项目部署

# 一、项目部署主要分以下几步

1.分配并搭建虚拟机

2.在各个节点中安装jdk，同步时间，关闭防火墙

3.安装FI集群

4.编译项目，上传编译打包项目文件

5.安装RocketMQ

6.安装Elasticsearch

7.数据库建表，ES生成索引

8.根据需要修改项目底下conf目录配置文件

9.服务启动

文档地址：[\\192.168.1.43\组内共享\集群搭建](file:///\\192.168.1.43\组内共享\集群搭建)

# 二、FI集群开发环境搭建

## 1. 分配并搭建虚拟机

三个节点：s109 172.18.18.109；s110 172.18.18.110；s111 72.18.18.111

服务器单节点最低配置：8核、64G内存、500G硬盘

操作系统CentOS-6.7

## 2.在各个节点中安装JDK

1.一般新装虚拟机JDK版本都是1.7，我们的开发环境需要JDK版本为1.8

java -version 查看jdk版本，不是1.8需要卸载：

yum remove java\*

2.下载并安装jdk（选择JDK版本为1.8）：

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

3.上传jdk安装包到/opt/tool 目录下

安装文件获取路径：[\\192.168.1.43\组内共享\集群搭建\安装包](file:///\\192.168.1.43\组内共享\集群搭建\安装包)

4.解压安装包并重命名为jdk

5.设置java环境变量

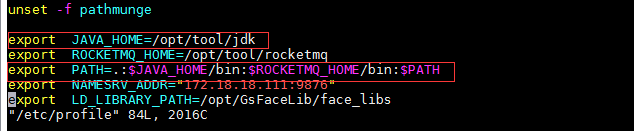
vim /etc/profile

文件最后添加：

export JAVA\_HOME=/opt/tool/jdk

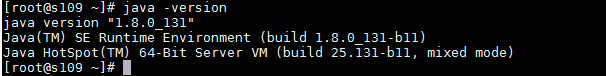
export PATH=.:$JAVA\_HOME/bin:$PATH

source /etc/profile



6.查看是否安装成功

java -version



## 3.FI集群搭建

按照《在Centos系统上安装FI集群文档》搭建FI集群

文档地址：[\\192.168.1.43\组内共享\集群搭建\FI](file:///\\192.168.1.43\组内共享\集群搭建\FI)

# 三、项目编译打包

## 1.开发环境Maven安装与配置

1.下载并安装maven：[http://maven.apache.org/download.cgi#](http://maven.apache.org/download.cgi)

2.设置maven环境变量

3.在D:\apache-maven-3.5.0\conf中替换项目需要的settings.xml文件，文件获取地址：[\\192.168.1.43\组内共享\Maven配置文件](file:///\\192.168.1.43\组内共享\Maven配置文件)

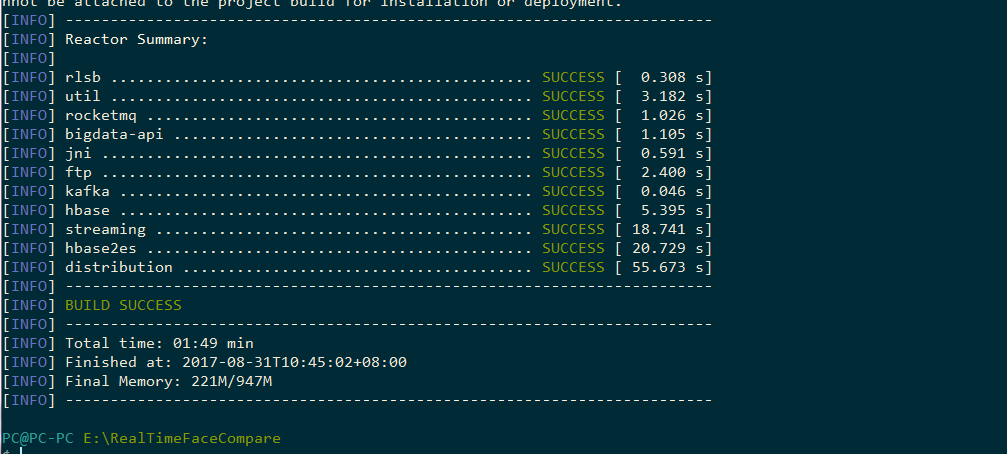
## 2.下载项目源码到本地

在gitHub上下载项目源码：<https://github.com/Zz897426182/RealTimeFaceCompare>

## 3.编译打包

打开项目文件，在空白处Shift+鼠标右键点击“在此处打开命令窗口”：

输入编译打包命名：mvn clean package



打包编译后：

项目打包文件存储路径：

E:\RealTimeFaceCompare\Distribution\target

项目中所有脚本存储路径：

E:\RealTimeFaceCompare\Distribution\target\RealTimeCompare\bin

项目中所有配置文件存储路径：

E:\RealTimeFaceCompare\Distribution\target\RealTimeCompare\conf

项目中所有编译打包后的jar包与依赖jar包存储路径：

E:\RealTimeFaceCompare\Distribution\target\RealTimeCompare\lib

项目中所有建表语句存储路径：

E:\RealTimeFaceCompare\Distribution\target\RealTimeCompare\sql

各个子模块编译后生成的jar包分别存储在各个子模块的\target下

# 四、安装RocketMQ

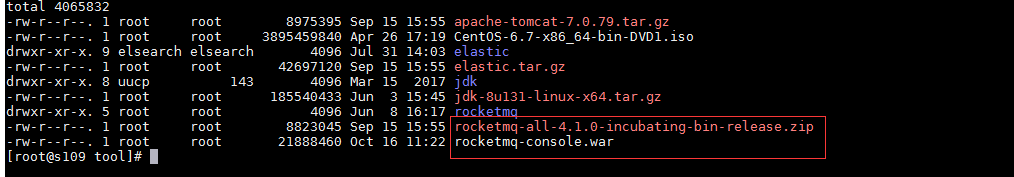
## 1.上传安装文件

上传安装下面2个安装文件到主节点的/opt/tool文件下：

rocketmq-all-4.1.0-incubating-bin-release.zip

rocketmq-console.war

安装文件获取路径：[\\192.168.1.43\组内共享\集群搭建\安装包](file:///\\192.168.1.43\组内共享\集群搭建\安装包)



