FastDFS:
开源的轻量级分布式文件系统，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~500MB)为载体的在线服务。
组件：
跟踪服务器(tracker server)
存储服务器(storage server)
客户端(client)
存储服务器:
以组或卷(group/volume)为单位组织，一个组内包含多台存储服务器，数据互为备份
注：存储空间以组内容量最小的存储为准，所以建议同组内的存储服务器尽量配置相同，以免造成存储空间的浪费。
功能：
应用隔离：将不同应用数据存到不同的组中
负载均衡：将应用分配到不同的组中
副本数定制：同组内存储服务器的数量即为副本数
缺点：
组的容量受单机存储容量的限制
当组内有机器坏掉时，数据恢复只能依赖组内其他机器，恢复时间长
跟踪服务器：
FastDFS 的协调者
负责管理所有的 storage server 和 group
每个 storage 在启动后会连接 Tracker，告知自己所属的 group 等信息，并保持周期性的心跳
tracker 根据 storage 的心跳信息，建立 group-->[storage server list]的映射表
Tracker 需要管理的元信息很少，会全部存储在内存中
tracker 上的元信息都是由 storage 汇报的信息生成的，本身不需要持久化任何数据
多台 tracker 可组成集群，集群中每个 tracker 之间是完全对等的，所有的 tracker 都接受 stroage 的心跳信息和读写请求
上传文件：tracker.conf
1.客户端在上传文件时任意选择一个 trakcer。当 tracker 接收到上传文件的请求时，会为该文件分配一个可以存储该文件的 group
选择 group 的规则：store_lookup
0：Round robin：所有的 group 间轮询
1：Specified group：指定某一个确定的 group
2:Load balance：剩余存储空间多多 group 优先
2.当选定 group 后，tracker 会在 group 内选择一个 storage server 给客户端
选择 storage 的规则：store_server
0:Round robin：在 group 内的所有 storage 间轮询
1:First server ordered by ip：按 ip 排序
2:First server ordered by priority：按优先级排序(优先级在 storage 上配置)
3.分配好 storage server 后，客户端将向 storage 发送写文件请求，storage 将会为文件分配一个数据存储目录
分配数据存储目录规则：store_path
0:Round robin：多个存储目录间轮询
2:剩余存储空间最多的优先

4.选定存储目录后，将文件路由到一个二级目录文件，存储到该目录下的某个子目录后，即认为该文件存储成功，会为该文件生成一个文件名
文件名命名规则：group、存储目录、两级子目录、fileid、文件后缀名(由客户端指定，主要用于区分文件类型)拼接而成
文件同步：
客户端将文件写入 group 内任一个 storage server 即认为写文件成功，之后会由后台线程将文件同步至同 group 内其他的 storage server
每个 storage 写文件后，同时会写一份 binlog，binlog 里不包含文件数据，只包含文件名等元信息，这份 binlog 用于后台同步，storage 会记录向 group 内其他 storage 同步的进度，以便重启后能继续上次的进度继续同步，进度以时间戳的方式进行记录。storage 的同步进度会作为元数据的一部分汇报到 tracker 上，tracke 在选择读 storage 的时候会参考同步进度。
下载文件：
1.tracker 发送 download 请求给某个 tracker，必须带上文件名信息
2.tracke 从文件名中解析出文件的 group、大小、创建时间等信息
3.为该请求选择一个 storage 用来服务读请求。
注：由于 group 内的文件同步在后台异步进行的，所以有可能出现在读的时候，文件还没有同步到某些 storage server 上，为了尽量避免访问到没完成的同步，tracker 按照一定规则选择 group 内可读的 storage。
规则：
1.源头 storage 只要存活着，肯定包含这个文件，源头的地址被编码在文件名中。
2.文件创建时间戳==storage 被同步到的时间戳且(当前时间-文件创建时间戳) > 文件同步最大时间(如 5 分钟)。文件创建后，认为经过最大同步时间后，肯定已经同步到其他 storage 了。
3.文件创建时间戳 < storage 被同步到的时间戳。同步时间戳之前的文件确定已经同步了
4.(当前时间-文件创建时间戳) > 同步延迟阀值（如一天）。超过同步延迟阈值时间，认为文件肯定已经同步了。
搭建 FastDFS：
下载网址：
FastDFS：https://github.com/happyfish100/fastdfs/tree/master
libfastcommon：https://github.com/happyfish100/libfastcommon
注：目录下都有一个 INSTALL，是安装说明
1.安装 libfastcommon：包含了 FastDFS 运行所需要的一些基础库
[root@client ~]# yum -y install gcc gcc-c++ libevent-devel git
[root@client ~]# git clone https://github.com/happyfish100/libfastcommon.git
[root@client ~]# cd libfastcommon
[root@client libfastcommon]# git checkout V1.0.38
[root@client libfastcommon]# ./make.sh
[root@client libfastcommon]# ./make.sh install
2.安装 FastDFS
[root@client ~]# git clone https://github.com/happyfish100/fastdfs.git
[root@client ~]# cd fastdfs/
[root@client fastdfs]# git checkout V5.11
[root@client fastdfs]# ./make.sh
[root@client fastdfs]# ./make.sh install

