华为云鲲鹏

BigData Pro集群搭建



华为技术有限公司

目录

[1 实验概览 3](#_Toc63611108)

[1.1 实验介绍 3](#_Toc63611109)

[1.2 实验目的 3](#_Toc63611110)

[1.3 实验流程 4](#_Toc63611111)

[2 华为云环境准备 5](#_Toc63611112)

[2.1 实验介绍 5](#_Toc63611113)

[2.1.1 关于本实验 5](#_Toc63611114)

[2.1.2 实验目的 5](#_Toc63611115)

[2.2 购买华为云ECS 5](#_Toc63611116)

[2.2.1 进入华为云控制台 5](#_Toc63611117)

[2.2.2 购买华为云ECS 7](#_Toc63611118)

[2.2.3 配置安全组规则 10](#_Toc63611119)

[2.3 准备OBS服务 11](#_Toc63611120)

[2.3.1 购买OBS 11](#_Toc63611121)

[2.3.2 获取访问密钥AK/SK 14](#_Toc63611122)

[3 搭建Hadoop集群 16](#_Toc63611123)

[3.1 实验介绍 16](#_Toc63611124)

[3.1.1 关于本实验 16](#_Toc63611125)

[3.1.2 实验目的 16](#_Toc63611126)

[3.2 Hadoop集群搭建 16](#_Toc63611127)

[3.2.1 配置ECS 16](#_Toc63611128)

[3.2.2 获取JDK的安装路径 20](#_Toc63611129)

[3.3 搭建Hadoop伪分布式集群 21](#_Toc63611130)

[3.3.1 Hadoop安装 21](#_Toc63611131)

[3.3.2 伪分布式配置 23](#_Toc63611132)

[3.3.3 Hadoop与OBS互联 29](#_Toc63611133)

[4 Spark集群搭建 33](#_Toc63611134)

[4.1 实验介绍 33](#_Toc63611135)

[4.1.1 关于本实验 33](#_Toc63611136)

[4.1.2 实验目的 33](#_Toc63611137)

[4.2 Spark集群存算分离 33](#_Toc63611138)

[4.2.1 搭建Spark集群 33](#_Toc63611139)

[4.2.2 验证存算分离 36](#_Toc63611140)

[5 释放华为云服务 39](#_Toc63611141)

[5.1 释放弹性云服务器ECS 39](#_Toc63611142)

[5.2 删除OBS桶 40](#_Toc63611143)

[6 附录 安装OpenJDK 41](#_Toc63611144)

[7 附录 Linux常用命令 42](#_Toc63611145)

[7.1 ls命令 42](#_Toc63611146)

[7.2 cd命令 42](#_Toc63611147)

[7.3 pwd命令 42](#_Toc63611148)

[7.4 mkdir命令 43](#_Toc63611149)

[7.5 rm命令 43](#_Toc63611150)

[7.6 rmdir命令 43](#_Toc63611151)

[7.7 mv命令 44](#_Toc63611152)

[7.8 cp命令 44](#_Toc63611153)

[7.9 cat 命令 44](#_Toc63611154)

[7.10 more命令 45](#_Toc63611155)

[7.11 less命令 46](#_Toc63611156)

[7.12 head命令 47](#_Toc63611157)

[7.13 tail命令 47](#_Toc63611158)

[7.14 vim命令 47](#_Toc63611159)

[7.14.1 启动vim 47](#_Toc63611160)

[7.14.2 vim的模式 48](#_Toc63611161)

[7.14.3 插入命令 48](#_Toc63611162)

[7.14.4 查找命令 48](#_Toc63611163)

[7.14.5 替换命令 48](#_Toc63611164)

[7.14.6 撤销和重做 48](#_Toc63611165)

[7.14.7 删除命令 48](#_Toc63611166)

[7.14.8 拷贝和粘贴 49](#_Toc63611167)

[7.14.9 退出命令 49](#_Toc63611168)

[7.14.10 帮助命令 49](#_Toc63611169)

# 实验概览

## 实验介绍

本实验基于华为云OBS和华为云ECS服务构建一个存算分离的基本架构，并通过运行一个计算程序来完成存算分离架构的验证。本实验的实验数据存储在OBS中，通过在ECS上部署开源组件（Hadoop和Spark）构成计算环境，最后编写Spark程序访问存储在OBS上的数据进行计算（单词出现次数统计）并输出结果。