## 2.解压安装文件，并重命名

unzip rocketmq-all-4.1.0-incubating-bin-release.zip

mv rocketmq-all-4.1.0-incubating-bin-release rocketmq

## 3.规划三个节点，一台安装NameServer，两台Master，没有slave。

NameServer服务器(非主节点)：172.18.18.111（hostname:s111）

两台Master服务器：172.18.18.109（hostname:s109）、172.18.18.110（hostname:s110）

## 4.配置环境变量

vim /etc/profile

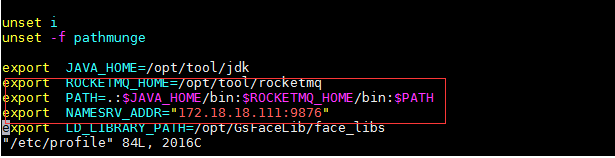
export ROCKETMQ\_HOME=/opt/rocketmq

export PATH=$PATH:$ROCKETMQ\_HOME/bin

export NAMESRV\_ADDR="172.18.18.111:9876"

（NAMESRV\_ADDR配置IP为NameServer服务器IP）

source /etc/profile



## 5.修改文件权限

cd rocketmq/bin/

chmod +x mqadmin mqbroker mqfiltersrv mqshutdown mqnamesrv

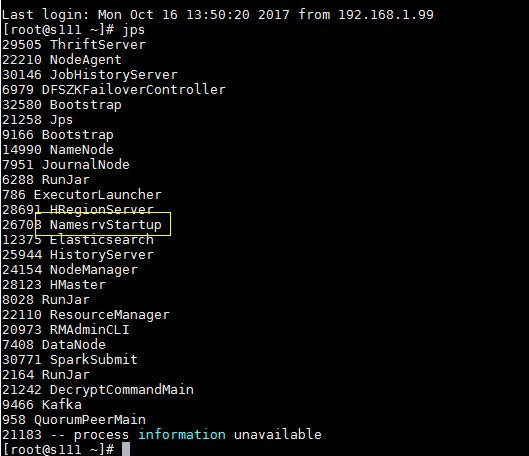
## 6.启动RocketMQ

### 1.启动NameServer

进入 /opt/rocketmq/bin路径下：

nohup ./mqnamesrv–n 172.18.18.111:9876  &

通过jps查看进程中是否启动NameServer



### 2.在两个Master上启动Broker

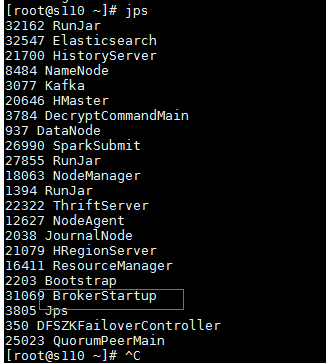
进入/opt/tool/rocketmq/bin路径下

两个Master节点启动Broker注意分别运行不同的命令：

nohup ./mqbroker -n 172.18.18.109:9876-c ../conf/2m-noslave/broker-a.properties &

nohup ./mqbroker -n 172.18.18.110:9876-c ../conf/2m-noslave/broker-b.properties &

通过jps查看两个Master节点进程中是否启动Broker

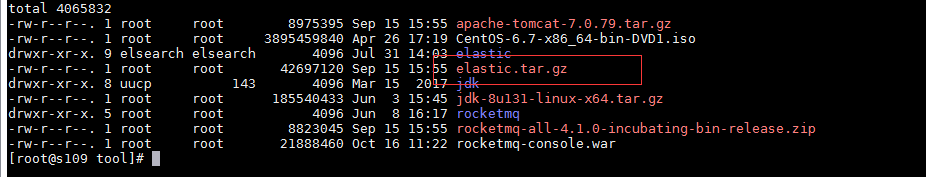


# 五、安装Elasticsearch

## 1.上传安装文件

上传elastic.tar.gz文件到主节点的/opt/tool文件下：

安装文件获取路径：[\\192.168.1.43\组内共享\集群搭建\安装包](file:///\\192.168.1.43\组内共享\集群搭建\安装包)



