<http://bbs.chinaunix.net/thread-1941456-1-1.html>

**首先是 tracker.conf**

# is this config file disabled

# false for enabled

# true for disabled

disabled=false

# 这个配置文件是否不生效,呵呵(改成是否生效是不是会让人感觉好点呢?) false 为生效(否则不生效) true反之

# bind an address of this host

# empty for bind all addresses of this host

bind\_addr=

# 是否绑定IP,

# bind\_addr= 后面为绑定的IP地址 (常用于服务器有多个IP但只希望一个IP提供服务)。如果不填则表示所有的(一般不填就OK),相信较熟练的SA都常用到类似功能,很多系统和应用都有

# the tracker server port

port=22122

# 提供服务的端口,不作过多解释了

# connect timeout in seconds

# default value is 30s

connect\_timeout=30

#连接超时时间，针对socket套接字函数connect

# network timeout in seconds

network\_timeout=60

# tracker server的网络超时，单位为秒。发送或接收数据时，如果在超时时间后还不能发送或接收数据，则本次网络通信失败。

# the base path to store data and log files

base\_path=/home/yuqing/fastdfs

# base\_path 目录地址(根目录必须存在,子目录会自动创建)

# 附目录说明:

tracker server目录及文件结构：

${base\_path}

|\_\_data

| |\_\_storage\_groups.dat：存储分组信息

| |\_\_storage\_servers.dat：存储服务器列表

|\_\_logs

|\_\_trackerd.log：tracker server日志文件

数据文件storage\_groups.dat和storage\_servers.dat中的记录之间以换行符（\n）分隔，字段之间以西文逗号（,）分隔。

storage\_groups.dat中的字段依次为：

1. group\_name：组名

2. storage\_port：storage server端口号

storage\_servers.dat中记录storage server相关信息，字段依次为：

1. group\_name：所属组名

2. ip\_addr：ip地址

3. status：状态

4. sync\_src\_ip\_addr：向该storage server同步已有数据文件的源服务器

5. sync\_until\_timestamp：同步已有数据文件的截至时间（UNIX时间戳）

6. stat.total\_upload\_count：上传文件次数

7. stat.success\_upload\_count：成功上传文件次数

8. stat.total\_set\_meta\_count：更改meta data次数

9. stat.success\_set\_meta\_count：成功更改meta data次数

10. stat.total\_delete\_count：删除文件次数

11. stat.success\_delete\_count：成功删除文件次数

12. stat.total\_download\_count：下载文件次数

13. stat.success\_download\_count：成功下载文件次数

14. stat.total\_get\_meta\_count：获取meta data次数

15. stat.success\_get\_meta\_count：成功获取meta data次数

16. stat.last\_source\_update：最近一次源头更新时间（更新操作来自客户端）

17. stat.last\_sync\_update：最近一次同步更新时间（更新操作来自其他storage server的同步）

# max concurrent connections this server supported

# max\_connections worker threads start when this service startup

max\_connections=256

# 系统提供服务时的最大连接数。对于V1.x，因一个连接由一个线程服务，也就是工作线程数。

# 对于V2.x，最大连接数和工作线程数没有任何关系

# work thread count, should <= max\_connections

# default value is 4

# since V2.00

# V2.0引入的这个参数，工作线程数，通常设置为CPU数

work\_threads=4

# the method of selecting group to upload files

# 0: round robin

# 1: specify group

# 2: load balance, select the max free space group to upload file

store\_lookup=2

# 上传组(卷) 的方式 0:轮询方式 1: 指定组 2: 平衡负载(选择最大剩余空间的组(卷)上传)

# 这里如果在应用层指定了上传到一个固定组,那么这个参数被绕过

# which group to upload file

# when store\_lookup set to 1, must set store\_group to the group name

store\_group=group2

# 当上一个参数设定为1 时 (store\_lookup=1，即指定组名时)，必须设置本参数为系统中存在的一个组名。如果选择其他的上传方式，这个参数就没有效了。

# which storage server to upload file

# 0: round robin (default)

# 1: the first server order by ip address

# 2: the first server order by priority (the minimal)

store\_server=0