本实验的基本步骤包含：

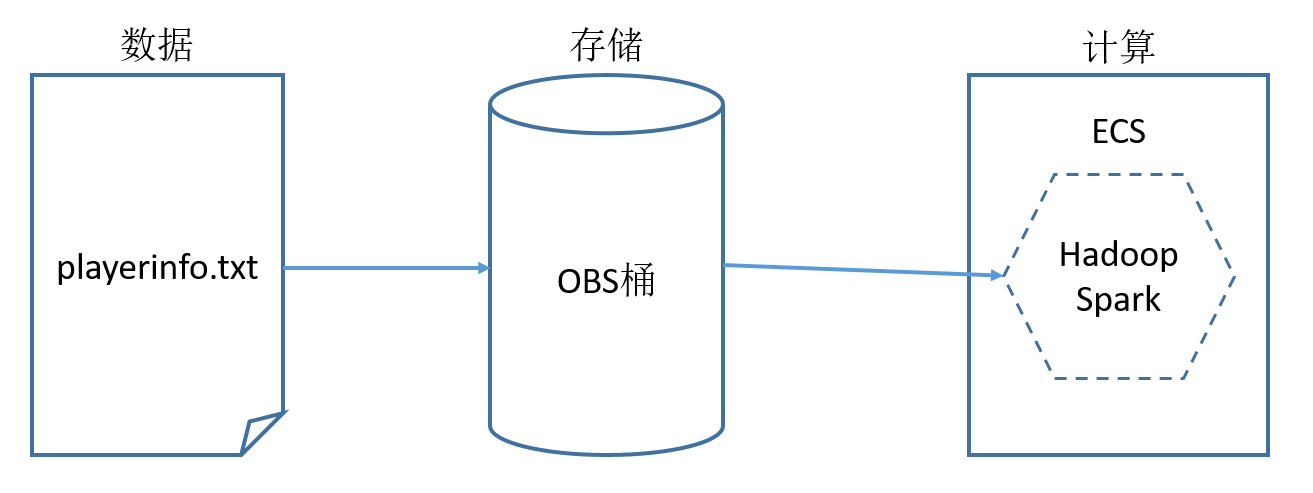
购买并配置ECS；

购买OBS并获取访问密钥AK/SK信息；

搭建Hadoop集群；

搭建Spark集群；

编写Spark程序验证存算分离。



## 实验目的

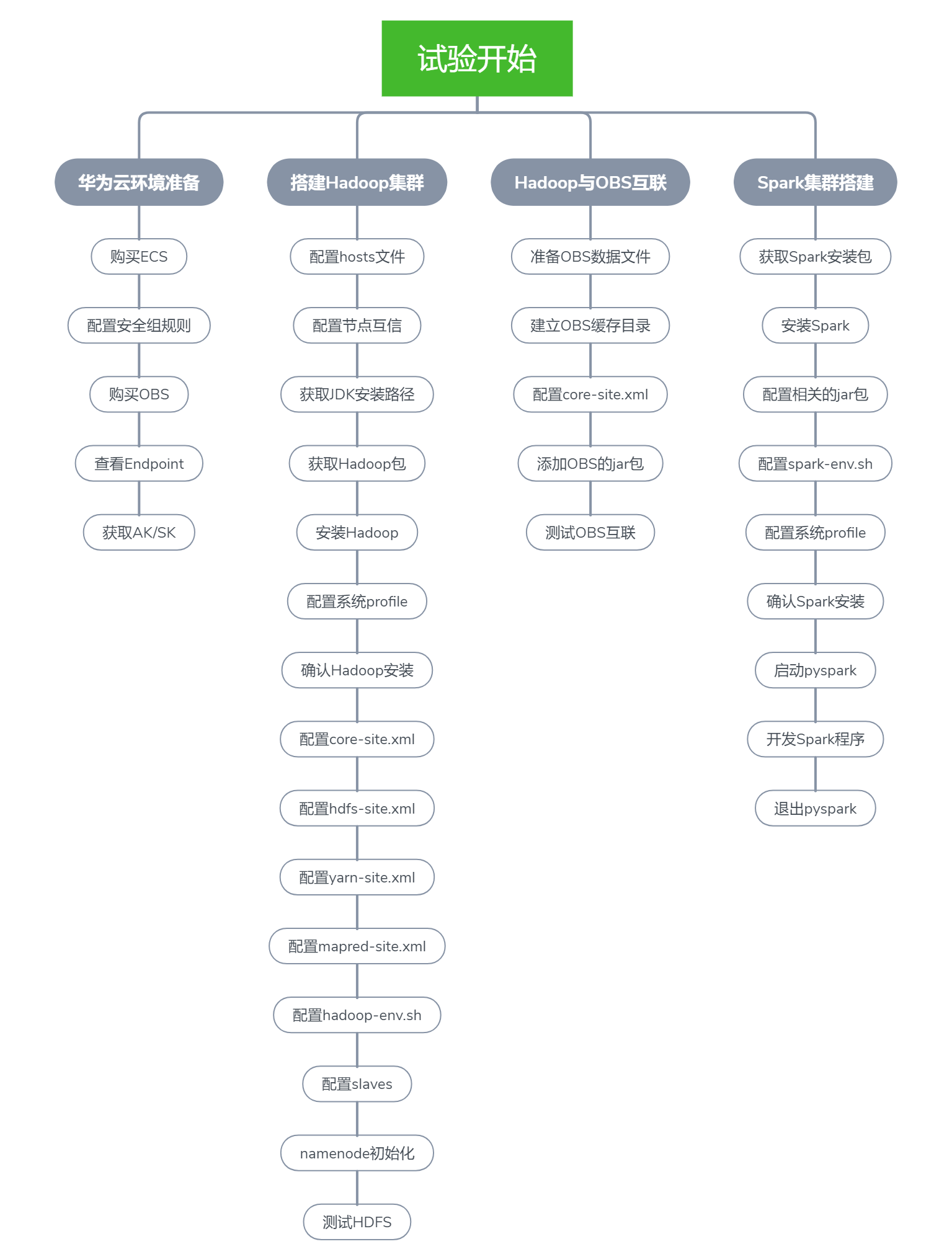
掌握华为云OBS的购买和使用。

掌握华为云ECS的购买和使用。

掌握Hadoop\Spark环境搭建。

掌握Spark程序读取OBS数据。

## 实验流程



验证存算分离

搭建Spark集群

配置ECS

安装Hadoop

伪分布式配置

# 华为云环境准备

## 实验介绍

### 关于本实验

本部分实验主要是通过购买华为云的相关服务构建起整个实验的云环境，提供后续集群搭建及实验的基础。主要包括购买华为云ECS和OBS，下载访问密钥文件获取AK/SK。

### 实验目的

掌握华为云ECS的购买

掌握华为云OBS的购买

掌握AK/SK的获取

## 购买华为云ECS

### 进入华为云控制台

登录华为云（网址：https://www.huaweicloud.com）

使用浏览器打开华为云页面，点击右上角的“登录”。

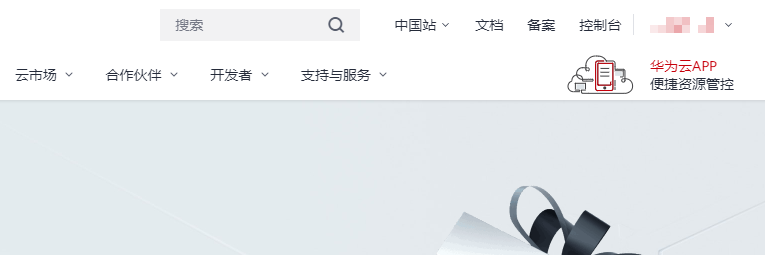


选择“账号登录”，输入账号、密码后点击“登录”按钮完成登录。

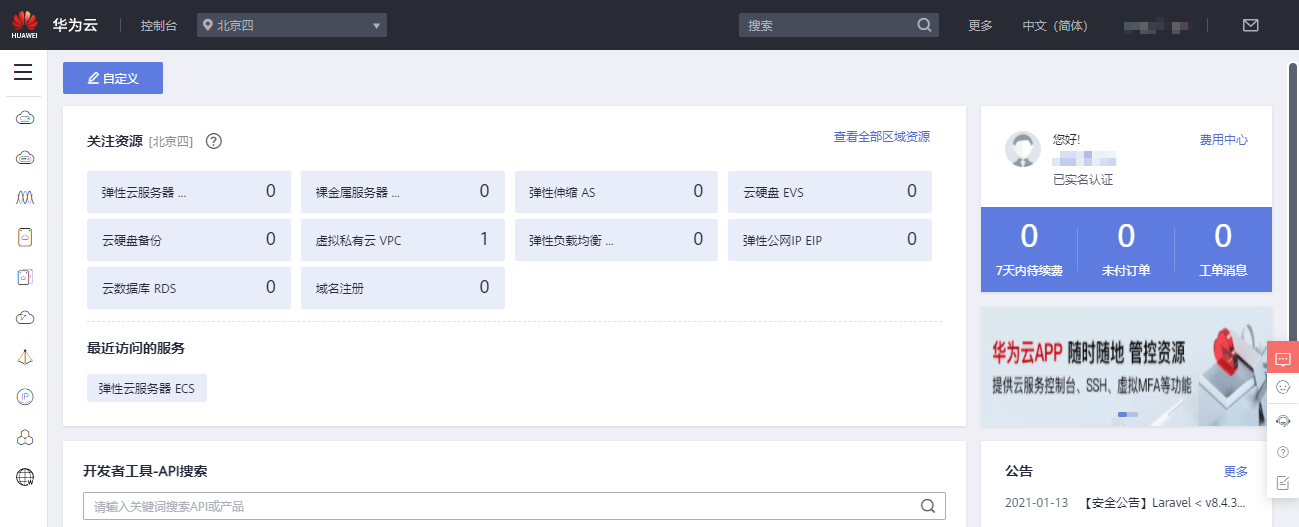


进入控制台

登录成功后点击“控制台”。



进入到华为云控制台，选择区域为“北京四”。



### 购买华为云ECS

进入弹性云服务器ECS购买界面

进入控制台后，鼠标移动到左侧三根横线处将“服务列表”展开，并在列表中点击“计算”分类里面的“弹性云服务器ECS”。



点击右上角的“购买弹性云服务器”



基础配置

选择“**自定义购买**”进入弹性云服务器的基础配置页面，选择“按需计费”，区域选择“北京四”，可用区“随机分配”。



CPU架构选择“鲲鹏计算”，规格选择“鲲鹏通用计算增强型”中的“kc1.xlarge.2 4vCPUs | 8GB”。



镜像选择“openEuler”中的“20.03 64bit with ARM(40GB)）”，系统盘选择“超高IO”。



点击“下一步：网络配置”。



网络配置

进入到网络配置页面，点击网络后面的下拉框选择VPC（后面的子网会自动选择，私有IP选择为“自动分配IP地址”），选择安全组。（如果没有VPC可以点击“前往控制台创建”）



弹性公网IP（现在购买，全动态BGP，按流量计费，50）。

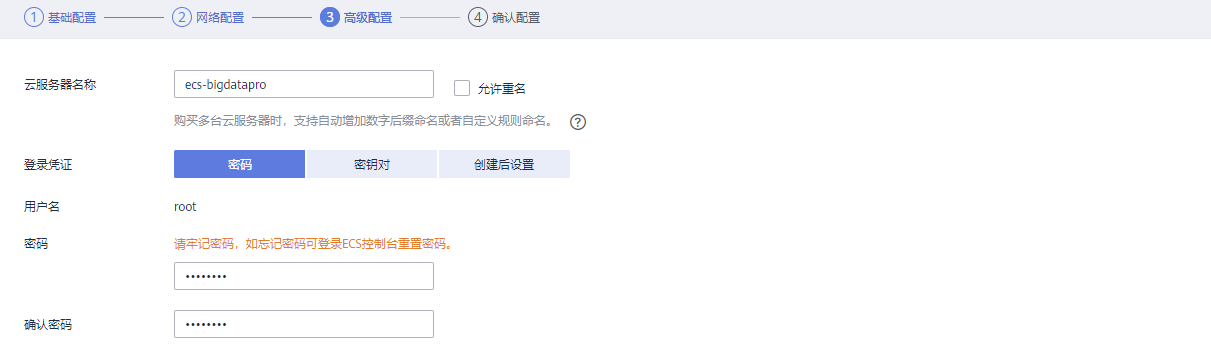


点击“下一步：高级配置”。



高级配置

进入到高级配置页面，服务器名称“ecs-bigdatapro”（可自定义），登录凭证选择“密码”，输入密码和确认密码（点击密码框会有密码规则提示），其他默认。





点击“下一步：确认配置”。



确认配置

进入确认配置界面，确认无误后购买数量选择“1”并勾选复选框。



点击“立即购买”



查看已购买的ECS

点击“返回云服务器列表”。



返回弹性云服务器列表页面，可以看到刚刚购买的弹性云服务器ecs-bigdatapro，等待其状态变为“运行中”则购买成功。



### 配置安全组规则

进入安全组

在云服务器控制台界面点击前面申请的ECS名称。



进入ECS后选择“安全组”。



修改规则

展开Sys-default安全组，点击“更改安全组规则”。



选择“入方向规则”，点击“添加规则”。



点击下拉框，选择“基本协议”中的“全部放通”，然后点击“确定”。



配置好安全组规则后，后面就可以通过远程工具进行连接了。如果提示规则已经存在，则无需处理。（实际上可以根据需要针对性的放开端口，这里为了试验方便选择了全部放通）

## 准备OBS服务

### 购买OBS

进入创建桶界面

在控制台的“服务列表”中点击“存储”下的“对象存储服务OBS”。



进入对象存储服务后，点击右上角的“创建桶”。



配置并创建桶

区域选择“华北-北京四”（与ECS选择同一区域），根据命名规则输入桶的名称（桶的名称需要全局唯一请自行确定，本例中为obs-bigdatapro），更多配置（多AZ存储，标准存储，私有），其他默认。