## 2.解压安装文件

tar -xvf elastic.tar.gz

## 3.修改配置

进入解压后config文件下，修改elasticsearch.yml文件

cd /opt/tool/elastic/config

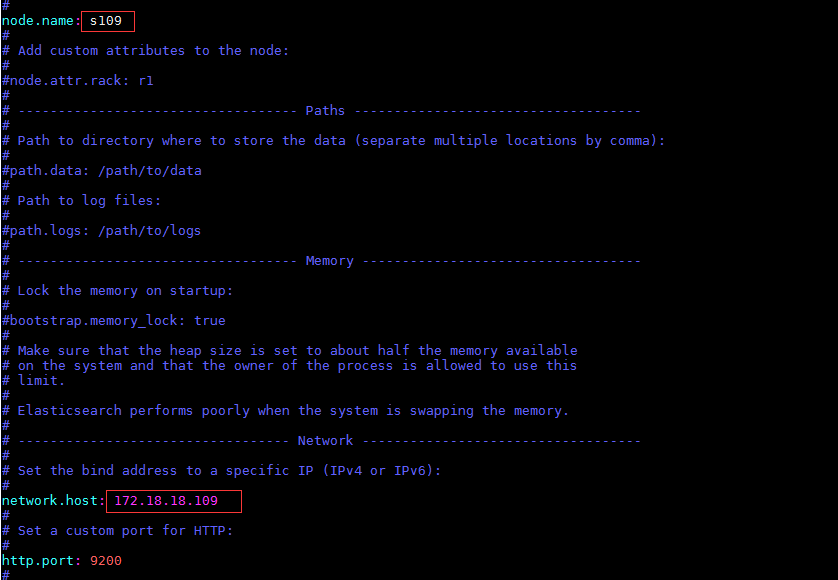
vim elasticsearch.yml

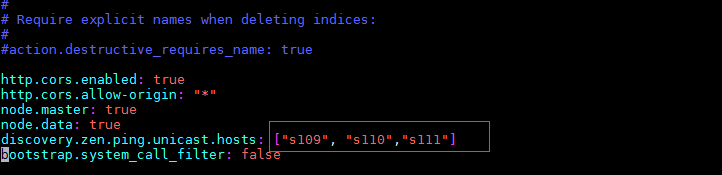
修改下面三处：

node.name: s109

network.host: 172.18.18.109

discovery.zen.ping.unicast.hosts: ["s109", "s110","s111"]





## 4.创建elsearch用户

因为用root用户运行es会报错，所以为了系统安全，需要单独创建一个elsearch用户使用

groupadd elsearch

useradd elsearch –g elsearch –p elastic

chown -R elsearch:elsearch elastic

## 5.启动ES

切换为elsearch用户

su elsearch

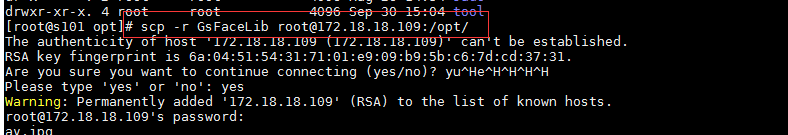
cd elastic/bin

./elasticsearch –d

# 六、配置算法库

## 1.拷贝别的集群中的GsFaceLib到本集群下

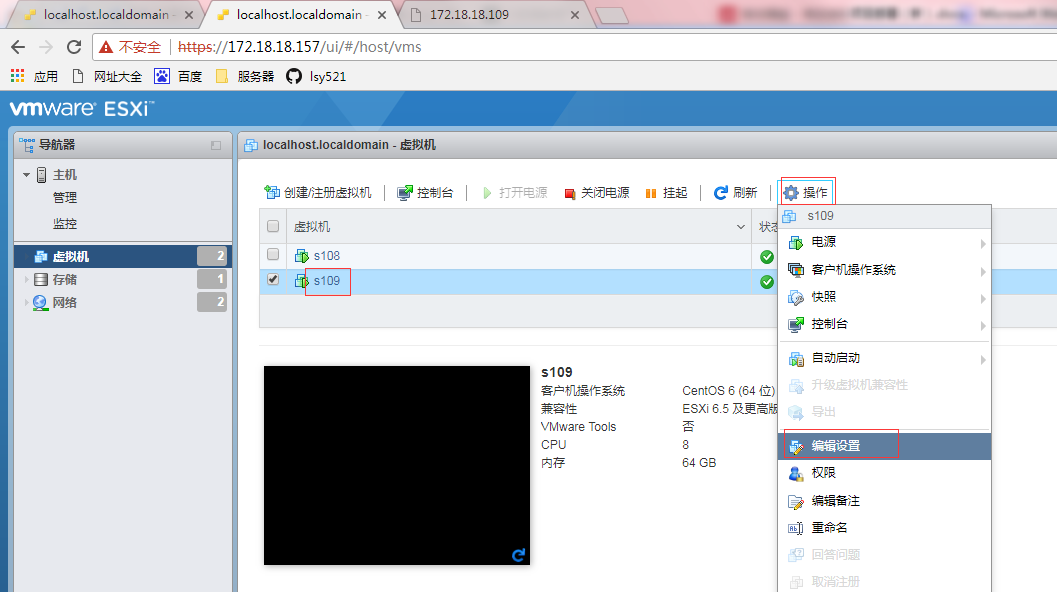
scp -r GsFaceLib root@172.18.18.109:/opt/