# 选择哪个storage server 进行上传操作(一个文件被上传后，这个storage server就相当于这个文件的storage server源，会对同组的storage server推送这个文件达到同步效果)

# 0: 轮询方式

# 1: 根据ip 地址进行排序选择第一个服务器（IP地址最小者）

# 2: 根据优先级进行排序（上传优先级由storage server来设置，参数名为upload\_priority）

# which path(means disk or mount point) of the storage server to upload file

# 0: round robin

# 2: load balance, select the max free space path to upload file

store\_path=0

# 选择storage server 中的哪个目录进行上传。storage server可以有多个存放文件的base path（可以理解为多个磁盘）。

# 0: 轮流方式，多个目录依次存放文件

# 2: 选择剩余空间最大的目录存放文件（注意：剩余磁盘空间是动态的，因此存储到的目录或磁盘可能也是变化的）

# which storage server to download file

# 0: round robin (default)

# 1: the source storage server which the current file uploaded to

download\_server=0

# 选择哪个 storage server 作为下载服务器

# 0: 轮询方式，可以下载当前文件的任一storage server

# 1: 哪个为源storage server 就用哪一个 (前面说过了这个storage server源 是怎样产生的) 就是之前上传到哪个storage server服务器就是哪个了

# reserved storage space for system or other applications.

# if the free(available) space of any stoarge server in

# a group <= reserved\_storage\_space,

# no file can be uploaded to this group.

# bytes unit can be one of follows:

### G or g for gigabyte(GB)

### M or m for megabyte(MB)

### K or k for kilobyte(KB)

### no unit for byte(B)

### XX.XX% as ratio such as reserved\_storage\_space = 10%

reserved\_storage\_space = 10%

# storage server 上保留的空间，保证系统或其他应用需求空间。可以用绝对值或者百分比（V4开始支持百分比方式）。

#(指出 如果同组的服务器的硬盘大小一样,以最小的为准,也就是只要同组中有一台服务器达到这个标准了,这个标准就生效,原因就是因为他们进行备份)

#standard log level as syslog, case insensitive, value list:

### emerg for emergency

### alert

### crit for critical

### error

### warn for warning

### notice

### info

### debug

log\_level=info

# 选择日志级别(日志写在哪?看前面的说明了,有目录介绍哦 呵呵)

#unix group name to run this program,

#not set (empty) means run by the group of current user

run\_by\_group=

# 操作系统运行FastDFS的用户组 (不填 就是当前用户组,哪个启动进程就是哪个)

#unix username to run this program,

#not set (empty) means run by current user

run\_by\_user=

# 操作系统运行FastDFS的用户 (不填 就是当前用户,哪个启动进程就是哪个)

# allow\_hosts can ocur more than once, host can be hostname or ip address,

# "\*" means match all ip addresses, can use range like this: 10.0.1.[1-15,20] or

# host[01-08,20-25].domain.com, for example:

# allow\_hosts=10.0.1.[1-15,20]

# allow\_hosts=host[01-08,20-25].domain.com

allow\_hosts=\*

# 可以连接到此 tracker server 的ip范围（对所有类型的连接都有影响，包括客户端，storage server）

# sync log buff to disk every interval seconds

# default value is 10 seconds

sync\_log\_buff\_interval = 10

# 同步或刷新日志信息到硬盘的时间间隔，单位为秒

# 注意：tracker server 的日志不是时时写硬盘的，而是先写内存。

# check storage server alive interval

check\_active\_interval = 120

# 检测 storage server 存活的时间隔，单位为秒。

# storage server定期向tracker server 发心跳，如果tracker server在一个check\_active\_interval内还没有收到storage server的一次心跳，那边将认为该storage server已经下线。所以本参数值必须大于storage server配置的心跳时间间隔。通常配置为storage server心跳时间间隔的2倍或3倍。

