视频在线APP部署手册

V1.0

一、服务器配置需求

a. 服务器系统：CentOS release 6.8

b. 服务器硬件指数（最低）：CPU:4核8G 带宽：2M以上 硬盘：500G以上

二、组件安装

1. JDK1.8安装（软件自行下载）

软件：jdk-8u25-linux-x64.rpm

安装JDK

 上传新的jdk-8u25-linux-x64.rpm软件到/usr/local/执行以下操作：

 [root@localhost local]# rpm -ivh jdk-8u25-linux-x64.rpm

 JDK默认安装在/usr/java中。

 验证安装

 执行以下操作，查看信息是否正常：

 [root@localhost ~]# java

 [root@localhost ~]# javac

 [root@localhost ~]# java -version

 java version "1.8.0\_25"

 Java(TM) SE Runtime Environment (build 1.8.0\_25-b17)

 Java HotSpot(TM) 64-Bit Server VM (build 25.25-b02, mixed mode)

 配置环境变量

 我的机器安装完jdk-7-linux-x64.rpm后不用配置环境变量也可以正常执行javac、java –version操作，因此我没有进行JDK环境变量的配置。但是为了以后的不适之需，这里还是记录一下怎么进行配置，操作如下：

 修改系统环境变量文件

 vi + /etc/profile

 向文件里面追加以下内容：

 JAVA\_HOME=/usr/java/jdk1.8.0\_151

 JRE\_HOME=/usr/java/jdk1.8.0\_151/jre

 PATH=$PATH:$JAVA\_HOME/bin:$JRE\_HOME/bin

 CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar:$JRE\_HOME/lib

 export JAVA\_HOME JRE\_HOME PATH CLASSPATH

 使修改生效

 [root@localhost ~]# source /etc/profile   //使修改立即生效

 [root@localhost ~]#        echo $PATH   //查看PATH值

 查看系统环境状态

 [root@localhost ~]# echo $PATH

 /usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin:/root/bi

n:/usr/java/jdk1.8.0\_25/bin:/usr/java/jdk1.8.0\_25/jre/bin

2.安装数据库（软件自行下载）

软件：

MySQL-server-5.5.53-1.linux2.6.x86\_64.rpm

MySQL-client-5.5.53-1.linux2.6.x86\_64.rpm

Centos 系统，可以提前将这些工具包安装上：

 yum -y install gcc libxml2-dev curl screen \libpng12-dev autoconf libpcre3-dev make bzip2 \libevent-dev patch libjpeg62-dev libcurl4-openssl-dev \libfreetype6-dev g++ libtool libncurses5-dev psmisc lrzsz

 卸载 系统原有mysql

执行 yum 命令，删除 MySQL 的 lib 库，服务文件

yum remove mysqlmysql-server mysql-libs

执行 find 命令，查找 MySQL 的残留文件，然后运行“rm -rf 文件名”删除残留

的 MySQL 文件

find / -name mysql

确保删除干净

rpm -qa | grep -imysql

执行完上述命令后，返回空数据，则删除完成。否则，执行下面的命令删除

MySQL 的相关包文件。

yum -y remove mysql-libs\*

安装 mysql-server

rpm -ivh MySQL-server-5.5.53-1.linux2.6.x86\_64.rpm

rpm -ivh MySQL-client-5.5.53-1.linux2.6.x86\_64.rpm

尝试启动服务

/etc/init.d/mysql start

停止服务

/etc/init.d/mysql stop

使用安全模式登陆，跳过密码验证

mysqld\_safe --user=mysql --skip-grant-tables --skip-

networking &

进入 mysql 界面

mysql -u root mysql

修改 登录密码

mysql> UPDATE user SET Password=PASSWORD(“abc123”)

where USER='root';

刷新权限

mysql>FLUSH PRIVILEGES;

mysql>quit;

重启服务

/etc/init.d/mysql restart

密码登录：

mysql -u root -p

Enter password: <newpassword>

授权任何机器连接

mysql>GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%'

IDENTIFIED BY 'abc123' WITH GRANT OPTION;

mysql>FLUSH PRIVILEGES;

mysql>quit;