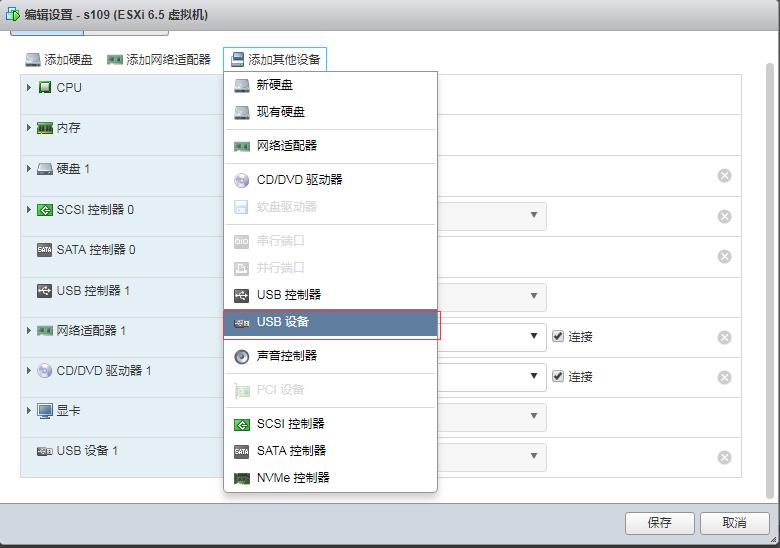
## 2.添加加密狗

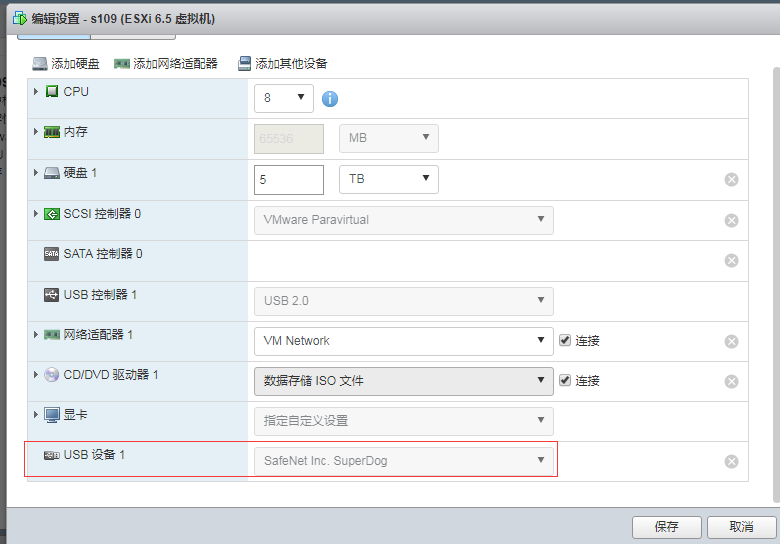
在对应的服务器中添加加密狗

登录对应服务器的vmware，编辑对应虚拟机设置：



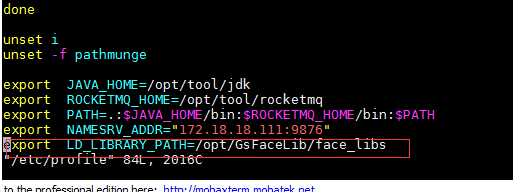
添加USB控制器，后选择添加加密狗：





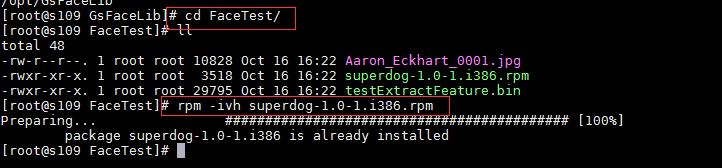
## 3.配置算法库

1.配置/etc/profile



2.进入/opt/GsFaceLib/FaceTest/

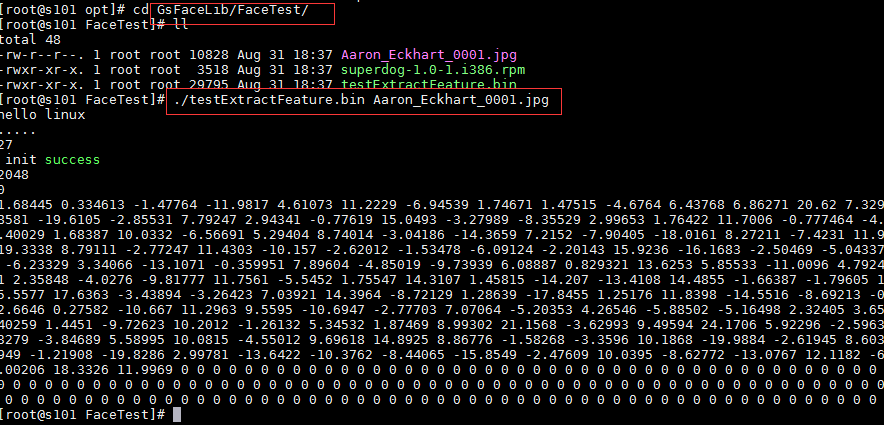
rpm -ivh superdog-1.0-1.i386.rpm



3.删除/opt/GsFaceLib下sn.ini文件

4. 继续进入/opt/GsFaceLib/FaceTest/

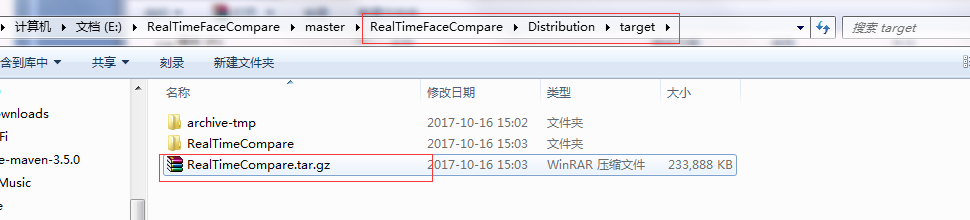
./testExtractFeature.bin Aaron\_Eckhart\_0001.jpg



# 七、编译打包上传项目

项目打包文件存储路径：

E:\RealTimeFaceCompare\Distribution\target

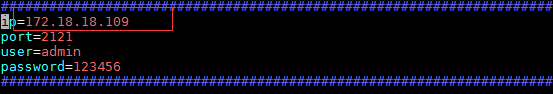


1. 将项目打包文件上传至主节点/opt/tool下
2. 解压项目文件到/opt下

# 八、服务配置：

## 1.FtpServer服务配置

1.进入/opt/RealTimeCompare/conf下，根据需要修改ftpAddress.properties文件中的ip地址，如下：



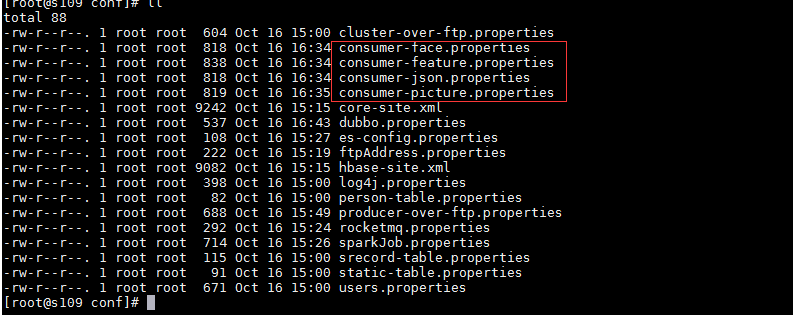
2.进入/opt/RealTimeCompare/conf下，根据需要修改以下文件中的bootstrap.servers地址，如下：