# thread stack size, should > 512KB

# default value is 1MB

thread\_stack\_size=1MB

# 线程栈的大小。FastDFS server端采用了线程方式。更正一下，tracker server线程栈不应小于64KB，不是512KB。

# 线程栈越大，一个线程占用的系统资源就越多。如果要启动更多的线程（V1.x对应的参数为max\_connections，

V2.0为work\_threads），可以适当降低本参数值。

# auto adjust when the ip address of the storage server changed

# default value is true

storage\_ip\_changed\_auto\_adjust=true

# 这个参数控制当storage server IP地址改变时，集群是否自动调整。注：只有在storage server进程重启时才完成自动调整。

# storage sync file max delay seconds

# default value is 86400 seconds (one day)

# since V2.00

storage\_sync\_file\_max\_delay = 86400

# V2.0引入的参数。存储服务器之间同步文件的最大延迟时间，缺省为1天。根据实际情况进行调整

# 注：本参数并不影响文件同步过程。本参数仅在下载文件时，判断文件是否已经被同步完成的一个阀值（经验值）

# the max time of storage sync a file

# default value is 300 seconds

# since V2.00

storage\_sync\_file\_max\_time = 300

# V2.0引入的参数。存储服务器同步一个文件需要消耗的最大时间，缺省为300s，即5分钟。

# 注：本参数并不影响文件同步过程。本参数仅在下载文件时，作为判断当前文件是否被同步完成的一个阀值（经验值）

# if use a trunk file to store several small files

# default value is false

# since V3.00

use\_trunk\_file = false

# V3.0引入的参数。是否使用小文件合并存储特性，缺省是关闭的。

# the min slot size, should <= 4KB

# default value is 256 bytes

# since V3.00

slot\_min\_size = 256

# V3.0引入的参数。

# trunk file分配的最小字节数。比如文件只有16个字节，系统也会分配slot\_min\_size个字节。

# the max slot size, should > slot\_min\_size

# store the upload file to trunk file when it's size <= this value

# default value is 16MB

# since V3.00

slot\_max\_size = 16MB

# V3.0引入的参数。

# 只有文件大小<=这个参数值的文件，才会合并存储。如果一个文件的大小大于这个参数值，将直接保存到一个文件中（即不采用合并存储方式）。

# the trunk file size, should >= 4MB

# default value is 64MB

# since V3.00

trunk\_file\_size = 64MB

# V3.0引入的参数。

# 合并存储的trunk file大小，至少4MB，缺省值是64MB。不建议设置得过大。

# if create trunk file advancely

# default value is false

trunk\_create\_file\_advance = false

# 是否提前创建trunk file。只有当这个参数为true，下面3个以trunk\_create\_file\_打头的参数才有效。

# the time base to create trunk file

# the time format: HH:MM

# default value is 02:00

trunk\_create\_file\_time\_base = 02:00

# 提前创建trunk file的起始时间点（基准时间），02:00表示第一次创建的时间点是凌晨2点。

# the interval of create trunk file, unit: second

# default value is 38400 (one day)

trunk\_create\_file\_interval = 86400

# 创建trunk file的时间间隔，单位为秒。如果每天只提前创建一次，则设置为86400

# the threshold to create trunk file

# when the free trunk file size less than the threshold, will create

# the trunk files

# default value is 0

trunk\_create\_file\_space\_threshold = 20G

# 提前创建trunk file时，需要达到的空闲trunk大小

# 比如本参数为20G，而当前空闲trunk为4GB，那么只需要创建16GB的trunk file即可。

# if check trunk space occupying when loading trunk free spaces

# the occupied spaces will be ignored

# default value is false

# since V3.09

# NOTICE: set this parameter to true will slow the loading of trunk spaces

# when startup. you should set this parameter to true when neccessary.

trunk\_init\_check\_occupying = false

#trunk初始化时，是否检查可用空间是否被占用

# if ignore storage\_trunk.dat, reload from trunk binlog

# default value is false

# since V3.10

# set to true once for version upgrade when your version less than V3.10

trunk\_init\_reload\_from\_binlog = false

# 是否无条件从trunk binlog中加载trunk可用空间信息

# FastDFS缺省是从快照文件storage\_trunk.dat中加载trunk可用空间，

# 该文件的第一行记录的是trunk binlog的offset，然后从binlog的offset开始加载

# if use storage ID instead of IP address

# default value is false

# since V4.00

use\_storage\_id = false