点击“立即创建”。



记录Endpoint值

返回对象存储服务，可以看到已经创建的桶。



点击刚创建的桶名称，进入概览后下拉滚动条可以看到桶的基本信息，记录其中的Endpoint内容（IPv4地址），后面配置连接OBS时会使用。

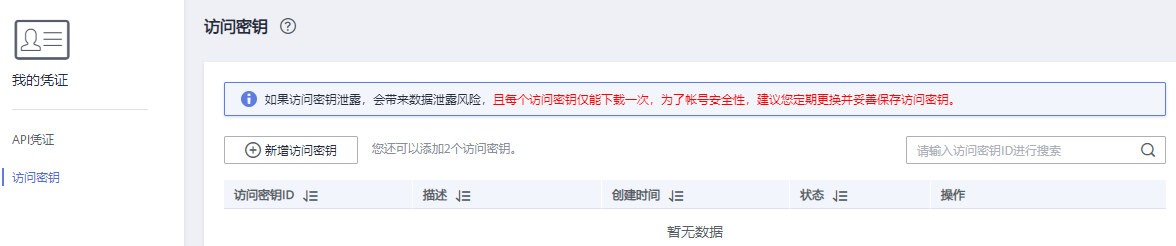


### 获取访问密钥AK/SK

在控制台鼠标放到右上角的账号上，在弹出的菜单中点击“我的凭证”。



在我的凭证界面点击“访问秘钥”，然后点击“新增访问秘钥”。



新增秘钥需要身份验证，输入收到的短息验证码后点击“确定”（如手机无法使用也可以通过邮件验证）。

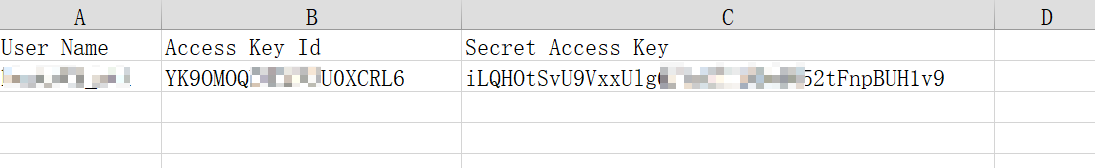


创建成功点击“立即下载”下载秘钥文件。



注意：每个密钥文件只能下载一次，请妥善保管。（若丢失可以重新创建密钥）

打开下载的密钥文件“credentials.csv”，即可得到AK/SK。



**SK**

**AK**

注：在后面配置OBS调用接口的时候，需要使用AK/SK进行签名验证。

# 搭建Hadoop集群

## 实验介绍

### 关于本实验

本部分实验需要在已经购买的ECS上搭建Hadoop集群，并且通过配置与华为云OBS服务互联，使Hadoop集群可读取OBS数据。

### 实验目的

掌握在ECS上搭建Hadoop集群方法

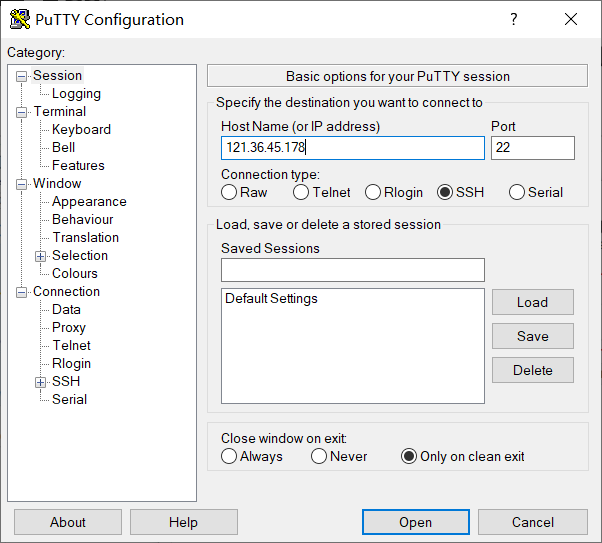
掌握Hadoop集群与华为云OBS互联方法

## Hadoop集群搭建

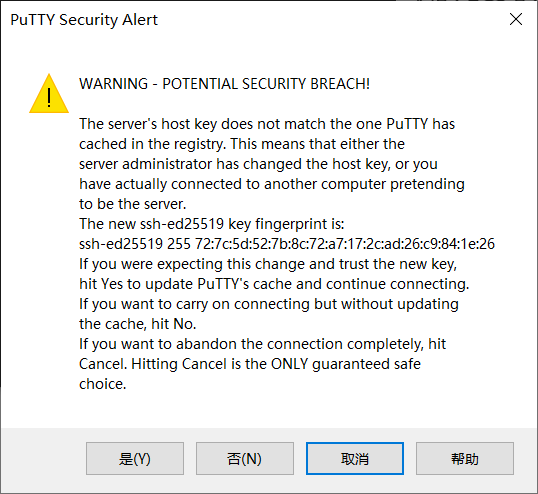
### 配置ECS

使用PuTTY远程登录ECS

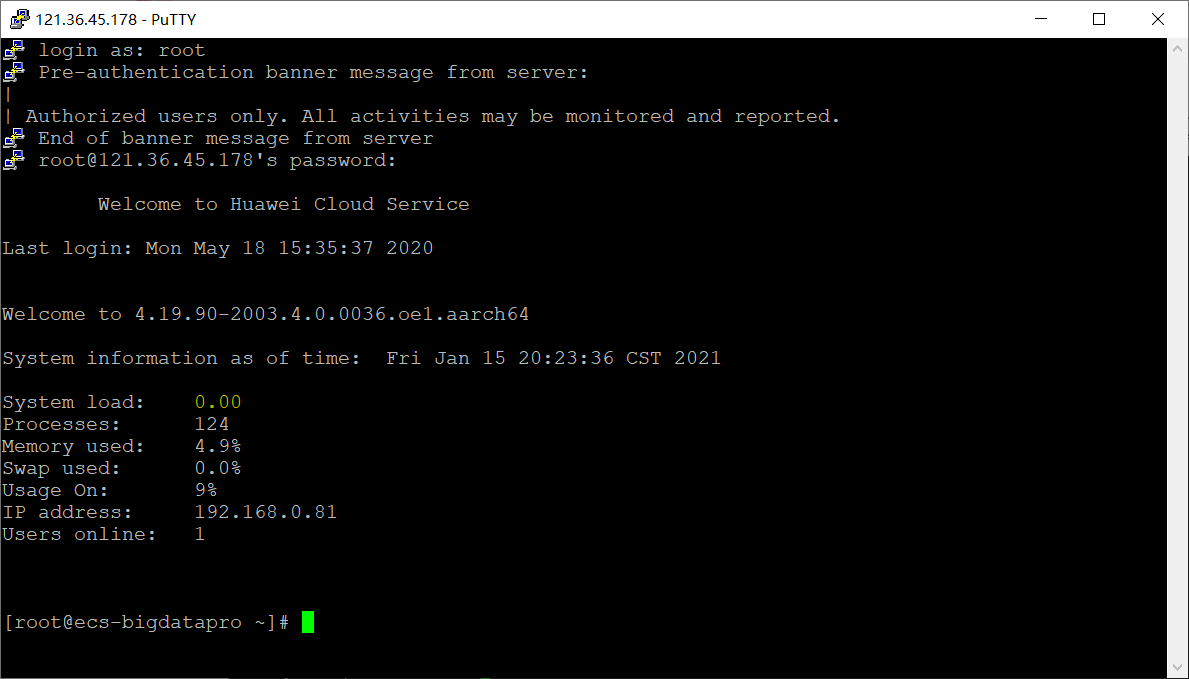
启动PuTTY（PuTTY是免费的SSH、Telnet程序，也可以使用自己习惯的工具），在Host Name处输入前面申请的ECS的**公网IP**地址（可以在云服务器控制台找到对应ECS的IP地址），选择SSH，点击“Open”。



弹出的对话框选择“是”



输入用户名root和密码（**此处输入密码不返显字符，直接输入即可**）进行登录。



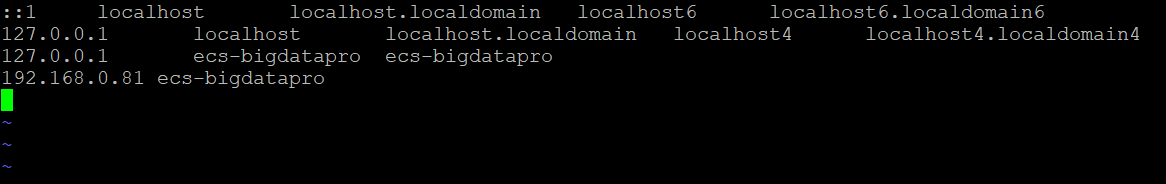
配置/etc/hosts文件

执行vim /etc/hosts命令使用vim编辑器编辑hosts文件。

vim /etc/hosts

按下i键进入编辑模式，添加ECS的**私有IP**（可以在云服务器控制台找到对应ECS的IP地址）和主机名（IP和主机名之间留有一个空格）。

192.168.0.81 ecs-bigdatapro

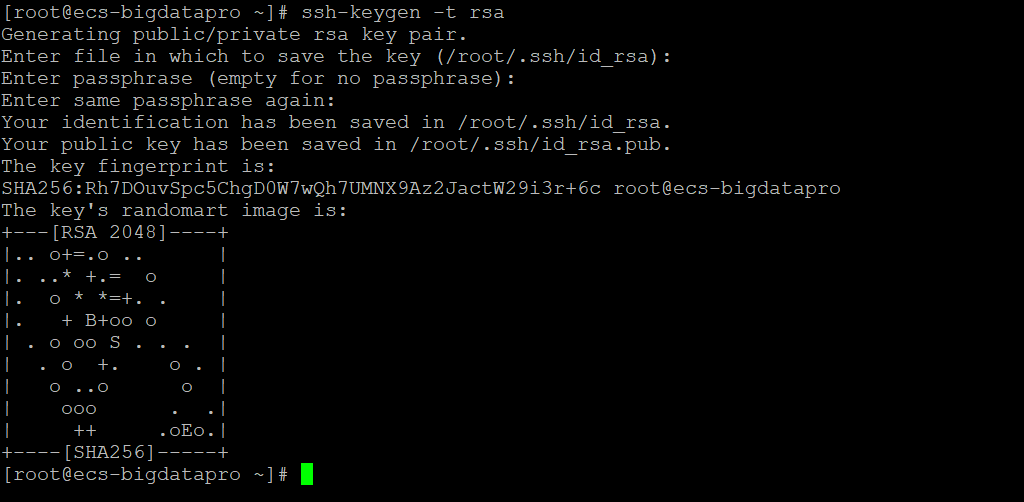


按ECS退出编辑模式，输入:wq保存并退出编辑器。

配置节点互信

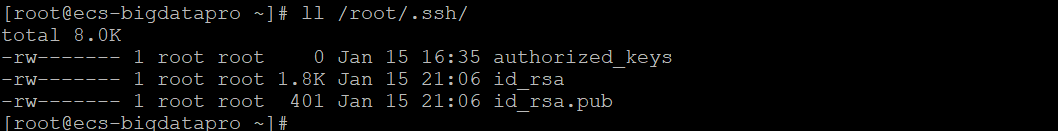
执行命令ssh-keygen -t rsa生成密钥文件（过程中连续按3回车直到命令结束）。