consumer-face.properties

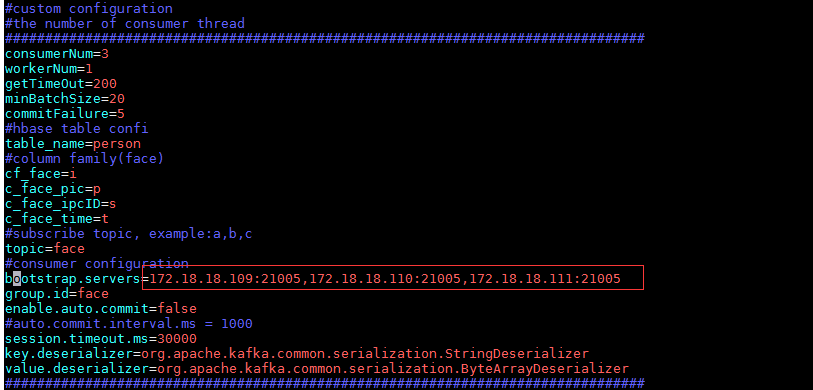
consumer-feature.properties

consumer-json.properties

consumer-picture.properties



修改bootstrap.servers地址：

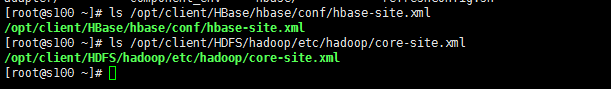


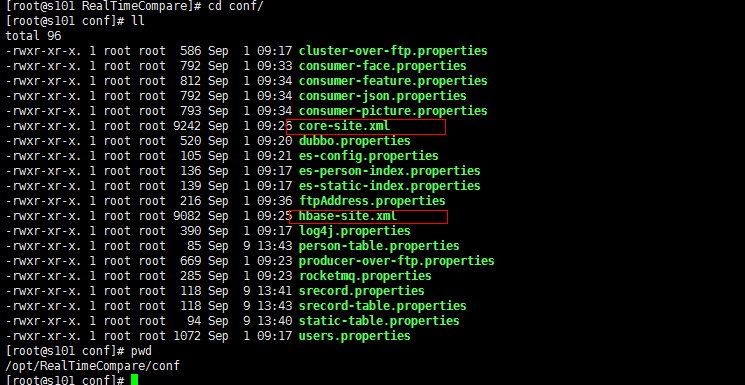
## 2.HBase服务配置

进入Fi客户端，将下面2个文件copy到/opt/RealTimeCompare/conf目录下：

/opt/client/HBase/hbase/conf/hbase-site.xml

/opt/client/HDFS/hadoop/etc/hadoop/core-site.xml





## 3.Elasticsearch服务配置

### 3.1 ES 索引及其映射关系生成

1.部署ES 索引映射关系，在/opt/RealTimeCompare/bin以找到下面两个脚本：

index-static.sh.templete

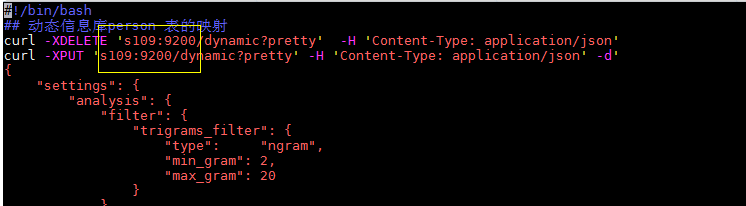
index-dynamic.sh.templete

2.进行如下操作：

cp index-static.sh.templeteindex-static.sh

cp index-dynamic.sh.templete index-dynamic.sh

3.修改index-static.sh 和 index-dynamic.sh里面的ES服务器的地址：



比如ES 服务在172.18.18.115 上面：

可以把地址直接换成如下：

curl –XPUT ' 172.18.18.115:9200/dynamic?pretty' -H 'Content-Type: application/json' -d'