# 是否使用server ID作为storage server标识

# specify storage ids filename, can use relative or absolute path

# since V4.00

storage\_ids\_filename = storage\_ids.conf

# use\_storage\_id 设置为true，才需要设置本参数

# 在文件中设置组名、server ID和对应的IP地址，参见源码目录下的配置示例：conf/storage\_ids.conf

# if store slave file use symbol link

# default value is false

# since V4.01

store\_slave\_file\_use\_link = false

# 存储从文件是否采用symbol link（符号链接）方式

# 如果设置为true，一个从文件将占用两个文件：原始文件及指向它的符号链接。

# if rotate the error log every day

# default value is false

# since V4.02

rotate\_error\_log = false

# 是否定期轮转error log，目前仅支持一天轮转一次

# rotate error log time base, time format: Hour:Minute

# Hour from 0 to 23, Minute from 0 to 59

# default value is 00:00

# since V4.02

error\_log\_rotate\_time=00:00

# error log定期轮转的时间点，只有当rotate\_error\_log设置为true时有效

# rotate error log when the log file exceeds this size

# 0 means never rotates log file by log file size

# default value is 0

# since V4.02

rotate\_error\_log\_size = 0

# error log按大小轮转

# 设置为0表示不按文件大小轮转，否则当error log达到该大小，就会轮转到新文件中

# 以下是关于http的设置了 默认编译是不生效的 要求更改 #WITH\_HTTPD=1 将 注释#去掉 再编译

# 关于http的应用 说实话 不是很了解 没有见到 相关说明 ,望 版主可以完善一下 以下是字面解释了

#HTTP settings

http.disabled=false # HTTP服务是否不生效

http.server\_port=8080 # HTTP服务端口

#use "#include" directive to include http other settiongs

##include http.conf # 如果加载http.conf的配置文件 去掉第一个#

哈哈 完成了一个 下面是 storage.conf

# is this config file disabled

# false for enabled

# true for disabled

disabled=false

#同上文了 就不多说了

# the name of the group this storage server belongs to

group\_name=group1

# 指定 此 storage server 所在 组(卷)

# bind an address of this host

# empty for bind all addresses of this host

bind\_addr=

# 同上文

# if bind an address of this host when connect to other servers

# (this storage server as a client)

# true for binding the address configed by above parameter: "bind\_addr"

# false for binding any address of this host

client\_bind=true

# bind\_addr通常是针对server的。当指定bind\_addr时，本参数才有效。

# 本storage server作为client连接其他服务器（如tracker server、其他storage server），是否绑定bind\_addr。

# the storage server port

port=23000

# storage server服务端口

# connect timeout in seconds

# default value is 30s

connect\_timeout=30

#连接超时时间，针对socket套接字函数connect

# network timeout in seconds

network\_timeout=60

# storage server 网络超时时间，单位为秒。发送或接收数据时，如果在超时时间后还不能发送或接收数据，则本次网络通信失败。

# heart beat interval in seconds

heart\_beat\_interval=30

# 心跳间隔时间，单位为秒 (这里是指主动向tracker server 发送心跳)

# disk usage report interval in seconds

stat\_report\_interval=60

# storage server向tracker server报告磁盘剩余空间的时间间隔，单位为秒。

# the base path to store data and log files

base\_path=/home/yuqing/fastdfs

# base\_path 目录地址,根目录必须存在 子目录会自动生成 (注 :这里不是上传的文件存放的地址,之前是的,在某个版本后更改了)

# 目录结构 因为 版主没有更新到 论谈上 这里就不发了 大家可以看一下置顶贴:

# max concurrent connections server supported

# max\_connections worker threads start when this service startup

max\_connections=256

# 同上文

# work thread count, should <= max\_connections

# default value is 4

# since V2.00

# V2.0引入的这个参数，工作线程数，通常设置为CPU数

work\_threads=4

# the buff size to recv / send data

# default value is 64KB

# since V2.00

buff\_size = 256KB