设置开机启动

chkconfig mysql on

查看是否加入成功

chkconfig --list

此列表中存在 mysql

安装成功后导入数据库脚本，并修改源码中

videoadmin\src\main\resources 和videoweb\src\main\resources

下的jdbc.properties 文件

1. 安装文件服务器Fastdfs

**a.安装编译所需依赖包,trackerd和stroage都需要安装**

yum install make cmake gcc gcc-c++

**b.安装libfastcommon，trackerd和stroage都需要安装**

yum install -y unzip zip

unzip libfastcommon-master.zip

cd libfastcommon-master/

./make.sh

./make.sh install

**c.、trackerd（跟踪器）**

unzip fastdfs-master.zip

cd fastdfs-master/

chmod +x make.sh

./make.sh

./make.sh install

默认脚本安装路径，查看

/etc/init.d是否存在该命令

/etc/init.d/fdfs\_storaged

/etc/init.d/fdfs\_tracker

配置文件在（样例配置文件）,查看

/etc/fdfs/下是否存在文件

/etc/fdfs/client.conf.sample

/etc/fdfs/storage.conf.sample

/etc/fdfs/tracker.conf.sample

命令工具在/usr/bin/目录下的，查看

 ls | grep fdfs

是否存在以下文件

fdfs\_appender\_test

fdfs\_appender\_test1

fdfs\_append\_file

fdfs\_crc32

fdfs\_delete\_file

fdfs\_download\_file

fdfs\_file\_info

fdfs\_monitor

fdfs\_storaged

fdfs\_test

fdfs\_test1

fdfs\_trackerd

fdfs\_upload\_appender

fdfs\_upload\_file

stop.sh

restart.sh

查询以下两个文件中PRG是否改成PRG=/usr/bin 路径下

/etc/init.d/fdfs\_storaged

/etc/init.d/fdfs\_trackerd

配置FastDFS文件

 cd /etc/fdfs/

 cp tracker.conf.sample tracker.conf

 vi /etc/fdfs/tracker.conf

 disabled=false

 port=22122

 base\_path=/home/yuqing/fastdfs

创建数据路径

mkdir -p  /home/yuqing/fastdfs

启动Tracker：

/etc/init.d/fdfs\_trackerd start

（初次成功启动，会在/fastdfs/tracker目录下创建data、logs两个目录）

查看FastDFS Tracker是否已成功启动：

ps -ef | grep fdfs

 关闭Tracker：

/etc/init.d/fdfs\_trackerd stop

 设置FastDFS跟踪器开机启动：

vi /etc/rc.d/rc.local

添加以下内容：

## FastDFS Tracker

/etc/init.d/fdfs\_trackerd start

**d. storage（存储节点）**

cd /etc/fdfs/

cp storage.conf.sample storage.conf

vi /etc/fdfs/storage.conf

#修改配置

disabled=false

#如果服务器部署在一台上，则不需要绑定IP

bind\_addr=192.168.199.114

port=23000

base\_path=/home/yuqing/fastdfs

store\_path0=/home/yuqing/fastdfs

tracker\_server=192.168.199.217:22122

http.server\_port=8888

创建存储文件路径

mkdir -p /home/yuqing/fastdfs

查看防火墙配置

service iptables status

如果防火墙未关闭 ，默认端口23000

vi /etc/sysconfig/iptables

添加如下端口行：

-A INPUT -m state --state NEW -m tcp -p tcp --dport 23000-ACCEPT

重启防火墙：

 service iptables restart

启动storage

/etc/init.d/fdfs\_storaged start

查看FastDFS Storage是否已成功启动

ps -ef | grep fdfs

关闭storage

/etc/init.d/fdfs\_storaged stop

设置FastDFS跟踪器开机启动：

vi /etc/rc.d/rc.local

添加以下内容：

## FastDFS Tracker

/etc/init.d/fdfs\_storaged start

**e.测试上传**

修改Tracker服务器中的客户端配置文件：

vi /etc/fdfs/client.conf

base\_path=/home/yuqing/fastdfs

tracker\_server=192.168.199.217:22122

执行上传命令

/usr/bin/fdfs\_upload\_file  /etc/fdfs/client.conf  /usr/software/news/fastdht-master.zip