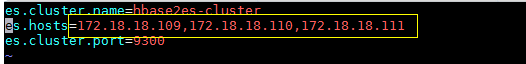
4.执行脚本，生成ES数据类型映射关系，创建索引：

sh index- static.sh

sh index-dynamic.sh

### 3.2 ES IP配置文件修改

进入/opt/RealTimeCompare/conf下，根据需要修改es-config.properties文件：



修改es.hosts 为ES服务所在的机器IP

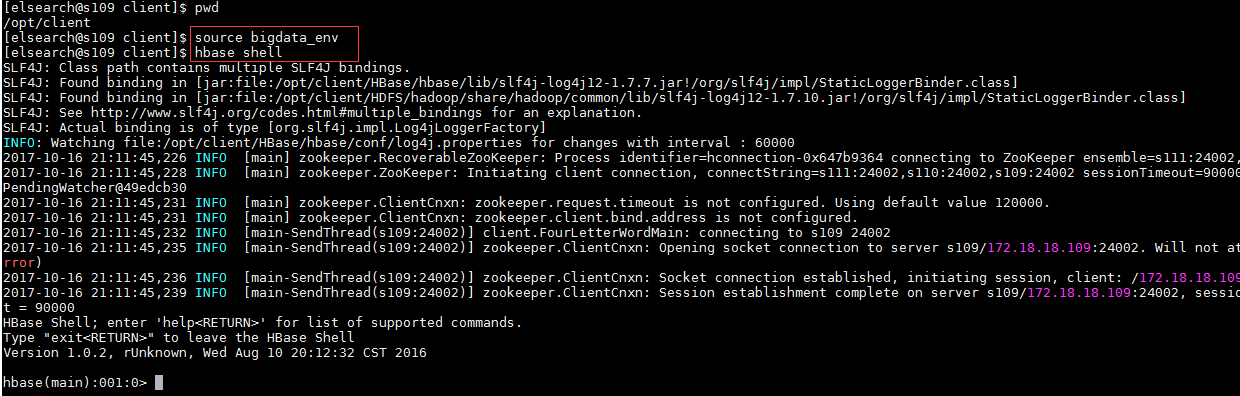
## 4.建表（创建ES索引之后建表）

### 1.新建一个xshell窗口连接Hbase客户端

cd /opt/client

source bigdata\_env

hbase shell



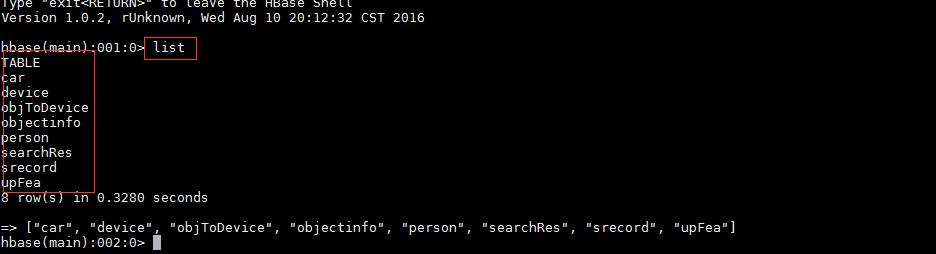
### 2.开始建表

设备表：先进入/opt/RealTimeCompare/sql路径下查看并复制device.sql文件中的建表语句，然后连接HBaseClient后粘贴建表语句并回车执行；设备一共有两张表（device、objToDevice）；

动态库建表：先进入/opt/RealTimeCompare/sql路径下查看并复制dynamic.sql文件中的建表语句，然后连接HBaseClient后粘贴建表语句并回车执行；动态库一共有四张表（person、car、upFea、searchRes）；

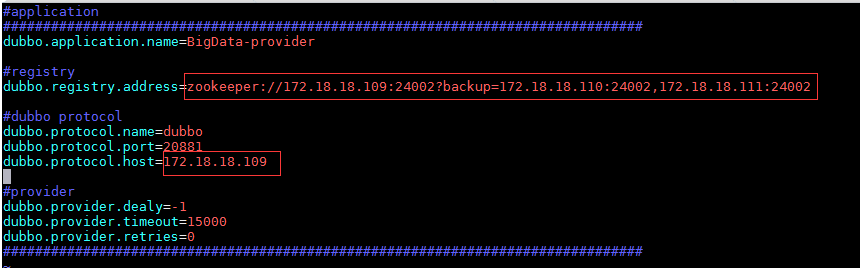
静态库建表：进入/opt/RealTimeCompare/bin路径下，执行create-table.sh脚本；静态库一共有两张表（objectinfo、srecord）；

### 3.在HBaseClient中查看是否建表成功



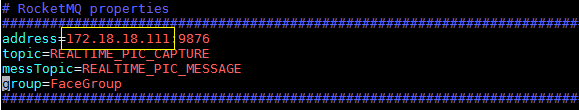
## 5.Dubbo服务配置

进入/opt/RealTimeCompare/conf下，根据需要修改dubbo.properties文件：



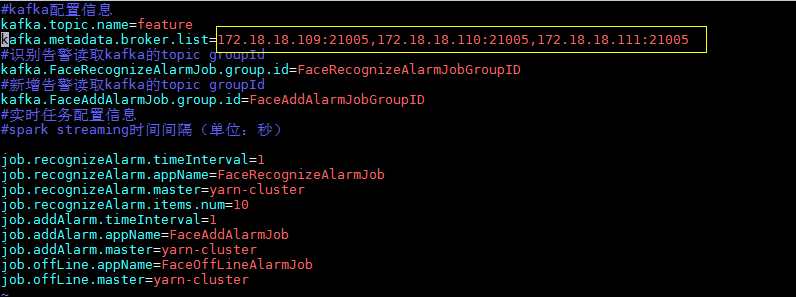
## 6. RocketMQ服务配置

进入/opt/RealTimeCompare/conf下，根据需要修改rocketmq.properties文件中的address地址为NameServer地址加端口号，如下：



## 7.SparkJob服务配置

进入/opt/RealTimeCompare/conf下，根据需要修改sparkJob.properties文件中的kafka.metadata.broker.list为集群kafak地址加kafka监听端口号，如下：