# V2.0引入本参数。设置队列结点的buffer大小。工作队列消耗的内存大小 = buff\_size \* max\_connections

# 设置得大一些，系统整体性能会有所提升。

# 消耗的内存请不要超过系统物理内存大小。另外，对于32位系统，请注意使用到的内存不要超过3GB

# if read / write file directly

# if set to true, open file will add the O\_DIRECT flag to avoid file caching

# by the file system. be careful to set this parameter.

# default value is false

disk\_rw\_direct = false

# V2.09引入本参数。设置为true，表示不使用操作系统的文件内容缓冲特性。

# 如果文件数量很多，且访问很分散，可以考虑将本参数设置为true

# if disk read / write separated

## false for mixed read and write

## true for separated read and write

# default value is true

# since V2.00

disk\_rw\_separated = true

# V2.0引入本参数。磁盘IO读写是否分离，缺省是分离的。

# disk reader thread count per store base path

# for mixed read / write, this parameter can be 0

# default value is 1

# since V2.00

disk\_reader\_threads = 1

# V2.0引入本参数。针对单个存储路径的读线程数，缺省值为1。

# 读写分离时，系统中的读线程数 = disk\_reader\_threads \* store\_path\_count

# 读写混合时，系统中的读写线程数 = (disk\_reader\_threads + disk\_writer\_threads) \* store\_path\_count

# disk writer thread count per store base path

# for mixed read / write, this parameter can be 0

# default value is 1

# since V2.00

disk\_writer\_threads = 1

# V2.0引入本参数。针对单个存储路径的写线程数，缺省值为1。

# 读写分离时，系统中的写线程数 = disk\_writer\_threads \* store\_path\_count

# 读写混合时，系统中的读写线程数 = (disk\_reader\_threads + disk\_writer\_threads) \* store\_path\_count

# when no entry to sync, try read binlog again after X milliseconds

# 0 for try again immediately (not need to wait)

sync\_wait\_msec=200

# 同步文件时，如果从binlog中没有读到要同步的文件，休眠N毫秒后重新读取。0表示不休眠，立即再次尝试读取。

# 出于CPU消耗考虑，不建议设置为0。如何希望同步尽可能快一些，可以将本参数设置得小一些，比如设置为10ms

# after sync a file, usleep milliseconds

# 0 for sync successively (never call usleep)

sync\_interval=0

# 同步上一个文件后，再同步下一个文件的时间间隔，单位为毫秒，0表示不休眠，直接同步下一个文件。

# sync start time of a day, time format: Hour:Minute

# Hour from 0 to 23, Minute from 0 to 59

sync\_start\_time=00:00

# sync end time of a day, time format: Hour:Minute

# Hour from 0 to 23, Minute from 0 to 59

sync\_end\_time=23:59

# 上面二个一起解释。允许系统同步的时间段 (默认是全天) 。一般用于避免高峰同步产生一些问题而设定，相信sa都会明白

# write to the mark file after sync N files

# default value is 500

write\_mark\_file\_freq=500

# 同步完N个文件后，把storage的mark文件同步到磁盘

# 注：如果mark文件内容没有变化，则不会同步

# path(disk or mount point) count, default value is 1

store\_path\_count=1

# 存放文件时storage server支持多个路径（例如磁盘）。这里配置存放文件的基路径数目，通常只配一个目录。

# store\_path#, based 0, if store\_path0 not exists, it's value is base\_path

# the paths must be exist

store\_path0=/home/yuqing/fastdfs

#store\_path1=/home/yuqing/fastdfs2

# 逐一配置store\_path个路径，索引号基于0。注意配置方法后面有0,1,2 ......，需要配置0到store\_path - 1。

# 如果不配置base\_path0，那边它就和base\_path对应的路径一样。

# subdir\_count \* subdir\_count directories will be auto created under each

# store\_path (disk), value can be 1 to 256, default value is 256

subdir\_count\_per\_path=256

# FastDFS存储文件时，采用了两级目录。这里配置存放文件的目录个数 (系统的存储机制,大家看看文件存储的目录就知道了)

# 如果本参数只为N（如：256），那么storage server在初次运行时，会自动创建 N \* N 个存放文件的子目录。

# tracker\_server can ocur more than once, and tracker\_server format is

# "host:port", host can be hostname or ip address

tracker\_server=10.62.164.84:22122

tracker\_server=10.62.245.170:22122

# tracker\_server 的列表 要写端口的哦 (再次提醒是主动连接tracker\_server )

# 有多个tracker server时，每个tracker server写一行

#standard log level as syslog, case insensitive, value list:

