**哈尔滨工业大学（深圳）**

**大数据实验指导书**

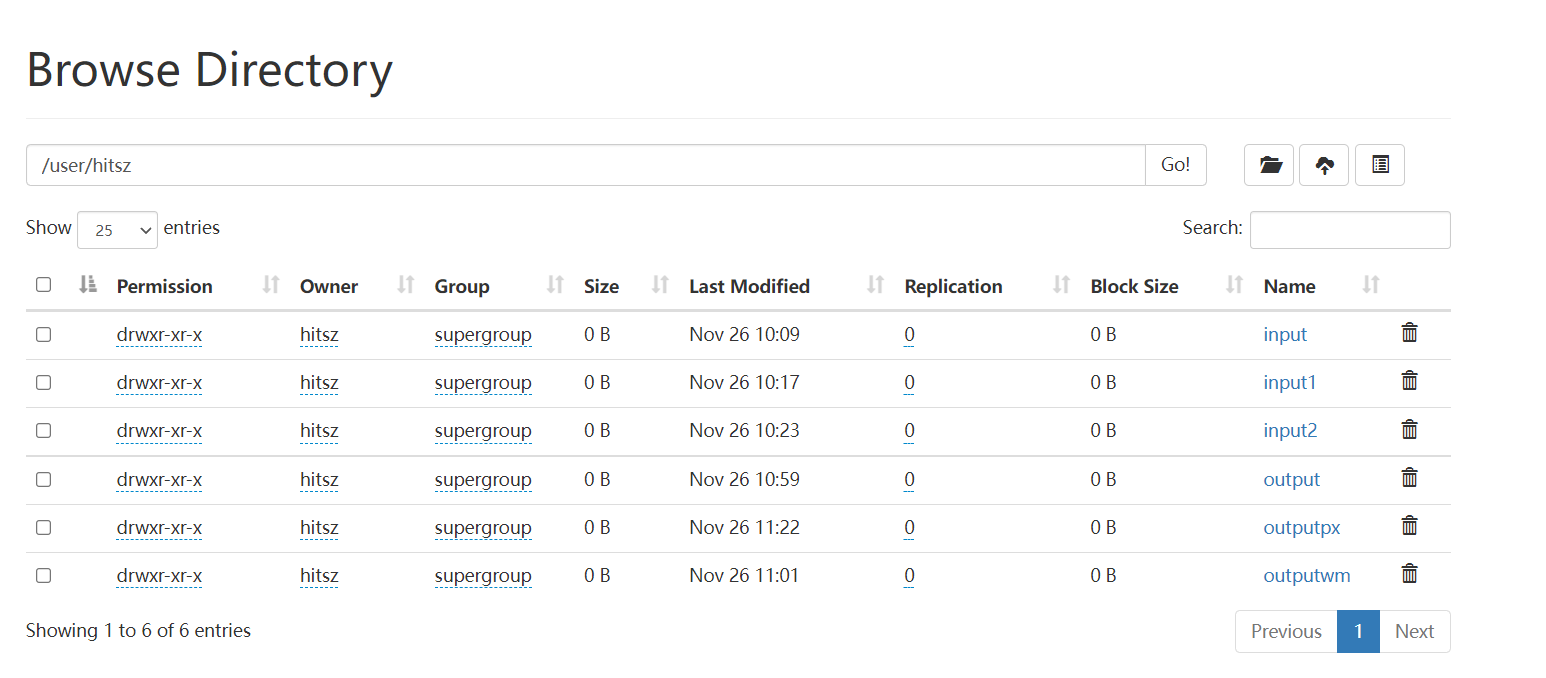
实验一 Hadoop环境配置与基本操作

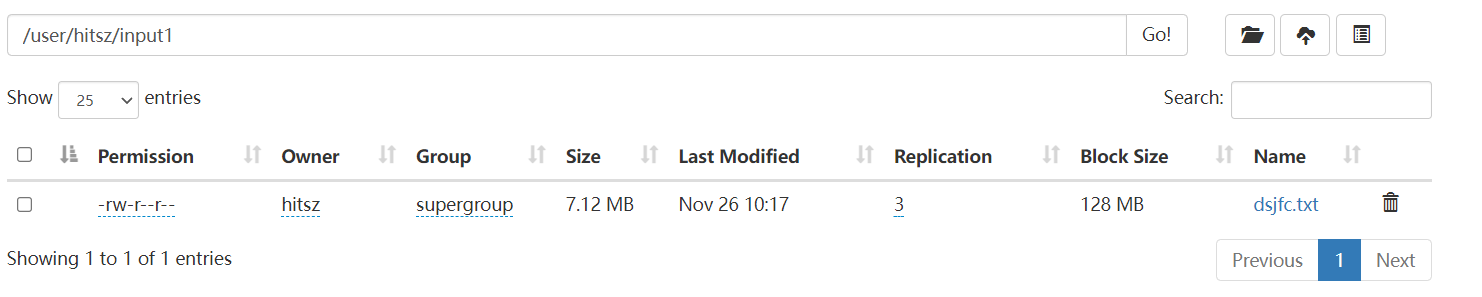
# 实验目的

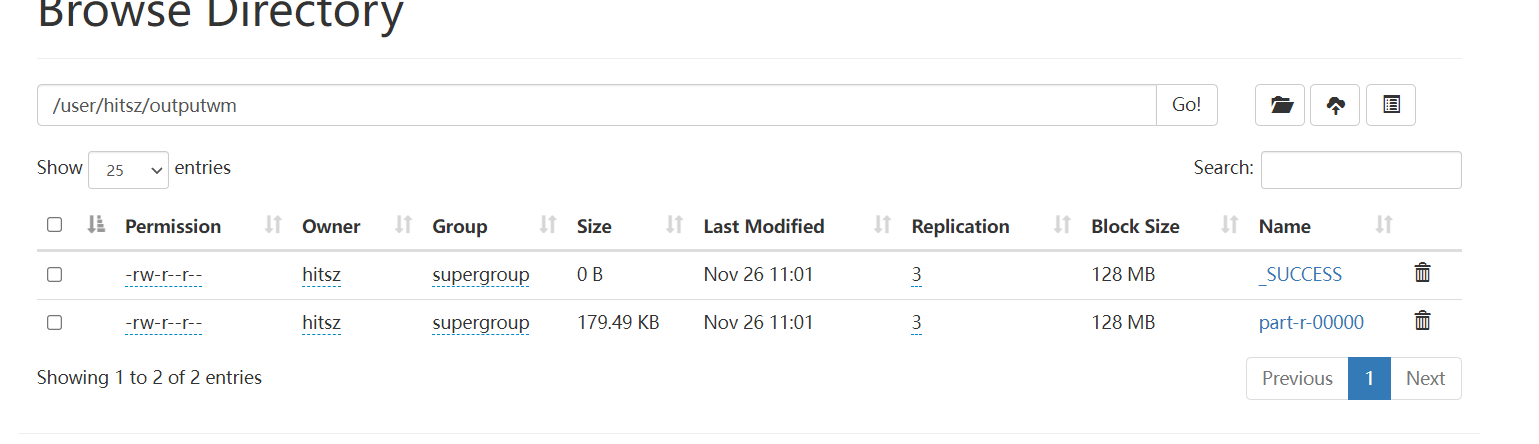
1. 熟悉Hadoop分布式配置的方法。
2. 熟悉命令行运行Mapreduce作业的原理和操作。

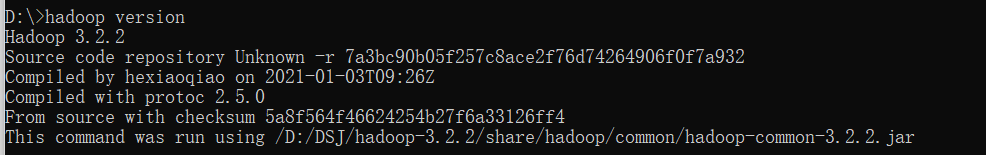