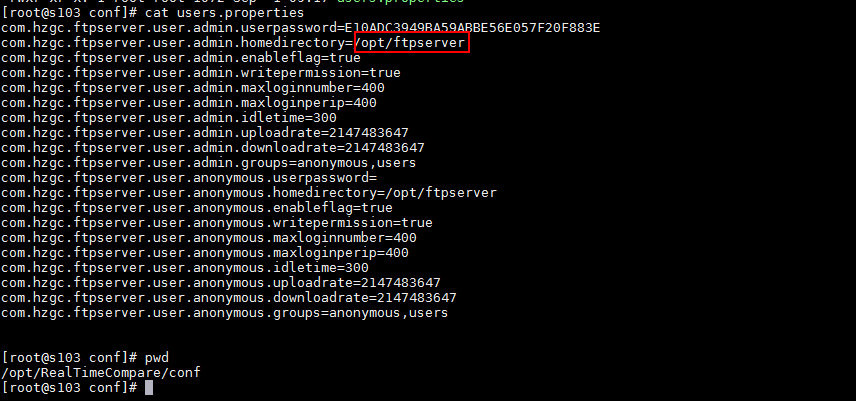


# 九、服务启动

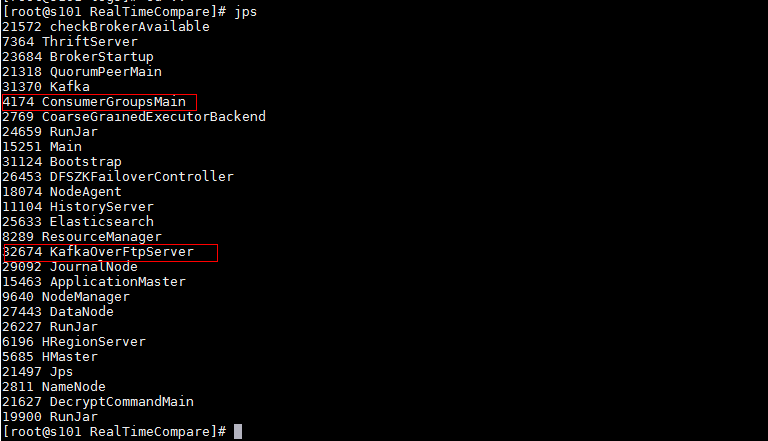
## 1.启动Ftp服务

进入s101机器下的/opt/RealTimeCompare 路径下：

查看/opt/RealTimeCompare/conf/users.properties文件中的homedirectory地址，并创建对应路径。



首先jps查看所有进程，查看是否有ConsumerGroupMain与KafkaOverFtpServer进程：



若没有直接执行脚本启动FtpServer服务与Kafaka consumer消费组；若存在使用“kill -9 进程号”将ConsumerGroupMain与KafkaOverFtpServer进程杀掉，若没有杀掉，多杀几次就好。

./bin/start-ftpserver.sh启动FtpServer服务

./bin/start-consumer.sh启动Kafaka consumer消费组

## 2.启动Dubbo服务

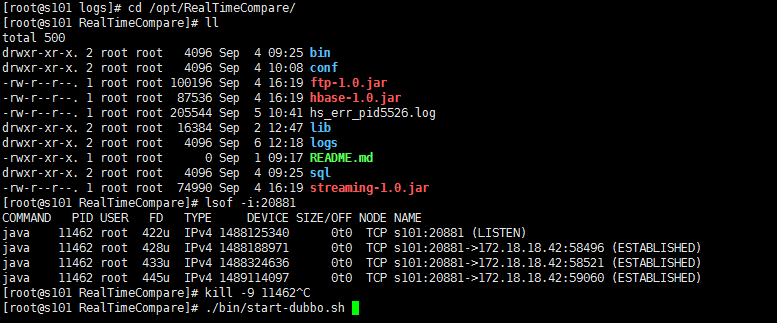
进入s101机器下的/opt/ RealTimeCompare路径下：

lsof -i:20881 （20881为Dubbo端口号）查看Dubbo进程号

kill -9 11462 杀掉Dubbo进程

./bin/start-dubbo.sh 启动Dubbo服务

tail -f logs/dubbo.log日志中没有报错信息，出现“Dubbo service server started!”，Dubbo启动成功。

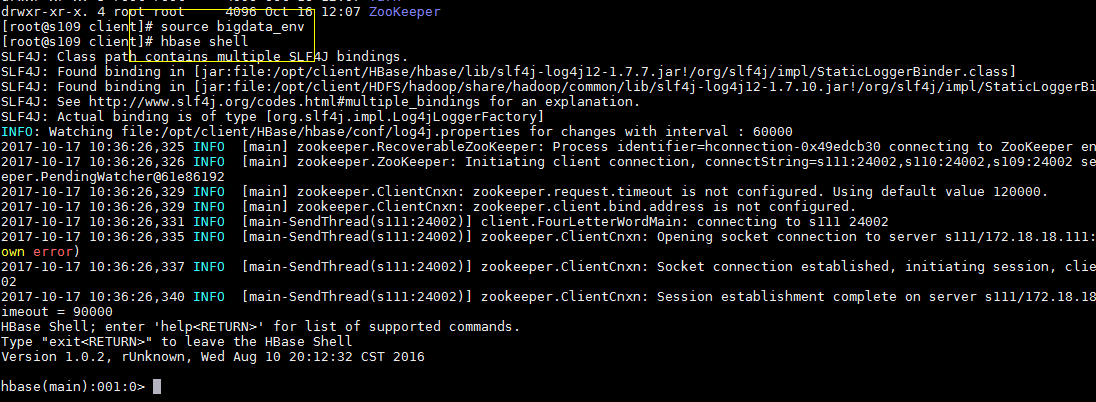


## 3.连接HbaseClient

进入/opt/client路径下：

source bigdata\_env

hbase shell



# 十、其他

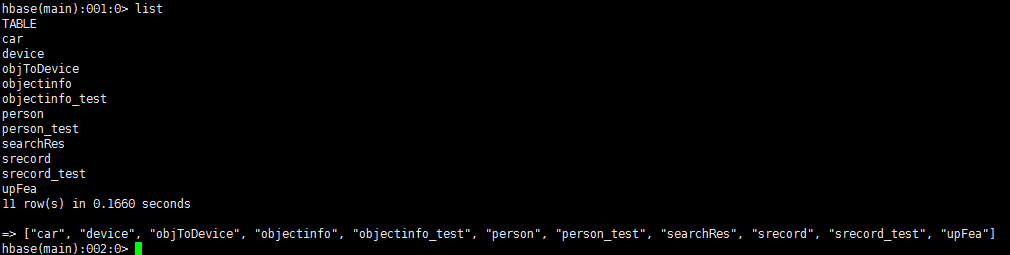
## 1.数据库建表

设备表：先进入s101机器下的/opt/RealTimeCompare/sql路径下查看并复制device.sql文件中的建表语句，然后连接HBaseClient后粘贴建表语句并回车执行；设备一共有两张表（device、objToDevice）；