ssh-keygen -t rsa



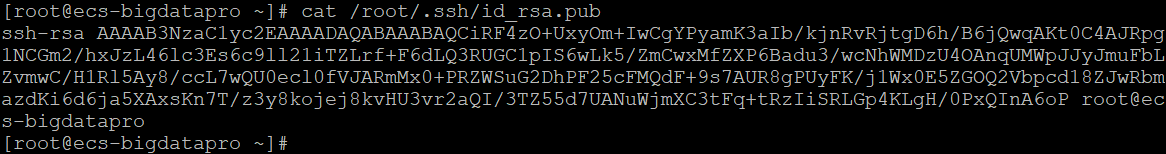
可以在/root/.ssh目录中看到生成的文件。

ll /root/.ssh/

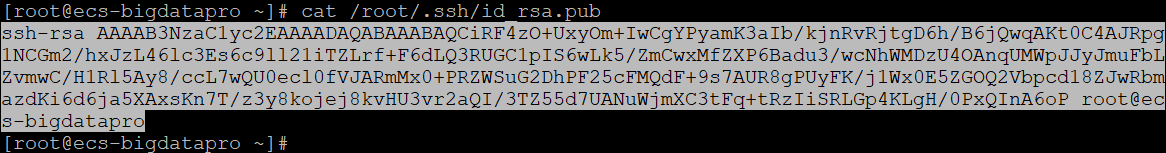


使用cat命令查看文件内容。

cat /root/.ssh/id\_rsa.pub

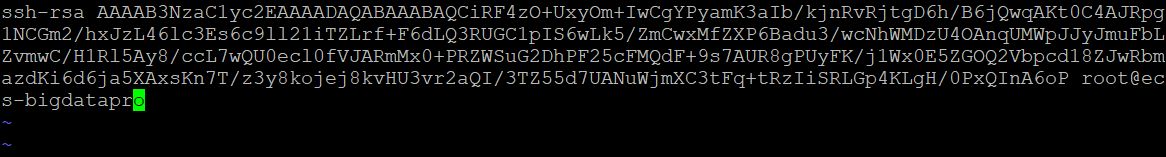


复制上面命令输出的内容（在PuTTY中选中即为复制）。



使用vim命令打开authorized\_keys文件，并把复制的内容粘贴到文件中（在PuTTY中右键即为粘贴）。

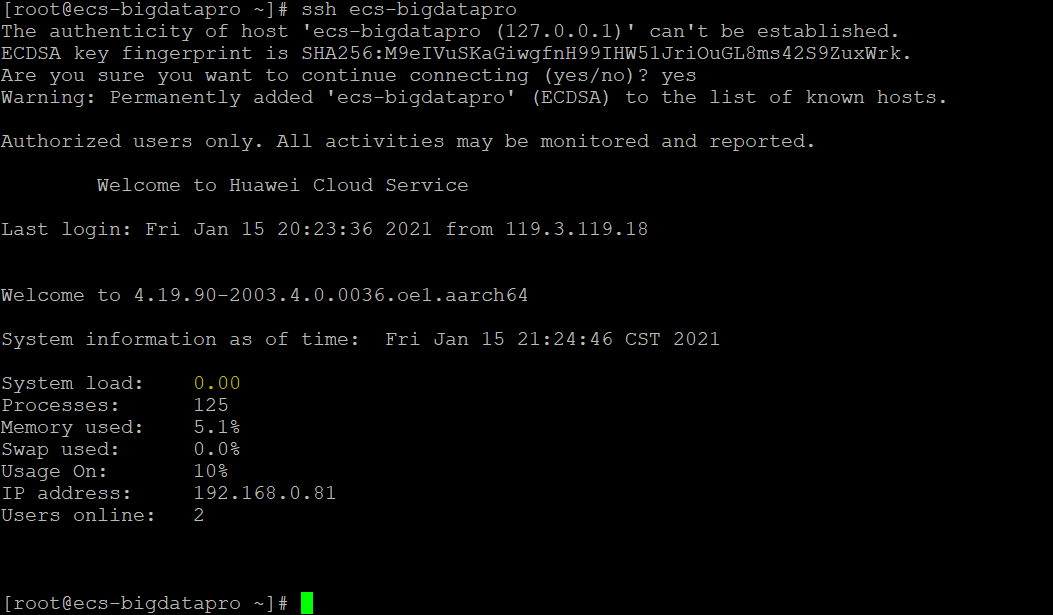
vim /root/.ssh/authorized\_keys



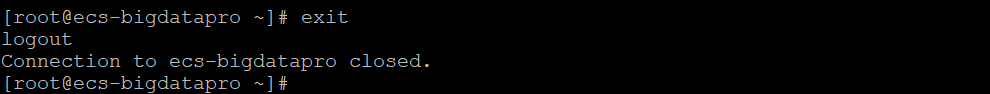
输入:wq保存并退出。

验证互信

输入ssh ecs-bigdatapro（注：ecs-bigdatapro为主机名），选择yes后，确保能够无密码登录即为成功。



输入exit可以关闭ssh连接。



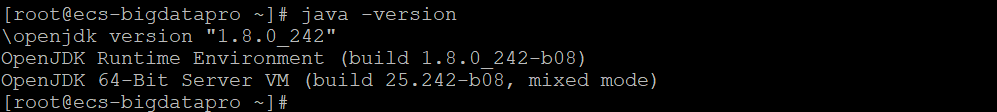
### 获取JDK的安装路径

确认是否已安装JDK

执行命令查询Java版本。

java -version

若输出如下结果，则说明ECS已安装JDK。



否则说明未安装JDK，则需要参考附录6安装OpenJDK。

获取JDK的实际安装目录（为后面配置JAVA\_HOME所需要的值）

输入命令which获取JDK的位置。

which java



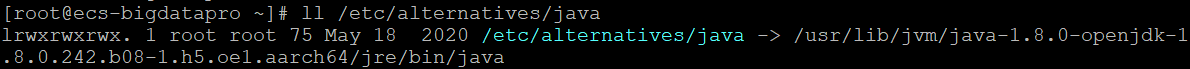
查看目录信息（可知目录为一个链接）。

ll /usr/bin/java



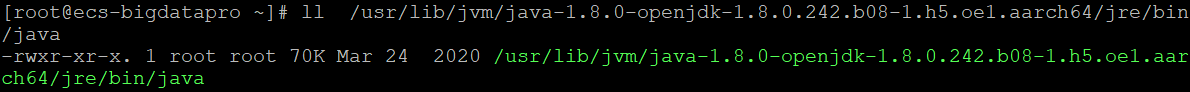
查看链接指向的目录（可知指向目录仍为链接）。

ll /etc/alternatives/java



继续查看链接指向的目录（可知目录已经是一个普通文件）。

ll /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b08-1.h5.oe1.aarch64/jre/bin/java



则jre之前的部分为JDK的实际安装目录。

/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b08-1.h5.oe1.aarch64

记录下此内容，作为后面配置**JAVA\_HOME**的值。

## 搭建Hadoop伪分布式集群

### Hadoop安装

下载hadoop安装包

进入root目录，使用wget命令下载hadoop安装包。

cd /root

wget https://archive.apache.org/dist/hadoop/common/hadoop-2.8.3/hadoop-2.8.3.tar.gz

创建hadoop所需的目录

mkdir -p /home/modules

mkdir -p /home/nm/localdir

**/home/modules**作为hadoop的安装目录；

**/home/nm/localdir**作为NodeManager的数据目录，存放执行Container所需的数据和运行过程中产生的临时数据，后面在配置文件yarn-site.xml中会使用yarnyarn.nodemanager.local-dirs属性指定。

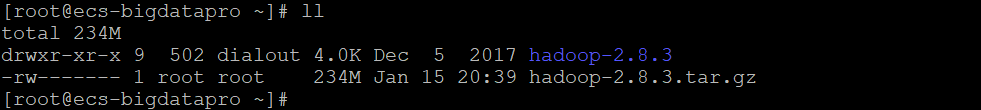
安装hadoop

解压hadoop安装包。

cd /root

tar -zxvf hadoop-2.8.3.tar.gz

ll

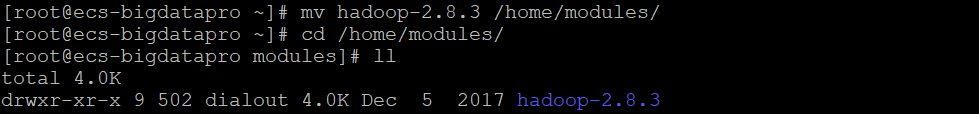


把解压后的目录移动到安装目录下。

mv hadoop-2.8.3 /home/modules/

cd /home/modules/

ll



配置系统环境变量

使用vim打开系统配置文件。

vim /etc/profile

按i键进入编辑模式，在文件尾部添加如下内容**（注意JAVA\_HOME的值）**：

export JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b08-1.h5.oe1.aarch64

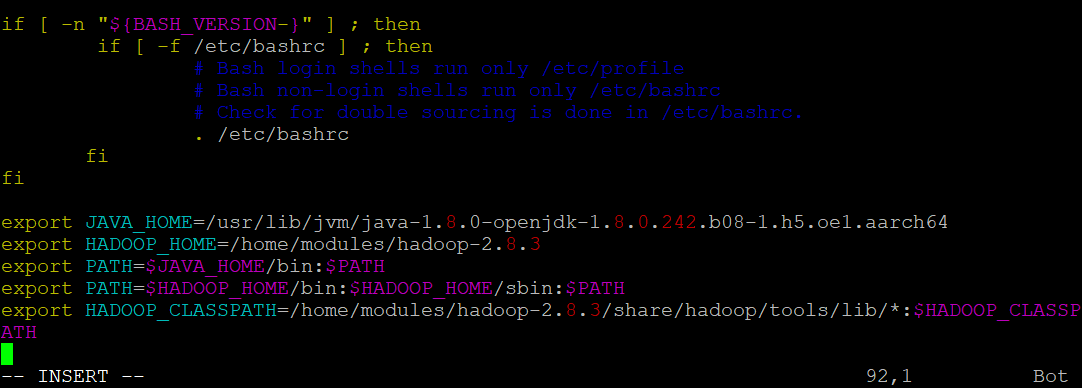
export HADOOP\_HOME=/home/modules/hadoop-2.8.3

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

export PATH=$HADOOP\_HOME/bin:$HADOOP\_HOME/sbin:$PATH

export HADOOP\_CLASSPATH=/home/modules/hadoop-2.8.3/share/hadoop/tools/lib/\*:$HADOOP\_CLASSPATH

**注：其中JAVA\_HOME的值为JDK安装路径，如果不一致需要自行修改。**



按ECS退出编辑模式，输入:wq保存并退出。

使用source命令使得环境变量生效。

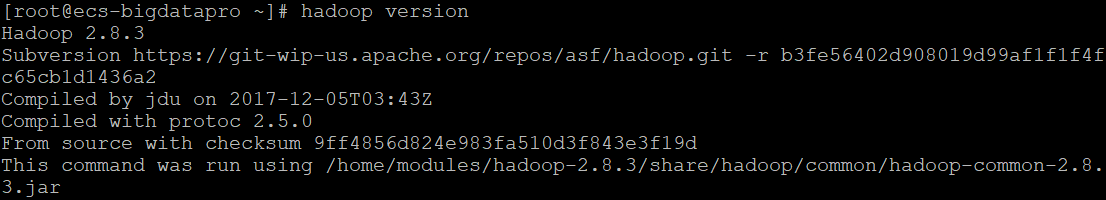
source /etc/profile

验证hadoop安装

进入root目录，查看hadoop版本信息。

cd /root

hadoop version



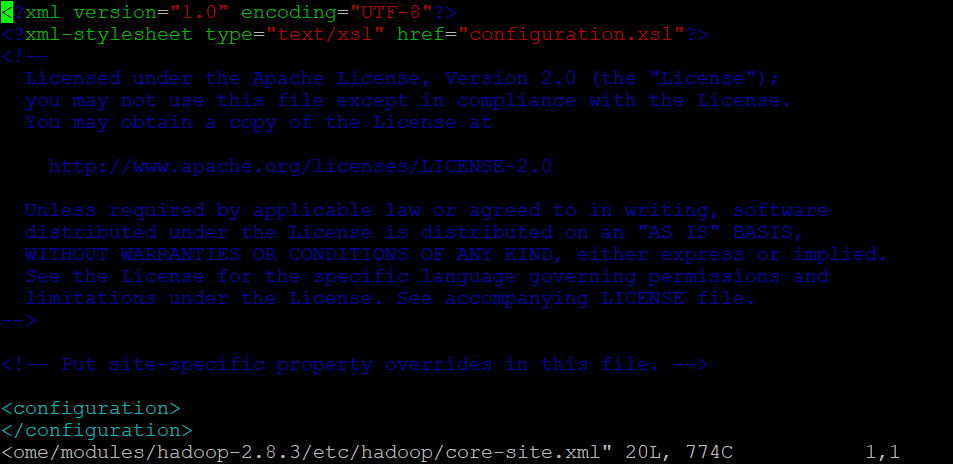
正确显示hadoop版本信息，安装完成。

### 伪分布式配置

配置hadoop的core-site.xml文件

使用vim命令打开配置文件。

vim /home/modules/hadoop-2.8.3/etc/hadoop/core-site.xml



按i键进入编辑模式，复制下面的配置内容粘贴到<configuration>和</configuration>之间**（其中 ecs-bigdatapro为主机名，如果不一致需要自行修改）**：

<property>

<name>fs.defaultFS</name>

<value>hdfs://**ecs-bigdatapro**:8020</value>

</property>

<property>

<name>hadoop.tmp.dir</name>

<value>/home/modules/hadoop-2.8.3/tmp</value>

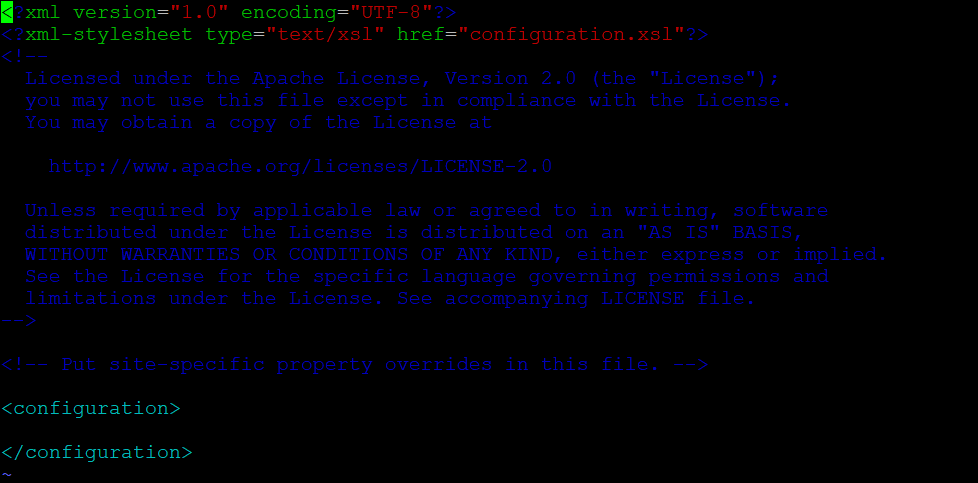
</property>

按ECS退出编辑模式，输入:wq保存并退出。

配置hadoop的hdfs-site.xml文件

使用vim打开配置文件。

vim /home/modules/hadoop-2.8.3/etc/hadoop/hdfs-site.xml



按i键进入编辑模式，复制下面的配置内容粘贴到<configuration>和</configuration>之间**注（其中两处ecs-bigdatapro为主机名，如果不一致需要自行修改）**：