### emerg for emergency

### alert

### crit for critical

### error

### warn for warning

### notice

### info

### debug

log\_level=info

# 日志级别不多说

#unix group name to run this program,

#not set (empty) means run by the group of current user

run\_by\_group=

# 同上文了

#unix username to run this program,

#not set (empty) means run by current user

run\_by\_user=

# 同上文了 (提醒注意权限 如果和 webserver不搭 可以会产生错误 哦)

# allow\_hosts can ocur more than once, host can be hostname or ip address,

# "\*" means match all ip addresses, can use range like this: 10.0.1.[1-15,20] or

# host[01-08,20-25].domain.com, for example:

# allow\_hosts=10.0.1.[1-15,20]

# allow\_hosts=host[01-08,20-25].domain.com

allow\_hosts=\*

# 允许连接本storage server的IP地址列表 （不包括自带HTTP服务的所有连接）

# 可以配置多行，每行都会起作用

# the mode of the files distributed to the data path

# 0: round robin(default)

# 1: random, distributted by hash code

file\_distribute\_path\_mode=0

# 文件在data目录下分散存储策略。

# 0: 轮流存放，在一个目录下存储设置的文件数后（参数file\_distribute\_rotate\_count中设置文件数），使用下一个目录进行存储。

# 1: 随机存储，根据文件名对应的hash code来分散存储。

# valid when file\_distribute\_to\_path is set to 0 (round robin),

# when the written file count reaches this number, then rotate to next path

# default value is 100

file\_distribute\_rotate\_count=100

# 当上面的参数file\_distribute\_path\_mode配置为0（轮流存放方式）时，本参数有效。

# 当一个目录下的文件存放的文件数达到本参数值时，后续上传的文件存储到下一个目录中。

# call fsync to disk when write big file

# 0: never call fsync

# other: call fsync when written bytes >= this bytes

# default value is 0 (never call fsync)

fsync\_after\_written\_bytes=0

# 当写入大文件时，每写入N个字节，调用一次系统函数fsync将内容强行同步到硬盘。0表示从不调用fsync

# sync log buff to disk every interval seconds

# default value is 10 seconds

sync\_log\_buff\_interval=10

# 同步或刷新日志信息到硬盘的时间间隔，单位为秒

# 注意：storage server 的日志信息不是时时写硬盘的，而是先写内存。

# sync binlog buff / cache to disk every interval seconds

# this parameter is valid when write\_to\_binlog set to 1

# default value is 60 seconds

sync\_binlog\_buff\_interval=60

# 同步binglog（更新操作日志）到硬盘的时间间隔，单位为秒

# 本参数会影响新上传文件同步延迟时间

# sync storage stat info to disk every interval seconds

# default value is 300 seconds

sync\_stat\_file\_interval=300

# 把storage的stat文件同步到磁盘的时间间隔，单位为秒。

# 注：如果stat文件内容没有变化，不会进行同步

# thread stack size, should >= 512KB

# default value is 512KB

thread\_stack\_size=512KB

# 线程栈的大小。FastDFS server端采用了线程方式。

# 对于V1.x，storage server线程栈不应小于512KB；对于V2.0，线程栈大于等于128KB即可。

# 线程栈越大，一个线程占用的系统资源就越多。

# 对于V1.x，如果要启动更多的线程（max\_connections），可以适当降低本参数值。

# the priority as a source server for uploading file.

# the lower this value, the higher its uploading priority.

# default value is 10

upload\_priority=10

# 本storage server作为源服务器，上传文件的优先级，可以为负数。值越小，优先级越高。这里就和 tracker.conf 中store\_server= 2时的配置相对应了

# if check file duplicate, when set to true, use FastDHT to store file indexes

# 1 or yes: need check

# 0 or no: do not check

# default value is 0

check\_file\_duplicate=0

# 是否检测上传文件已经存在。如果已经存在，则不存在文件内容，建立一个符号链接以节省磁盘空间。

# 这个应用要配合FastDHT 使用，所以打开前要先安装FastDHT

# 1或yes 是检测，0或no 是不检测

# file signature method for check file duplicate

## hash: four 32 bits hash code

## md5: MD5 signature

# default value is hash

# since V4.01

file\_signature\_method=hash

# 文件去重时，文件内容的签名方式：

## hash： 4个hash code

## md5：MD5

# namespace for storing file indexes (key-value pairs)

# this item must be set when check\_file\_duplicate is true / on

key\_namespace=FastDFS

# 当上个参数设定为1 或 yes时 (true/on也是可以的) ， 在FastDHT中的命名空间。

# set keep\_alive to 1 to enable persistent connection with FastDHT servers

# default value is 0 (short connection)

keep\_alive=0