静态库建表：进入s101机器下的/opt/RealTimeCompare/bin路径下，执行create-table.sh脚本；静态库一共有两张表（objectinfo、srecord）；

动态库建表：先进入s101机器下的/opt/RealTimeCompare/sql路径下查看并复制dynamic.sql文件中的建表语句，然后连接HBaseClient后粘贴建表语句并回车执行；动态库一共有四张表（person、car、upFea、searchRes）；

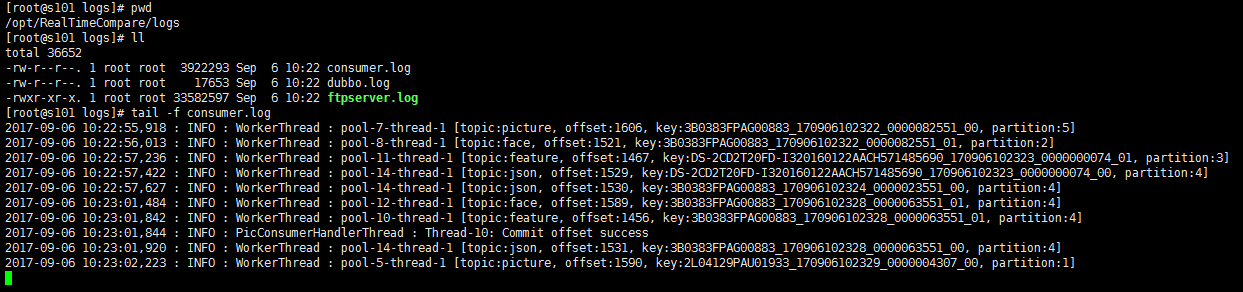
最后使用“list”命令查看是否建表成功。



## 2.查看log文件

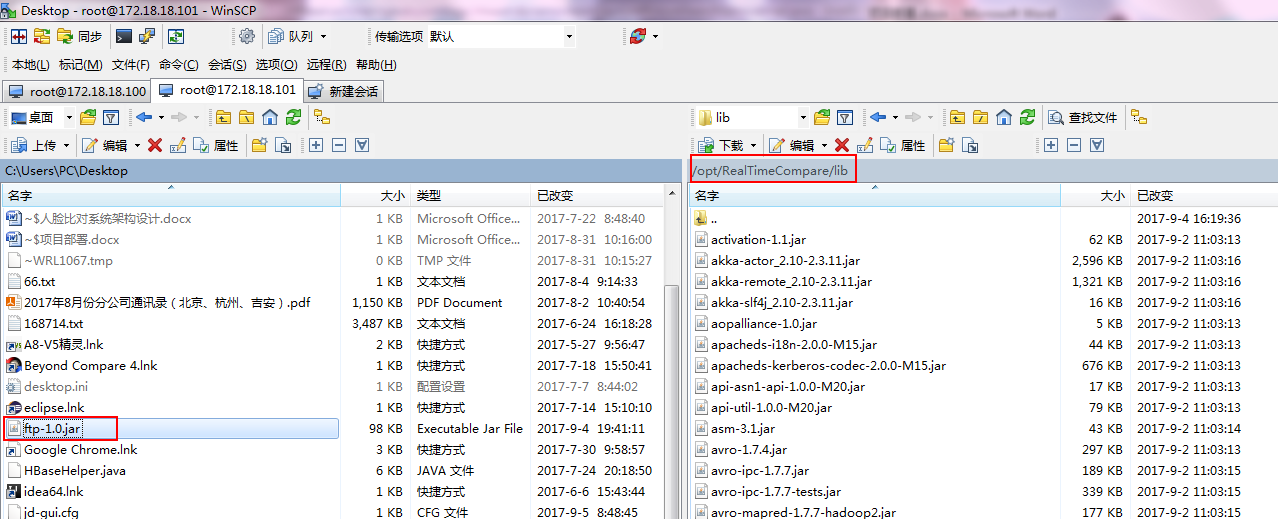
进入/opt/RealTimeCompare/logs路径下：

tail -f filename



## 3.替换项目中jar包

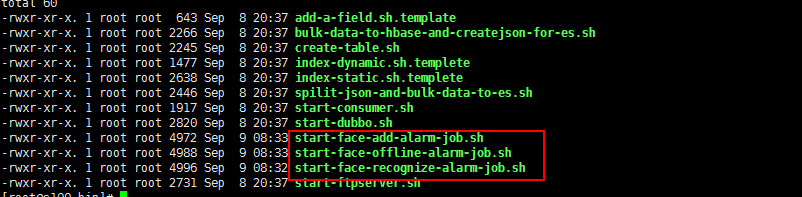
使用winSCP连接s101机器，进入/opt/RealTimeCompare/lib路径下：



选择左面本地机器编译打包完成的jar包，按“F5”拷贝至linux服务器中，或者使用MobaXterm软件在相同路径下替换所需jar包。替换完成后一般只需要重启Dubbo服务就可以了，若是修改了[ftp.jar](ftp://ftp.jar)[包需要先重启ftpserver](ftp://ftp.jar包需要先重启ftpserver)与consumer服务。

## 4.告警任务启动

进入/opt/RealTimeCompare/bin路径下：



选择对应的告警任务脚本并执行，然后在查看/opt/RealTimeCompare/logs中对应告警任务log日志是否有报错信息。

## 5.修改表字段

修改DeviceTable（设备）字段：package com.hzgc.hbase.device. DeviceTable

修改DynamicTable（动态库）字段：package com.hzgc.hbase.dynamicrepo. DynamicTable

修改ObjectInfoTable（静态库） 字段：package com.hzgc.dubbo.staticrepo.ObjectInfoTable

修改SrecordTable （静态库）字段：package com.hzgc.dubbo.staticrepo.SrecordTable