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

<name>dfs.namenode.secondary.http-address</name>

<value>**ecs-bigdatapro**:50090</value>

</property>

<property>

<name>dfs.namenode.secondary.https-address</name>

<value>**ecs-bigdatapro**:50091</value>

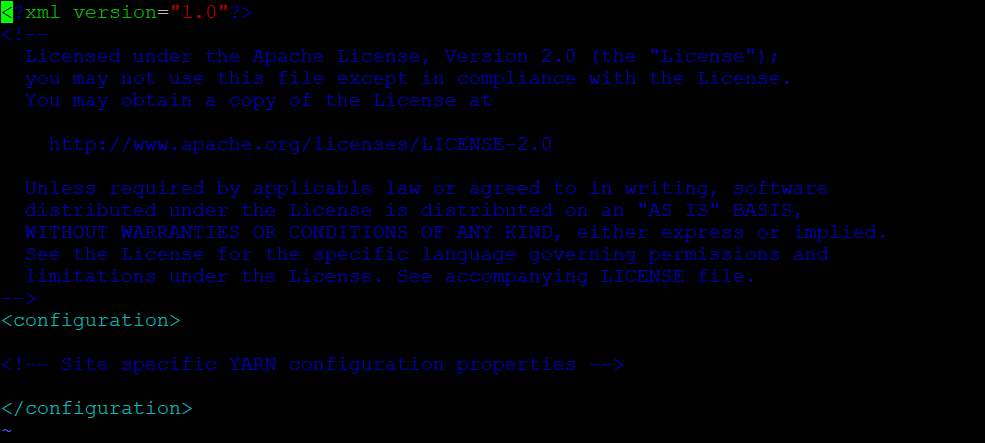
</property>

按ECS退出编辑模式，输入:wq保存并退出。

配置hadoop的yarn-site.xml文件

使用vim打开配置文件。

vim /home/modules/hadoop-2.8.3/etc/hadoop/yarn-site.xml



按i键进入编辑模式，复制下面的配置内容粘贴到<configuration>和</configuration>之间**（其中两处ecs-bigdatapro为主机名，如果不一致需要自行修改）**：

<property>

<name>yarn.nodemanager.local-dirs</name>

<value>/home/nm/localdir</value>

</property>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.resourcemanager.hostname</name>

<value>**ecs-bigdatapro**</value>

</property>

<property>

<name>yarn.resourcemanager.scheduler.class</name>

<value>org.apache.hadoop.yarn.server.resourcemanager.scheduler.fair.FairScheduler</value>

</property>

<property>

<name>yarn.log-aggregation-enable</name>

<value>true</value>

</property>

<property>

<name>yarn.log.server.url</name>

<value>http://**ecs-bigdatapro**:19888/jobhistory/logs</value>

</property>

按ECS退出编辑模式，输入:wq保存并退出。

配置hadoop的mapred-site.xml文件

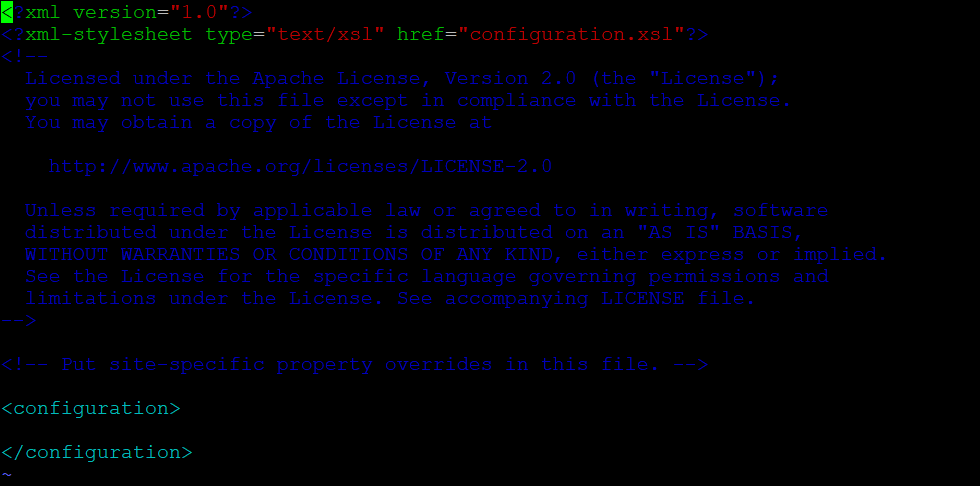
使用hadoop自带模板文件创建mapred-site.xml配置文件：

cd /home/modules/hadoop-2.8.3/etc/hadoop/

mv mapred-site.xml.template mapred-site.xml

使用vim打开配置文件。

vim /home/modules/hadoop-2.8.3/etc/hadoop/mapred-site.xml



按i键进入编辑模式，复制下面的配置内容粘贴到<configuration>和</configuration>之间**（其中两处ecs-bigdatapro为主机名，如果不一致需要自行修改）**：

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

<property>

<name>mapreduce.jobhistory.address</name>

<value>**ecs-bigdatapro**:10020</value>

</property>

<property>

<name>mapreduce.jobhistory.webapp.address</name>

<value>**ecs-bigdatapro**:19888</value>

</property>

<property>

<name>mapred.task.timeout</name>

<value>1800000</value>

</property>

按ECS退出编辑模式，输入:wq保存并退出。

配置hadoop环境变量

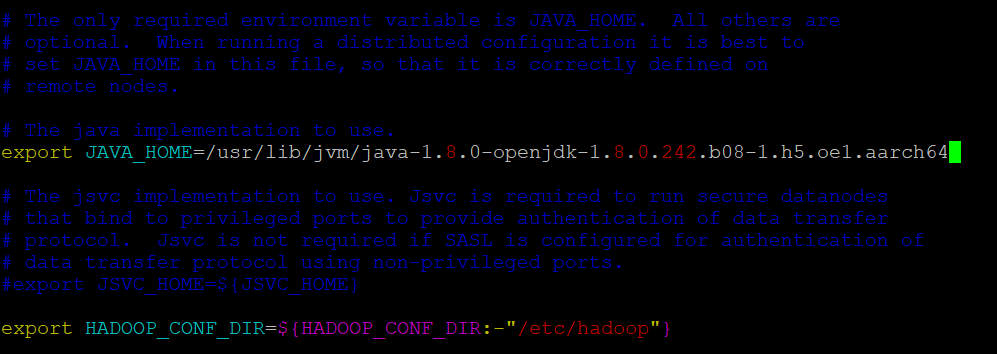
使用vim命令打开配置文件。

vim /home/modules/hadoop-2.8.3/etc/hadoop/hadoop-env.sh

按i键进入编辑模式，找到JAVA\_HOME并修改如下：

export JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b08-1.h5.oe1.aarch64

