

# Fastdfs 部署手册

内部资料 请勿外传

## 文档属性

客户名称		文件类别	文档
文件名	Fastdfs部署手册	是否保密	否
编制者	赵天旭	编制者职位	售后工程师
编制者邮箱	zhaotianxu@kylinos.cn	编制日期	2021-8-19
版本修订记录			
版本号	修订时间	修订说明	
V1.0	2021-8-19	新建	

一、 文档说明.....	4
二、 背景.....	4
三、 方案.....	4

内部参考 请勿外传

## 一、文档说明

在 V10-server-0711-arm64 上安装 Fastdfs,解决 FASTDFS 安装使用问题,V10 的源并未提供 fastdfs,采用源码编译的方式安装

## 二、背景

FastDFS 是一个开源的高性能分布式文件系统 (DFS)。它的主要功能包括:文件存储,文件同步和文件访问,以及高容量和负载平衡。主要解决了海量数据存储问题,特别适合以中小文件(建议范围:4KB < file\_size < 500MB)为载体的在线服务。

FastDFS 系统有三个角色:跟踪服务器(Tracker Server)、存储服务器(Storage Server)和客户端(Client)。

**Tracker Server:** 跟踪服务器,主要做调度工作,起到均衡的作用;负责管理所有的 storage server 和 group,每个 storage 在启动后会连接 Tracker,告知自己所属 group 等信息,并保持周期性心跳。

**Storage Server:** 存储服务器,主要提供容量和备份服务;以 group 为单位,每个 group 内可以有多个 storage server,数据互为备份。

**Client:** 客户端,上传下载数据的服务器,也就是我们自己的项目所部署在的服务器。

## 三、方案

### 1. 编译

#### (1) 编译环境配置

```
yum -y install gcc gcc-c++ libstdc++-devel
```

源码包准备：

（安装 FastDFS 必须先安装 libfastcommon 类库，否则会导致报错）

```
wget https://github.com/happyfish100/libfastcommon/archive/master.zip
```

下载 FastDFS V5.11 版本：

```
wget https://github.com/happyfish100/fastdfs/archive/V5.11.zip
```

## （2）解压 libfastcommon 源码包

```
# unzip master.zip
```

进入 libfastcommon-master 目录

```
# cd libfastcommon-master
```

执行编译

```
# ./make.sh
```

执行安装

```
# ./make.sh install
```

## （3）解压 FastDFS V5.11 版本

```
# unzip V5.11.zip
```

进入 fastdfs-5.11 目录

```
# cd fastdfs-5.11
```

执行编译

```
# ./make.sh
```

## 执行安装

```
# ./make.sh install
```

## 2.环境搭建及测试

### (1) 配置 Tracker

#### 1).创建 tracker 数据文件和日志目录

```
#mkdir /home/kylin/tracker （按需求自定义）
```

重命名 sample 文件或者是复制 sample 配置文件，

```
#cp /etc/fdfs/tracker.conf.sample /etc/fdfs/tracker.conf
```

并编辑 tracker.conf 完成参数配置

```
#sudo vi /etc/fdfs/tracker.conf
```

```
disabled=false # 启用配置文件
```

```
port=22122 # 设置 tracker 的端口号
```

```
base_path=/home/kylin/tracker # 设置 tracker 的数据文件和日志目录（需手动创建）
```

```
http.server_port=8080 # 设置 http 端口号
```

```
thread_stack_size = 128KB #需要>=64KB
```

#### 2).启动 tracker

```
#/usr/bin/fdfs_trackerd /etc/fdfs/tracker.conf restart
```

查看端口 22122 是否开始监听，确认启动是否成功

```
#ps -ef|grep fdfs
```

```
#netstat -nulp |grep fdfs
```

也可以查看 tracker 的日志是否启动成功

```
#cat /home/kylin/tracker/logs/trackerd.log
```

## (2)配置 storage

### 1).创建 storage 数据文件和日志目录

```
#mkdir /home/kylin/storage （按需求自定义）
```

重命名 sample 文件或者是复制 sample 配置文件，

```
#cp /etc/fdfs/storage.conf.sample /etc/fdfs/storage.conf
```

并编辑 storage.conf 完成参数配置

```
disabled=false # 启用配置文件
```

```
group_name=group1
```

```
port=23000 # 设置 storage 的端口号
```

```
base_path=/home/kylin/storage # 设置 storage 的数据文件和日志目录（需手动创建）
```

```
store_path_count=1 #存储路径个数，需要和 store_path 个数匹配
```

```
store_path0=/home/kylin/storage #存储路径
```

```
tracker_server=192.168.154.135:22122 #tracker 服务器 IP 地址（根据实际情况修改）和端口
```

号

```
http.server_port=8080 # 设置 http 端口号
```

```
thread_stack_size = 128KB #需要>=64KB
```

### 2).启动 storage

```
#/usr/bin/fdfs_storaged /etc/fdfs/storage.conf restart
```

查看端口 23000 是否开始监听，确认启动是否成功

```
#ps -ef|grep fdfs
```

```
#netstat -nulp |grep fdfs
```

也可以查看 tracker 的日志是否启动成功

```
#cat /home/kylin/storage/logs/storage.log
```

### (3) 配置 client

创建 client 数据文件和日志目录

```
#mkdir /home/kylin/client （按需求自定义）
```

重命名 sample 文件或者是复制 sample 配置文件，

```
#cp /etc/fdfs/client.conf.sample /etc/fdfs/client.conf
```

编辑 client.conf 的文件内容：

```
base_path=/home/kylin/client #之前新建的 client 目录
```

```
tracker_server=192.168.154.135:22122 #tracker 的服务器 IP 地址
```

```
http.tracker_server_port=8080 #tracker 的 http 服务器端口
```

### (4) 文件上传测试

```
#/usr/bin/fdfs_upload_file /etc/fdfs/client.conf master.zip(要上传的文件名)
```

```
[root@localhost fastdfs]# /usr/bin/fdfs_upload_file /etc/fdfs/client.conf master.zip
group1/M00/00/00/wKh6WmEbWx2AcSV0AABX3L0q8UU115.zip
[root@localhost fastdfs]#
```

根据提示，文件将上传到 storage 服务器的 storage/data/00/00 目录，该目录由 storage 启动时自动创建

```
#ls storage/data/00/00/
```

```
[root@localhost fastdfs]# ls storage/data/00/00/
wKh6WmEbWx2AcSV0AABX3L0q8UU115.zip
[root@localhost fastdfs]#
```



文件上传成功，Fastdfs 可正常使用

内部资料 请勿外传