# 与FastDHT servers 的连接方式 (是否为持久连接) ，默认是0（短连接方式）。可以考虑使用长连接，这要看FastDHT server的连接数是否够用。

# 下面是关于FastDHT servers 的设定 需要对FastDHT servers 有所了解,这里只说字面意思了

# you can use "#include filename" (not include double quotes) directive to

# load FastDHT server list, when the filename is a relative path such as

# pure filename, the base path is the base path of current/this config file.

# must set FastDHT server list when check\_file\_duplicate is true / on

# please see INSTALL of FastDHT for detail

##include /home/yuqing/fastdht/conf/fdht\_servers.conf

# 可以通过 #include filename 方式来加载 FastDHT servers 的配置，装上FastDHT就知道该如何配置啦。

# 同样要求 check\_file\_duplicate=1 时才有用，不然系统会忽略

# fdht\_servers.conf 记载的是 FastDHT servers 列表

# if log to access log

# default value is false

# since V4.00

use\_access\_log = false

# 是否将文件操作记录到access log

# if rotate the access log every day

# default value is false

# since V4.00

rotate\_access\_log = false

# 是否定期轮转access log，目前仅支持一天轮转一次

# rotate access log time base, time format: Hour:Minute

# Hour from 0 to 23, Minute from 0 to 59

# default value is 00:00

# since V4.00

access\_log\_rotate\_time=00:00

# access log定期轮转的时间点，只有当rotate\_access\_log设置为true时有效

# if rotate the error log every day

# default value is false

# since V4.02

rotate\_error\_log = false

# 是否定期轮转error log，目前仅支持一天轮转一次

# rotate error log time base, time format: Hour:Minute

# Hour from 0 to 23, Minute from 0 to 59

# default value is 00:00

# since V4.02

error\_log\_rotate\_time=00:00

# error log定期轮转的时间点，只有当rotate\_error\_log设置为true时有效

# rotate access log when the log file exceeds this size

# 0 means never rotates log file by log file size

# default value is 0

# since V4.02

rotate\_access\_log\_size = 0

# access log按文件大小轮转

# 设置为0表示不按文件大小轮转，否则当access log达到该大小，就会轮转到新文件中

# rotate error log when the log file exceeds this size

# 0 means never rotates log file by log file size

# default value is 0

# since V4.02

rotate\_error\_log\_size = 0

# error log按文件大小轮转

# 设置为0表示不按文件大小轮转，否则当error log达到该大小，就会轮转到新文件中

# if skip the invalid record when sync file

# default value is false

# since V4.02

file\_sync\_skip\_invalid\_record=false

# 文件同步的时候，是否忽略无效的binlog记录

下面是http的配置了。如果系统较大，这个服务有可能支持不了，可以自行换一个webserver，我喜欢lighttpd，当然ng也很好了。具体不说明了。相应这一块的说明大家都懂，不明白见上文。

#HTTP settings

http.disabled=false

# the port of the web server on this storage server

http.server\_port=8888

http.trunk\_size=256KB

# http.trunk\_size表示读取文件内容的buffer大小（一次读取的文件内容大小），也就是回复给HTTP client的块大小。

# use the ip address of this storage server if domain\_name is empty,

# else this domain name will ocur in the url redirected by the tracker server

http.domain\_name=

# storage server上web server域名，通常仅针对单独部署的web server。这样URL中就可以通过域名方式来访问storage server上的文件了，

# 这个参数为空就是IP地址的方式。

#use "#include" directive to include HTTP other settiongs

##include http.conf

**补充：**

storage.conf中影响storage server同步速度的参数有如下几个：  
# when no entry to sync, try read binlog again after X milliseconds  
# 0 for try again immediately (not need to wait)  
sync\_wait\_msec=200  
# 同步文件时，如果从binlog中没有读到要同步的文件，休眠N毫秒后重新读取。0表示不休眠，立即再次尝试读取。  
# 不建议设置为0，如何希望同步尽可能快一些，可以将本参数设置得小一些，比如设置为10ms  
  
# after sync a file, usleep milliseconds  
# 0 for sync successively (never call usleep)  
sync\_interval=0  
# 同步上一个文件后，再同步下一个文件的时间间隔，单位为毫秒，0表示不休眠，直接同步下一个文件。  
  
# sync start time of a day, time format: Hour:Minute  
# Hour from 0 to 23, Minute from 0 to 59  
sync\_start\_time=00:00  
  
# sync end time of a day, time format: Hour:Minute  
# Hour from 0 to 23, Minute from 0 to 59  
sync\_end\_time=23:59  
# 上面二个一起解释。允许系统同步的时间段 (默认是全天) 。一般用于避免高峰同步产生一些问题而设定，相信sa都会明白  
  
# sync binlog buff / cache to disk every interval seconds  
# this parameter is valid when write\_to\_binlog set to 1  
# default value is 60 seconds  
sync\_binlog\_buff\_interval=60  
# 同步binglog（更新操作日志）到硬盘的时间间隔，单位为秒  
# 本参数会影响新上传文件同步延迟时间