**注：其中JAVA\_HOME的值和前面一样要确认。**



按ECS退出编辑模式，输入:wq保存并退出。

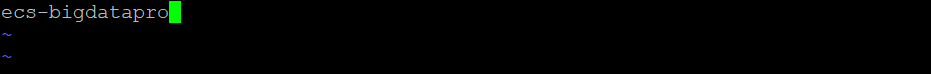
配置slaves

使用vim命令打开配置文件

vim /home/modules/hadoop-2.8.3/etc/hadoop/slaves

按i键进入编辑模式，删去原有内容，添加以下内容**（其中ecs-bigdatapro为主机名，如果不一致需要自行修改）**：

ecs-bigdatapro

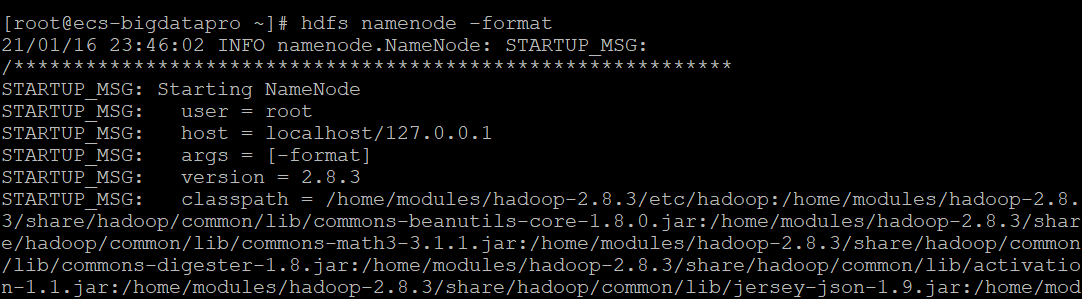


按ECS退出编辑模式，输入:wq保存并退出。

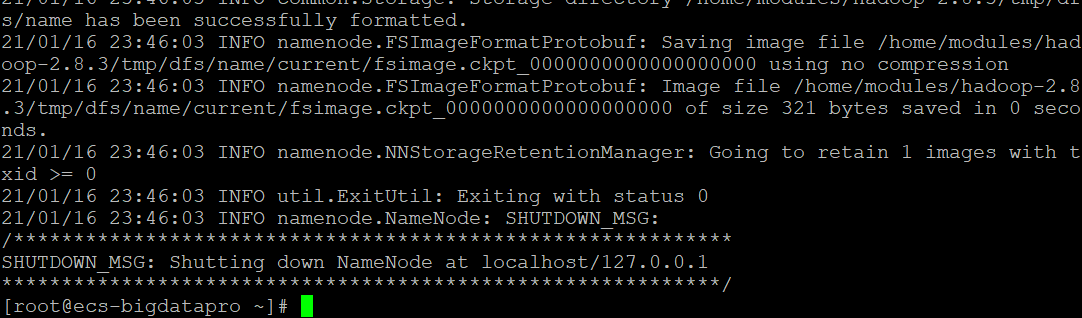
namenode初始化

执行下列命令初始化namenode。

hdfs namenode -format



中间略去若干输出信息…

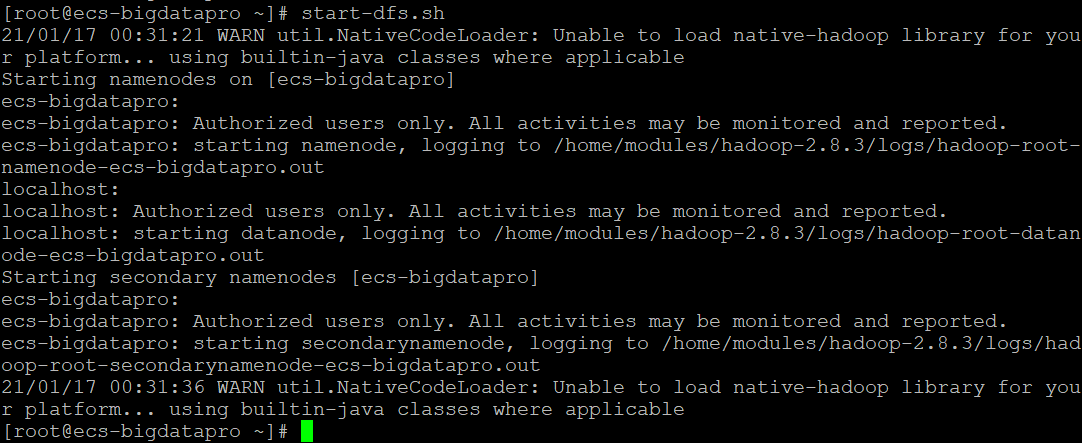


初始化成功。

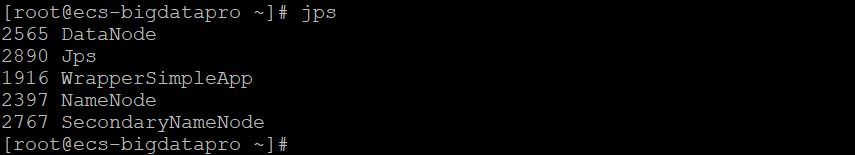
启动HDFS

执行以下命令启动HDFS。

start-dfs.sh



使用JPS查看启动的进程。

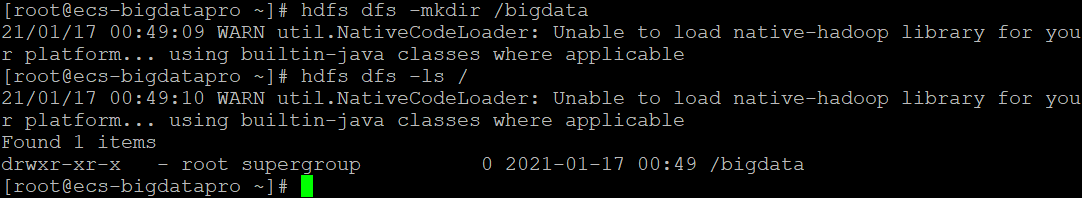


测试HDFS命令

使用mkdir命令建立bigdata目录

hdfs dfs -mkdir /bigdata

hdfs dfs -ls /



### Hadoop与OBS互联

在OBS上传文件

进入前面建立的OBS桶obs-bigdatapro，选择“对象”点击“上传对象”。



在弹出的对话框中点击“添加文件”，选择playerinfo.txt文件。



点击“上传”。



OBS文件上传成功。



建立OBS的缓存目录

使用mkdir建立OBS所需的缓存目录。

mkdir -p /home/modules/data/buf

配置OBS信息

使用vim打开hadoop的配置文件core-site.xml。

vim /home/modules/hadoop-2.8.3/etc/hadoop/core-site.xml

按i键进入编辑模式，在<configuration>和</configuration>之间添加如下内容（不要删除原有内容，并按代码后面的说明进行参数替换）。

<property>

<name>fs.obs.readahead.inputstream.enabled</name>

<value>true</value>

</property>

<property>

<name>**fs.obs.access.key**</name>

<value>**YK9OMOQYDOYB9U0XCRL8**</value>

</property>

<property>

<name>**fs.obs.secret.key**</name>

<value>**iLQHOtSvU9VxxUlg1YH1U0YpQbGW52tFnpBUH1v6**</value>

</property>

<property>

<name>**fs.obs.endpoint**</name>

<value>**obs.cn-north-4.myhuaweicloud.com:**5080</value>

</property>

<property>

<name>fs.obs.buffer.dir</name>

<value>/home/modules/data/buf</value>

</property>

<property>

<name>fs.obs.impl</name>

<value>org.apache.hadoop.fs.obs.OBSFileSystem</value>

</property>

<property>

<name>fs.obs.connection.ssl.enabled</name>

<value>false</value>

</property>

<property>

<name>fs.obs.fast.upload</name>

<value>true</value>

</property>

注意：其中fs.obs.access.key、fs.obs.secret.key、fs.obs.endpoint的值要替换成实际内容。

**fs.obs.access.key、fs.obs.secret.key**为2.3.2步骤5中获取的AK和SK的值（在下载的密钥文件中可以找到）。

**fs.obs.endpoint**为2.3.1步骤5中记录的OBS的endpoint的值（可以进入obs-bigdatapro桶后在“概览”的“基本信息”中再次查看到endpoint，端口号不修改）。

按ECS退出编辑模式，输入:wq保存并退出。

添加所需jar包

进入root目录，使用wget命令下载OBSFileSystem相关jar包：

cd /root

wget https://raw.githubusercontent.com/huaweicloud/obsa-hdfs/master/release/hadoop-huaweicloud-2.8.3-hw-39.jar

把下载的jar包复制到所需目录中。

cp hadoop-huaweicloud-2.8.3-hw-39.jar /home/modules/hadoop-2.8.3/share/hadoop/common/lib/

cp hadoop-huaweicloud-2.8.3-hw-39.jar /home/modules/hadoop-2.8.3/share/hadoop/tools/lib

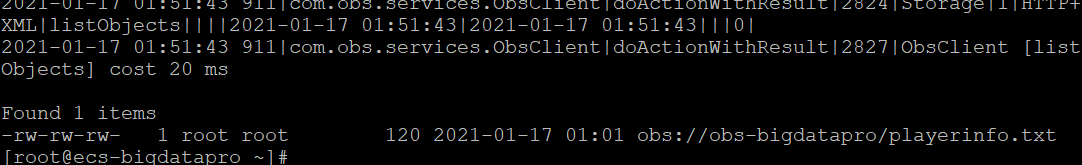
cp hadoop-huaweicloud-2.8.3-hw-39.jar /home/modules/hadoop-2.8.3/share/hadoop/httpfs/tomcat/webapps/webhdfs/WEB-INF/lib/

cp hadoop-huaweicloud-2.8.3-hw-39.jar /home/modules/hadoop-2.8.3/share/hadoop/hdfs/lib/

测试OBS互联

执行HDFS命令查看OBS文件（注意目录最后要有“/”）。

hdfs dfs -ls obs://obs-bigdatapro/



Hadoop集群与OBS互联成功。

**注：命令参数“obs-bigdatapro”为前面创建桶的名称，需要根据实际情况修改。**

# Spark集群搭建

## 实验介绍

### 关于本实验

本部分实验介绍安装Spark集群，并使Spark能够读取OBS数据，使用Python编写Spark程序处理OBS中的数据（单词统计）。该实验使用Spark集群+OBS实现存算分离，提高计算性能。

### 实验目的

掌握Spark集群搭建

掌握Spark集群与OBS互联

使用Python编写Spark程序

## Spark集群存算分离

### 搭建Spark集群

获取Spark安装包

进入root目录下载Spark安装包。

cd /root

wget <https://archive.apache.org/dist/spark/spark-2.3.0/spark-2.3.0-bin-without-hadoop.tgz>

解压Spark安装包

使用tar命令解压Spark的安装包。

tar -zxvf spark-2.3.0-bin-without-hadoop.tgz

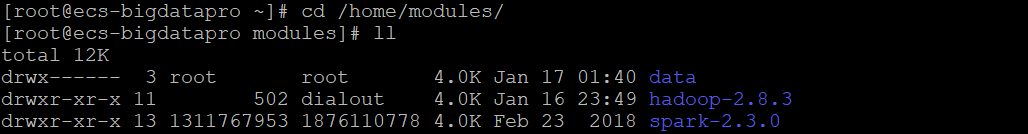
移动Spark到/home/modules目录下并重命名为spark-2.3.0。

mv spark-2.3.0-bin-without-hadoop /home/modules/spark-2.3.0

进入安装目录查看Spark。

cd /home/modules/

ll



配置相关的jar包

复制前面下载的OBSFileSystem相关jar包和hadoop的相关jar包到Spark的jars目录下。

cp /root/hadoop-huaweicloud-2.8.3-hw-39.jar /home/modules/spark-2.3.0/jars/

cp /home/modules/hadoop-2.8.3/share/hadoop/common/lib/snappy-java-1.0.4.1.jar /home/modules/spark-2.3.0/jars/

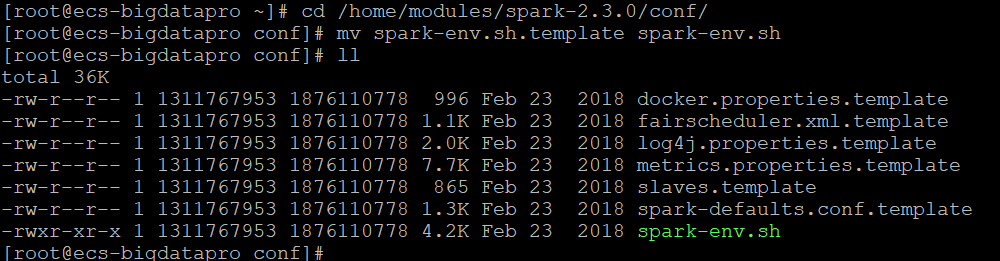
进行Spark的配置

通过mv命令使用Spark自带的配置文件模板创建出所需的配置文件。

cd /home/modules/spark-2.3.0/conf/

mv spark-env.sh.template spark-env.sh

ll



使用vim命令打开配置文件。

vim spark-env.sh

按下i键盘进入编辑模式，在文件末尾添加如下内容**（注意确认JAVA\_HOME值是否与本机一致）**。

export JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.242.b08-1.h5.oe1.aarch64

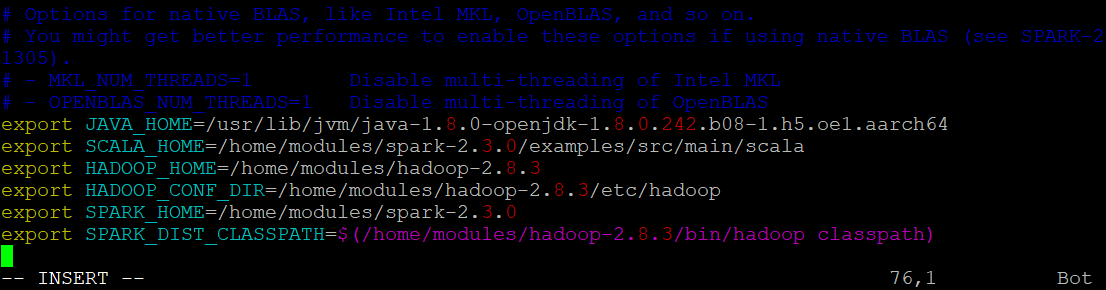
export SCALA\_HOME=/home/modules/spark-2.3.0/examples/src/main/scala

export HADOOP\_HOME=/home/modules/hadoop-2.8.3

export HADOOP\_CONF\_DIR=/home/modules/hadoop-2.8.3/etc/hadoop

export SPARK\_HOME=/home/modules/spark-2.3.0

export SPARK\_DIST\_CLASSPATH=$(/home/modules/hadoop-2.8.3/bin/hadoop classpath)



