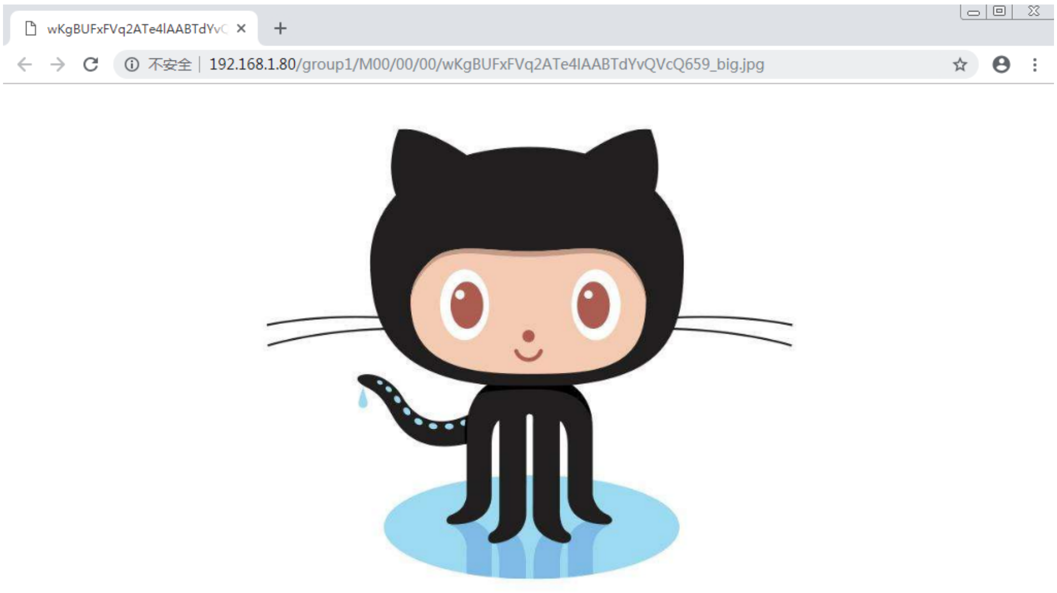
**firewall-cmd --zone=public --add-port=80/tcp --permanent**

**firewall-cmd --zone=public --add-port=22122/tcp --permanent**

**firewall-cmd --zone=public --add-port=23000/tcp --permanent**

**firewall-cmd --reload**

这时已经可以通过浏览器访问刚刚测试上传的图片 test.jpg！


wKgBUFxFVq2ATe4lAABTdYvC
不安全
192.168.1.80/group1/M00/00/00/wKgBUFxFVq2ATe4lAABTdYvQVcQ659_big.jpg

## 11.FastDFS 和 Nginx 开机自启配置

编辑 /etc/rc.d/rc.local 文件，增加启动项

**vi /etc/rc.d/rc.local**

```
# fastdfs start
/usr/bin/fdfs_trackerd /etc/fdfs/tracker.conf restart
/usr/bin/fdfs_storaged /etc/fdfs/storage.conf restart
```

```
# nginx start
/usr/local/nginx/sbin/nginx
```

给rc.local 文件增加可执行的权限

**chmod +x /etc/rc.d/rc.local**