返回ID：group1/M00/00/00/wKjHclkaBxmAV9gPAAHENP0mxkk067.zip

说明上传成功。

**f. nginx以及fastdfs-nginx-module-master安装**

fastdfs-nginx-module  安装：在storaged上安装

 unzip fastdfs-nginx-module-master.zip

 cd fastdfs-nginx-module-master/src

安装niginx

tar -zxvf nginx-1.13.0.tar.gz

cd nginx-1.13.0/

./configure --add-module=/usr/software/news/fastdfs-nginx-module-master/src

make && make install

复制fastdfs-nginx-module源码中的配置文件到/etc/fdfs目录，并修改

cp /usr/software/news/fastdfs-nginx-module-master/src/mod\_fastdfs.conf /etc/fdfs/

vi /etc/fdfs/mod\_fastdfs.conf

配置文件修改如下

connect\_timeout=10

base\_path=/tmp

tracker\_server=192.168.199.217:22122

storage\_server\_port=23000

group\_name=group1

url\_have\_group\_name = true

store\_path0=/home/yuqing/fastdfs

**g. nginx以及fastdfs-nginx-module-master安装**

在/home/yuqing/fastdfs文件存储目录下创建软连接,将其链接到实际存放数据的目录

ln -s /home/yuqing/fastdfs/data/ /home/yuqing/fastdfs/data/M00

nginx配置

cd /usr/local/nginx/conf

vi nginx.conf

---------------------------------------------

user  root;

worker\_processes  1;

events {

    worker\_connections  1024;

}

http {

    include       mime.types;

    default\_type  application/octet-stream;

    sendfile        on;

    keepalive\_timeout  65;

    server {

        listen       8888;

        server\_name  localhost;

        location ~/group([0-9])/M00 {

            #alias /fastdfs/storage/data;

            ngx\_fastdfs\_module;

        }

        error\_page   500 502 503 504  /50x.html;

        location = /50x.html {

            root   html;

        }

    }

}

启动Nginx

/usr/local/nginx/sbin/nginx

启动后存在，ngx\_http\_fastdfs\_set pid=23542，则启动成功

如果nginx 不能访问，则去fastdfs/conf下将

 cp http.conf mime.types /etc/fdfs/

**h.FastDHT安装**

解压

unzip fastdht-master.zip

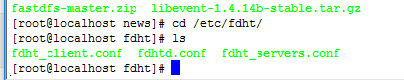
cd fastdht-master

./make.sh clean

./make.sh

./make.sh install

然后再 cd /etc/fdht/生成文件 三个文件



如果缺失文件，就从以下路径去copy过来

cd fastdht-master/conf/

配置FastDHT

cd /etc/fdht/

vim /etc/fdht/fdht\_servers.conf

配置文件新增

group\_count = 1

group0 = 192.168.199.114:11411

配置

vi fdhtd.conf

bind\_addr=192.168.199.114

port=11411

bash\_path=/home/yuqing/fastdht （该目录必须是已经存在的）

cache\_size = 32MB

#include /etc/fdht/fdht\_servers.conf -> (本行前有#表示打开，如果想关闭此选项，则应该为##开头)