按ESC退出编辑模式，输入:wq保存并退出。

配置系统环境变量

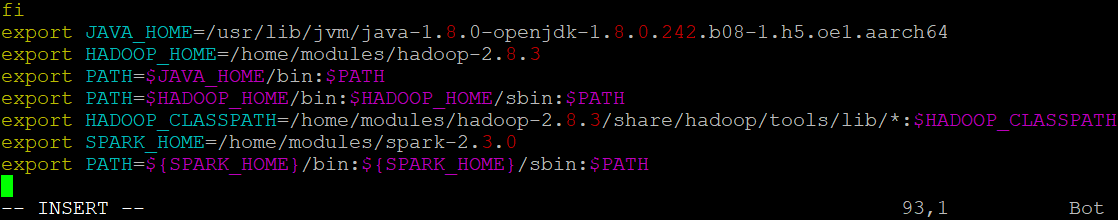
使用vim命令打开系统配置文件profile。

vim /etc/profile

按i键进入编辑模式，在文件尾添加如下内容。

export SPARK\_HOME=/home/modules/spark-2.3.0

export PATH=${SPARK\_HOME}/bin:${SPARK\_HOME}/sbin:$PATH



按ECS退出编辑，输入:wq保存并退出。

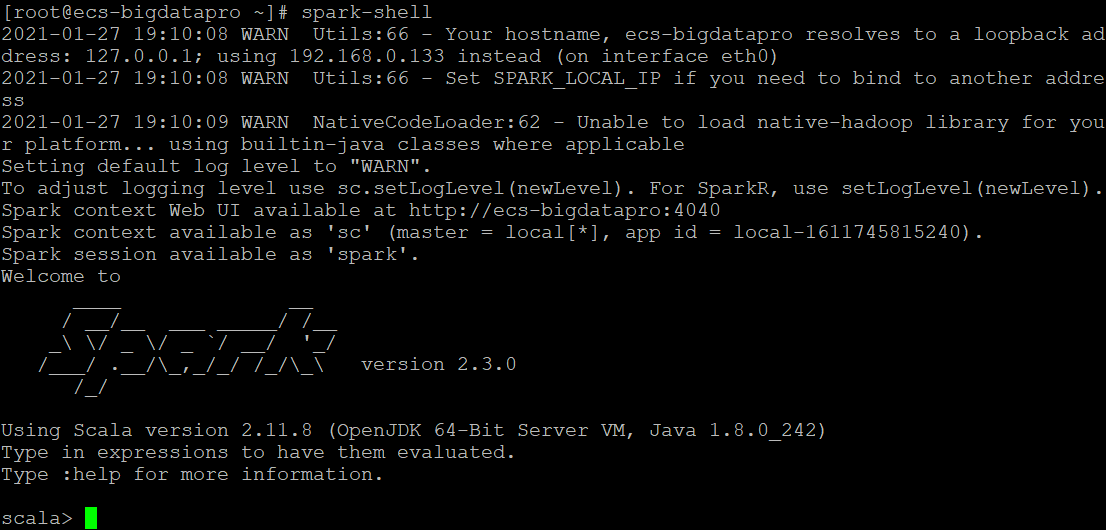
使用source命令使用环境变量生效。

source /etc/profile

验证Spark安装

进入root目录，输入spark-shell命令。

spark-shell



显示版本号并进入命令提示符，Spark安装完成。

输入:quit退出spark-shell。

:quit



### 验证存算分离

查看要计算的文件

本次实验数据使用实验3.3.2中上传的playerinfo.txt文件。

数据内容如下：

Alex James Lax Genu

Kerry Mary Olivia William

Hale Edith Vera Robert

Mary Olivia James Lax

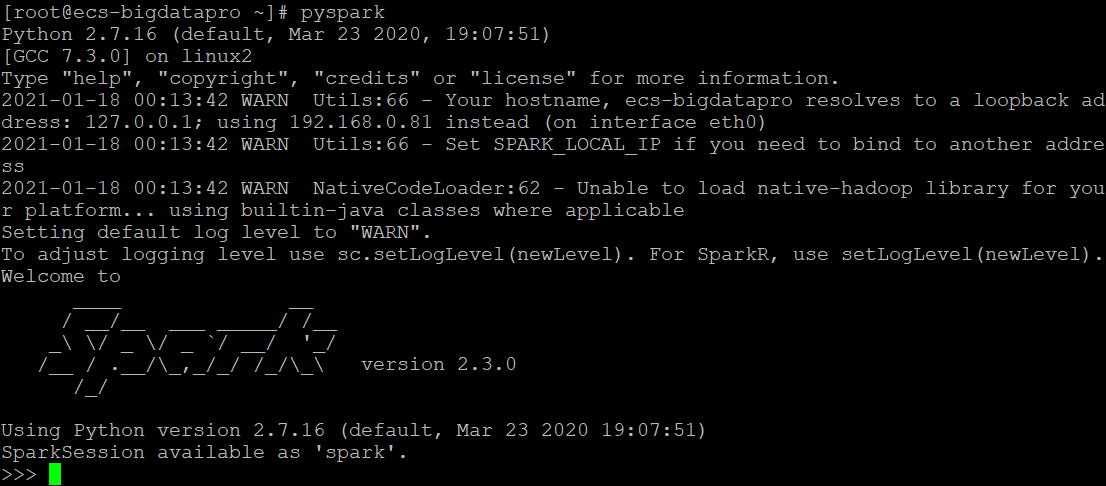
Edith Vera Robertm Genu

启动pyspark

Spark是Scala语言实现的，为了支持Python语言使用Spark，Apache Spark社区开发了一个工具PySpark。PySpark提供了PySpark Shell，它是一个结合了Python API和spark core的工具，同时能够初始化Spark环境。

直接输入**pyspark**回车即可启动。

pyspark



进入提示符“>>>”则启动成功。

验证存算分离

编写Spark程序，读取OBS中存储的数据，然后计算单词出现次数。在pyspark中依次输入以下代码。

**读取OBS内容（注意OBS桶的名称）：**

lines = spark.read.text("obs://**obs-bigdatapro**/").rdd.map(lambda r: r[0])



**进行统计：**

counts = lines.flatMap(lambda x: x.split(' ')).map(lambda x: (x, 1)).reduceByKey(lambda x, y: x + y)

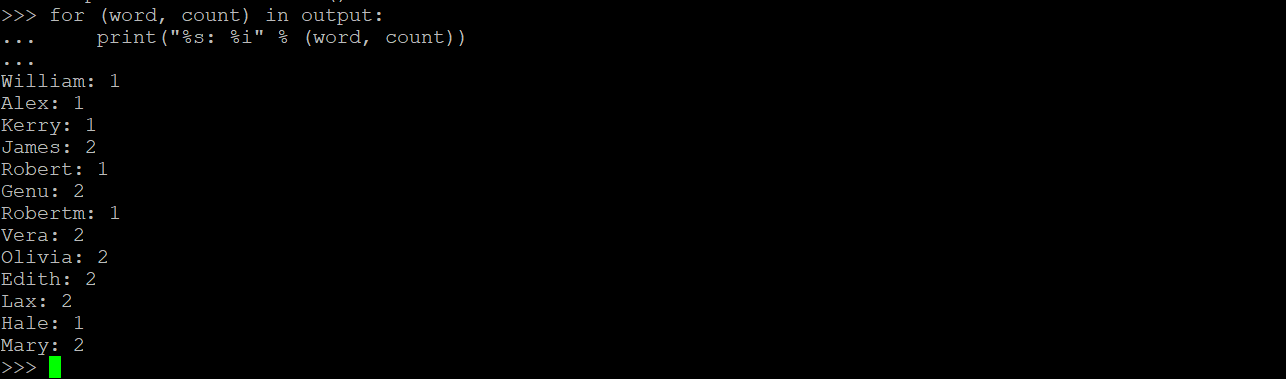
output = counts.collect()



**输出结果（注意输出语句print前面至少输入一个空格，Python使用缩进区分语句块，输入完毕需要再敲一次回车）：**

for (word, count) in output:

print("%s: %i" % (word, count))



存算分离验证成功！

退出pyspark

输入exit()退出pyspark。



说明：此处验证存算分离的Spark程序是在shell编写Python代码，如果想使用Python脚本的方式请参见附录七。

# 释放华为云服务

## 释放弹性云服务器ECS

在控制台进入弹性云服务管理页面，选中要删除的实例对应的复选框，然后点击上面的“更多”在弹出的菜单中点击“删除”。



在弹出的对话框中选中两个复选框，然后点击“是”即可删除ECS和相关资源。



## 删除OBS桶

在控制台进入OBS服务，选择“对象存储”，然后点击对应桶后面的“删除”（如果桶非空需要先点击桶名进入桶内删除所有的对象文件）。



弹出的对话框选择“是”即可删除桶。



# 附录 安装OpenJDK

安装OpenJDK

执行如下命令，将jdk安装包拷贝到安装目录下。

cd /root

cp OpenJDK8U-jdk\_aarch64\_linux\_hotspot\_8u191b12.tar.gz /usr/lib/jvm/

进入安装目录解压安装包。

cd /usr/lib/jvm/

tar zxvf OpenJDK8U-jdk\_aarch64\_linux\_hotspot\_8u191b12.tar.gz

在/etc/profile增加如下的配置

export JAVA\_HOME=/usr/lib/jvm/jdk8u191-b12

执行source命令使环境变量生效。

source /etc/profile

查看java安装版本

java -version

[root@ecs-bigdatapro ~]# java -version

openjdk version "1.8.0\_191"

OpenJDK Runtime Environment (AdoptOpenJDK)(build 1.8.0\_191-b12)

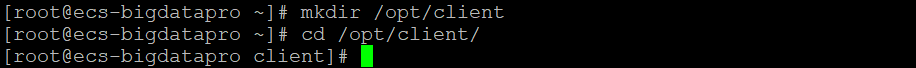
# 附录 存算分离脚本开发

编写Python脚本

新建目录 /opt/client 并进入。

mkdir /opt/client

cd /opt/client/



使用vi命令编写Python脚本：playercount.py。

vim playercount.py

复制下面的脚本代码，粘贴到PuTTY的vim中（在PuTTY中右键即为粘贴）。

# -\*- coding:utf-8 -\*-

# 创建SparkSession

from pyspark.sql.session import SparkSession

spark = SparkSession.builder.getOrCreate()

spark.sparkContext.setLogLevel("WARN")

# 读取OBS数据

lines = spark.read.text("obs://obs-bigdatapro/").rdd.map(lambda r: r[0])

# 统计单词出现次数

counts = lines.flatMap(lambda x: x.split(' ')).map(lambda x: (x, 1)).reduceByKey(lambda x, y: x + y)

output = counts.collect()

# 输出统计结果

for (word, count) in output:

print("%s: %i" % (word, count))

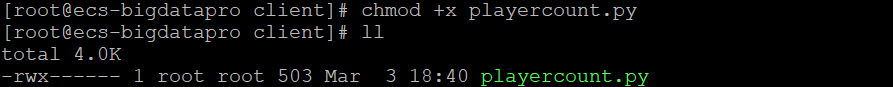
输入:wq保存并退出vim编辑器。

运行Python脚本

使用chmod命令给脚本文件赋予执行权限。

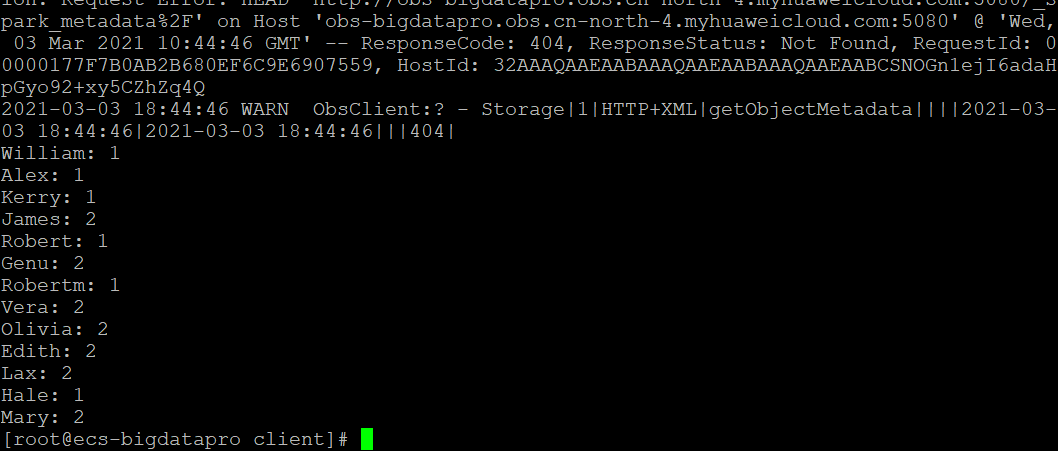
chmod +x playercount.py

ll



使用spark-submit执行脚本。

spark-submit playercount.py



得到计算结果。

说明：从Spark1.0.0开始，Spark提供了一个应用程序部署工具spark-submit（它在Spark的bin目录中），可以完成Spark应用程序在local、Standalone、YARN、Mesos上的快捷部署。