3.修改跟踪服务器配置文件

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

[root@client ~]# vim /etc/fdfs/tracker.conf

11：port=22122

22：base_path=/fastdfs

49：store_lookup=2

54：store_group=group2

60：store_server=0

65：store_path=0

260：http.server_port=80

4.启动跟踪服务

[root@client ~]# mkdir /fastdfs

[root@client ~]# fdfs_trackerd /etc/fdfs/tracker.conf start

或

[root@client ~]# service fdfs_trackerd start

注：

早期版本的 libfastcommon 将 libfastcommon.so 放在/lib，后来移到了/lib64。可以通过 ldd /usr/bin/fdfs_trackerd 查看。

如果安装过早期版本，只需将/local 下的动态库文件删除，指向位置就会自动更新

[root@client ~]# rm -rf /usr/local/lib/libfastcommon.so.1

5./fastdfs 目录说明

data：存数据

logs：存日志

6.修改存储服务器配置文件

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

[root@client ~]# vim /etc/fdfs/storage.conf

11：group_name=group1　　#定义组名，默认为 group1

41：base_path=/fastdfs

109：store_path0=/fastdfs

118：tracker_server=172.16.186.96:22122

284：http.server_port=8888

注：

单机测试 IP 为 172.16.186.96

如果有多个挂载磁盘则定义多个 store_path，例如 110 行

有多个跟踪服务器则配置多行 tracker，一行只能指定一个跟踪服务器

tracker_server 不允许指定为 127.0.0.1

7.启动存储服务

[root@client ~]# fdfs_storaged /etc/fdfs/storage.conf start

或

[root@client ~]# service fdfs_storaged start

8.使用 FastDFS 自带 client 测试

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

[root@client ~]# vim /etc/fdfs/client.conf

10:base_path=/fastdfs

14:tracker_server=172.16.186.96:22122
[root@client ~]# fdfs_test /etc/fdfs/client.conf upload test.txt
group_name=group1, remote_filename=M00/00/00/rBC6YFvbxKSAfRNaAAAABelEJbQ726.txt
example file url: http://172.16.186.96/group1/M00/00/00/rBC6YFvbxKSAfRNaAAAABelEJbQ726.txt
example file url: http://172.16.186.96/group1/M00/00/00/rBC6YFvbxKSAfRNaAAAABelEJbQ726_big.txt
FastDFS +nginx：需要完全前面 7 步
下载地址：https://github.com/happyfish100/fastdfs-nginx-module
1.下载
[root@client ~]# git clone https://github.com/happyfish100/fastdfs-nginx-module.git
[root@client ~]# cd fastdfs-nginx-module/
[root@client fastdfs-nginx-module]# git checkout V1.20
2.修改配置文件
[root@client fastdfs-nginx-module]# cd src/
[root@client src]# vim mod_fastdfs.conf
10：base_path=/fastdfs
40：tracker_server=172.16.186.96:22122
47：group_name=group1 #同 storage.conf
53：url_have_group_name = true #URL 中是否包含 group 名
62：store_path0=/fastdfs #指定文件存储路径，同 storage.conf
[root@client src]# cp mod_fastdfs.conf /etc/fdfs/
3.移动目录，Nginx 需要 src 目录下其他几个文件
[root@client ~]# cp -r fastdfs-nginx-module /usr/local/
[root@client ~]# vim /usr/local/fastdfs-nginx-module/src/config
6：ngx_module_incs="/usr/include/fastdfs /usr/include/fastcommon"
15：CORE_INCS="$CORE_INCS /usr/include/fastdfs /usr/include/fastcommon"
4.安装 Nginx
[root@client ~]# mkdir -p /temp/nginx/{client,proxy,fastcgi,uwsgi,scgi}
[root@client ~]# yum -y install pcre-devel zlib-devel
[root@client nginx-1.15.5]# ./configure --user=nginx --group=nginx --add-module=/usr/local/fastdfs-nginx-module/src
--http-client-body-temp-path=/temp/nginx/client --http-proxy-temp-path=/temp/nginx/proxy
--http-fastcgi-temp-path=/temp/nginx/fastcgi --http-uwsgi-temp-path=/temp/nginx/uwsgi
--http-scgi-temp-path=/temp/nginx/scgi
[root@client nginx-1.15.5]# make && make install
5.拷贝 FastDFS 的配置文件到/etc/fdfs
[root@client ~]# cp ~/fastdfs/conf/http.conf ~/fastdfs/conf/mime.types /etc/fdfs/
6.修改 Nginx 的配置文件
注：group1 为组名。M00 对应 storage.conf 的 store_path0 字段
[root@client ~]# vim /usr/local/nginx/conf/nginx.conf
http {
server {
server_name 172.16.186.96;

```
...
location /group1/M00/ {
        root /fastdfs/data/;
        ngx_fastdfs_module;
}
...
}}
```

7.启动 Nginx

```
[root@client ~]# /usr/local/nginx/sbin/nginx
```