配置storaged.conf文件

vim /etc/fdfs/storage.conf

#是否检测上传文件已经存在。如果已经存在，则建立一个索引链接以节省磁盘空间

check\_file\_duplicate=1

#当上个参数设定为1时 ， 在FastDHT中的命名空间

key\_namespace=FastDFS

#长连接配置选项，如果为0则为短连接 1为长连接

keep\_alive=1

在最后一行改成如选，#必须加

#include /etc/fdht/fdht\_servers.conf

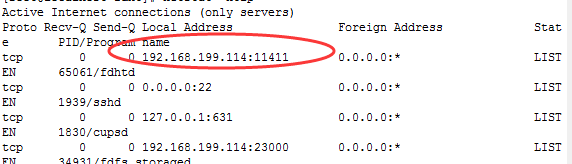
启动DHT

/usr/local/bin/fdhtd /etc/fdht/fdhtd.conf

执行

 netstat -ntlp

看到11411 说明安装成功了



安装成功后修改源码中

videoadmin\src\main\resources 和videoweb\src\main\resources

下的serverurl.properties文件

remote.file.uri.prefix=<http://ip>:8888/

fastDFS.tracker\_server=ip:22122

fastDHT.group0=ip:11411

1. redis安装

去/usr/下 mkdir 新建文件夹 redis

[root@localhost /]# cd /usr/

[root@localhostusr]# mkdir redis

进入 redis 文件目录，wget 下载下来 redis

[root@localhostredis]#wget http://download.redis.io/releases/redis-

3.2.5.tar.gz

解压如下

[root@localhostredis]# tar xzf redis-3.2.5.tar.gz

修改权限如下

[root@localhostusr]# chmod 777 redis/

[root@localhostredis-3.2.5]#make

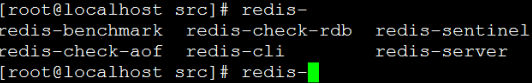
make 编译成功之后，然后使用 make install 安装

root@localhostredis-3.2.5]#make install

[root@localhostredis-3.2.5]#make test

编译完成后，在 src 目录下，有四个可执行文件 redis-server、redis-benchmark、

redis-cli 和 redis.conf，注意：redis.conf 必须跟其他 3 个放在一个路径如下图。



[root@localhostsrc]# cp redis-server /usr/redis/

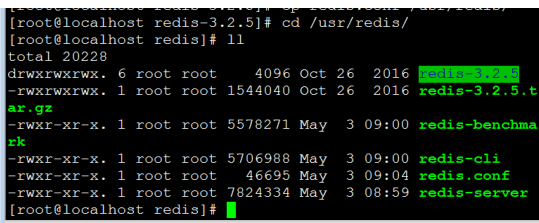
[root@localhostsrc]# cp redis-benchmark /usr/redis/

[root@localhostsrc]# cp redis-cli /usr/redis/

把在/usr/redis/redis-3.2.5 下面的 redis.conf 也拷贝到同上一样的目录。

[root@localhost redis-3.2.5]# cp redis.conf /usr/redis/

去查看下目录是否跟下图一样：



redis.conf 中配置中 涉及的参数 要与下面保持一致：

修改 redis 配置文件设置：

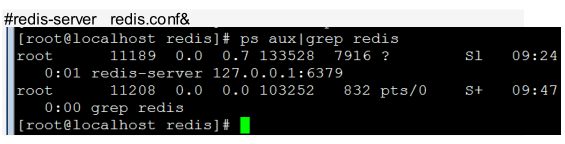
# vi /usr/redis/conf/redis.conf

修改如下配置

daemonize no 改为>daemonize yes

pidfile /var/run/redis.pid 改为>pidfile /var/run/redis\_6379.pid

启动 redis



然后用客户端测试一下 s redis  是否启动成功

#redis-cli

redis> set foo bar

OK

redis> get foo

"bar"

redis  做成后台服务

去 下，把 redis\_init\_script 拷贝到/etc/init.d 下面。

并修改 redis\_init\_script 名称为 redis。

[root@localhostutils]#cp /usr/redis/redis-

3.2.5/utils/redis\_init\_script/etc/init.d/

[root@localhostredis]# cd /etc/init.d/

#mv redis\_init\_scriptredis

[root@localhostinit.d]# pwd

/etc/init.d

并去/usr/redis 下拷贝 redis.conf 为 6379.conf

[root@localhostinit.d]# cd /usr/redis/

[root@localhostredis]#cp redis.conf 6379.conf

并进行修改 redis。

[root@localhostredis]#vi redis

#!/bin/sh

#

# Simple Redisinit.d script conceived to work on Linux systems

# as it does use of the /proc filesystem.

REDISPORT=6379

EXEC=/usr/redis/redis-server

CLIEXEC=/usr/redis/redis-cli

PIDFILE=/var/run/redis\_${REDISPORT}.pid

CONF="/usr/redis/${REDISPORT}.conf"

case "$1" in

start)

if [ -f $PIDFILE ]

then

echo "$PIDFILE exists, process is already running or crashed"

else

echo "Starting Redis server..."

$EXEC $CONF

fi

;;

stop)

if [ ! -f $PIDFILE ]

then

echo "$PIDFILE does not exist, process is not running"

else

PID=$(cat $PIDFILE)

echo "Stopping ..."

$CLIEXEC -p $REDISPORT shutdown

while [ -x /proc/${PID} ]

do

echo "Waiting for Redis to shutdown ..."

sleep 1

done

echo "Redis stopped"

fi

;;

\*)

echo "Please use start or stop as first argument"

;;

esac

6379.conf 涉及要核对的配置如下，请核对附件 6379.conf：

port 6379

daemonize yes

supervised no

启动 redis

[root@localhostredis]#service redis start

查看 redis 进程：ps –ef| grep redis是否存在

1. 应用安装

下载apache-tomcat-7.0.90。并上传到服务器。

然后对videoadmin和videoweb进行打包，把war放在tomcat/webapps下面。

然后启动tomcat.启动步骤省略，自行百度。

注意修改tomcat/conf下的server.xml的端口号和访问路径，尽量不要一样，防止冲突。