# 附录 Linux常用命令

## ls命令

ls 命令不仅可以查看 linux 文件夹包含的文件而且可以查看文件权限(包括目录、文件夹、文件权限)查看目录信息等等。

命令格式：ls [选项][目录名]

常用参数

-l ：列出长数据串，包含文件的属性与权限数据等

-a ：列出全部的文件，连同隐藏文件（开头为.的文件）一起列出来（常用）

-d ：仅列出目录本身，而不是列出目录的文件数据

-h ：将文件容量以较易读的方式（GB，kB等）列出来

-R ：连同子目录的内容一起列出（递归列出），等于该目录下的所有文件都会显示出来

## cd命令

最基本的命令语句，其他的命令语句要进行操作，都是建立在使用 cd 命令上的。用于切换当前目录至dirName。

命令格式：cd [目录名]

## pwd命令

查看"当前工作目录"的完整路径。

命令格式：pwd [选项]

常用参数：

-P :显示实际物理路径，而非使用连接（link）路径

-L :当目录为连接路径时，显示连接路径

## mkdir命令

用来创建指定的名称的目录，要求创建目录的用户在当前目录中具有写权限，并且指定的目录名不能是当前目录中已有的目录。

命令格式：mkdir [选项] 目录

常用参数

-m, --mode=模式，设定权限<模式> (类似 chmod)，而不是 rwxrwxrwx 减 umask

-p, --parents 可以是一个路径名称。此时若路径中的某些目录尚不存在,加上此选项后,系统将自动建立好那些尚不存在的目录,即一次可以建立多个目录;

-v, --verbose 每次创建新目录都显示信息

--help 显示此帮助信息并退出

--version 输出版本信息并退出

## rm命令

删除一个目录中的一个或多个文件或目录，如果没有使用- r选项，则rm不会删除目录。如果使用 rm 来删除文件，通常仍可以将该文件恢复原状。

命令格式：rm [选项] 文件

常用参数

-f, --force 忽略不存在的文件，从不给出提示。

-i, --interactive 进行交互式删除

-r, -R, --recursive 指示rm将参数中列出的全部目录和子目录均递归地删除。

-v, --verbose 详细显示进行的步骤

--help 显示此帮助信息并退出

--version 输出版本信息并退出

## rmdir命令

该命令从一个目录中删除一个或多个子目录项，删除某目录时也必须具有对父目录的写权限。

命令格式：rmdir [选项] 目录

常用参数

-p 递归删除目录dirname，当子目录删除后其父目录为空时，也一同被删除。如果整个路径被删除或者由于某种原因保留部分路径，则系统在标准输出上显示相应的信息。

-v, --verbose 显示指令执行过程

## mv命令

可以用来移动文件或者将文件改名（move (rename) files）。当第二个参数类型是文件时，mv命令完成文件重命名。当第二个参数是已存在的目录名称时，源文件或目录参数可以有多个，mv命令将各参数指定的源文件均移至目标目录中。

命令格式：mv [选项] 源文件或目录 目标文件或目录

常用参数

-b ：若需覆盖文件，则覆盖前先行备份

-f ：force 强制的意思，如果目标文件已经存在，不会询问而直接覆盖

-i ：若目标文件 (destination) 已经存在时，就会询问是否覆盖

-u ：若目标文件已经存在，且 source 比较新，才会更新(update)

-t ：--target-directory=DIRECTORY move all SOURCE arguments into DIRECTORY，即指定mv的目标目录，该选项适用于移动多个源文件到一个目录的情况，此时目标目录在前，源文件在后

## cp命令

将源文件复制至目标文件，或将多个源文件复制至目标目录。

命令格式：cp [选项] 源文件 目录 或 cp [选项] -t 目录 源文件

常用参数

-t --target-directory 指定目标目录

-i --interactive 覆盖前询问（使前面的 -n 选项失效）

-n --no-clobber 不要覆盖已存在的文件（使前面的 -i 选项失效）

-f --force 强行复制文件或目录，不论目的文件或目录是否已经存在

-u --update 使用这项参数之后，只会在源文件的修改时间较目的文件更新时，或是对应的目的文件并不存在，才复制文件

## cat 命令

用来显示文件内容，或者将几个文件连接起来显示，或者从标准输入读取内容并显示，它常与重定向符号配合使用。

命令格式：cat [选项] [文件]

常用参数

-A, --show-all 等价于 -vET

-b, --number-nonblank 对非空输出行编号

-e 等价于 -vE

-E, --show-ends 在每行结束处显示 $

-n, --number 对输出的所有行编号,由1开始对所有输出的行数编号

-s, --squeeze-blank 有连续两行以上的空白行，就代换为一行的空白行

-t 与 -vT 等价

-T, --show-tabs 将跳格字符显示为 ^I

-v, --show-nonprinting 使用 ^ 和 M- 引用，除了 LFD 和 TAB 之外

## more命令

more 命令和 cat 的功能一样都是查看文件里的内容，但有所不同的是more可以按页来查看文件的内容，还支持直接跳转行等功能。

命令格式：more [-dlfpcsu ] [-num ] [+/ pattern] [+ linenum] [file ... ]

常用参数

+n 从笫n行开始显示

-n 定义屏幕大小为n行

+/pattern 在每个档案显示前搜寻该字串（pattern），然后从该字串前两行之后开始显示

-c 从顶部清屏，然后显示

-d 提示“Press space to continue，’q’ to quit（按空格键继续，按q键退出）”，禁用响铃功能

-l 忽略Ctrl+l（换页）字符

-p 通过清除窗口而不是滚屏来对文件进行换页，与-c选项相似

-s 把连续的多个空行显示为一行

-u 把文件内容中的下画线去掉

操作指令

Enter：向下n行，需要定义。默认为1行

Ctrl+F：向下滚动一屏

空格键：向下滚动一屏

Ctrl+B：返回上一屏

= ：输出当前行的行号

：f ：输出文件名和当前行的行号

V ：调用vi编辑器

!命令 ：调用Shell，并执行命令

q ：退出more

## less命令

less 与 more 类似，但使用 less 可以随意浏览文件，而 more 仅能向前移动，却不能向后移动，而且 less 在查看之前不会加载整个文件。

命令格式：less [参数] 文件

常用参数

-b <缓冲区大小> 设置缓冲区的大小

-e 当文件显示结束后，自动离开

-f 强迫打开特殊文件，例如外围设备代号、目录和二进制文件

-g 只标志最后搜索的关键词

-i 忽略搜索时的大小写

-m 显示类似more命令的百分比

-N 显示每行的行号

-o <文件名> 将less 输出的内容在指定文件中保存起来

-Q 不使用警告音

-s 显示连续空行为一行

-S 行过长时间将超出部分舍弃

-x <数字> 将“tab”键显示为规定的数字空格

操作命令

/字符串：向下搜索“字符串”的功能

?字符串：向上搜索“字符串”的功能

n：重复前一个搜索（与 / 或 ? 有关）

N：反向重复前一个搜索（与 / 或 ? 有关）

b 向后翻一页

d 向后翻半页

h 显示帮助界面

Q 退出less 命令

u 向前滚动半页

y 向前滚动一行

空格键 滚动一行

回车键 滚动一页

[pagedown]：向下翻动一页

[pageup]：向上翻动一页

## head命令

head 用来显示档案的开头至标准输出中，默认 head 命令打印其相应文件的开头 10 行。

命令格式：head [参数] [文件]

常用参数

-q 隐藏文件名

-v 显示文件名

-c<字节> 显示字节数

-n<行数> 显示的行数

## tail命令

显示指定文件末尾内容，不指定文件时，作为输入信息进行处理。常用查看日志文件。

命令格式：tail [必要参数] [选择参数] [文件]

常用参数

-f 循环读取

-q 不显示处理信息

-v 显示详细的处理信息

-c<数目> 显示的字节数

-n<行数> 显示行数

--pid=PID 与-f合用,表示在进程ID,PID死掉之后结束.

-q, --quiet, --silent 从不输出给出文件名的首部

-s, --sleep-interval=S 与-f合用,表示在每次反复的间隔休眠S秒

## vim命令

vim编辑器是所有Unix及Linux系统下标准的编辑器，它的强大不逊色于任何最新的文本编辑器;vi也是Linux中最基本的文本编辑器,vim就是vi的升级版。

### 启动vim

在命令行窗口中输入以下命令即可

vim

直接启动vim

vim filename 打开vim并创建名为filename的文件

### vim的模式

正常模式（按Esc或Ctrl+[进入） 左下角显示文件名或为空

插入模式（按i键进入） 左下角显示--INSERT--

可视模式（不知道如何进入） 左下角显示--VISUAL--

导航命令

% 括号匹配

### 插入命令

i 在当前位置生前插入

I 在当前行首插入

a 在当前位置后插入

### 查找命令

/text　查找text，按n健查找下一个，按N健查找前一个。

?text　查找text，反向查找，按n健查找下一个，按N健查找前一个。

:set hlsearch　高亮搜索结果，所有结果都高亮显示，而不是只显示一个匹配。

:set nohlsearch　关闭高亮搜索显示

:nohlsearch　关闭当前的高亮显示，如果再次搜索或者按下n或N键，则会再次高亮。

:set incsearch　逐步搜索模式，对当前键入的字符进行搜索而不必等待键入完成。

:set wrapscan　重新搜索，在搜索到文件头或尾时，返回继续搜索，默认开启。

:set nu 显示行号

### 替换命令

ra 将当前字符替换为a，当期字符即光标所在字符。

s/old/new/ 用old替换new，替换当前行的第一个匹配

### 撤销和重做

u 撤销（Undo）

U 撤销对整行的操作

Ctrl + r 重做（Redo），即撤销的撤销。

### 删除命令

x 删除当前字符

3x 删除当前光标开始向后三个字符

X 删除当前字符的前一个字符。X=dh

dl 删除当前字符， dl=x

dh 删除前一个字符

dd 删除当前行

10d 删除当前行开始的10行。

D 删除当前字符至行尾。D=d$

d$ 删除当前字符之后的所有字符（本行）

### 拷贝和粘贴

yy 拷贝当前行

nyy 拷贝当前后开始的n行，比如2yy拷贝当前行及其下一行。

p 在当前光标后粘贴,如果之前使用了yy命令来复制一行，那么就在当前行的下一行粘贴。

shift+p 在当前行前粘贴

### 退出命令

:wq 保存并退出

ZZ 保存并退出

:q! 强制退出并忽略所有更改

:e! 放弃所有修改，并打开原来文件。

### 帮助命令

:help or F1 显示整个帮助

:help xxx 显示xxx的帮助，比如 :help i, :help CTRL-[（即Ctrl+[的帮助）。

:help 'number' Vim选项的帮助用单引号括起

:help 特殊键的帮助用<>扩起

:help -t Vim启动参数的帮助用-

：help i\_ 插入模式下Esc的帮助，某个模式下的帮助用模式\_主题的模式

帮助文件中位于||之间的内容是超链接，可以用Ctrl+]进入链接，Ctrl+o（Ctrl + t）返回

其他非编辑命令