# 实验内容

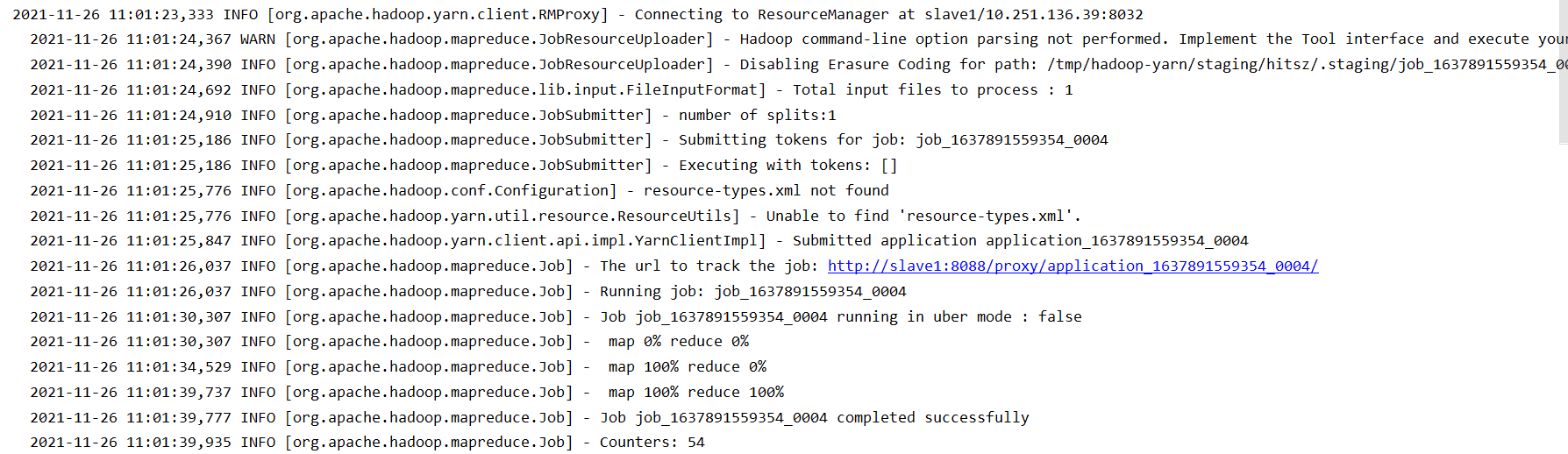
1. 根据实验指导书，小组合作搭建Hadoop分布式环境并截取WebUI界面（HDFS界面DataNode页面，该页面包含了IP地址，实验报告必须包含此截图）。
2. 配置Hadoop客户端（自己电脑），实现文本文件（爬虫得到的文件）的上传（此处可以截图命令行）。
3. 为WordCount项目添加集群运行配置，运行并截图（可以截取HDFS WebUI,jobhistory webUI, yarn WebUI, IDEA控制台，wordcount输出文本文件等）。

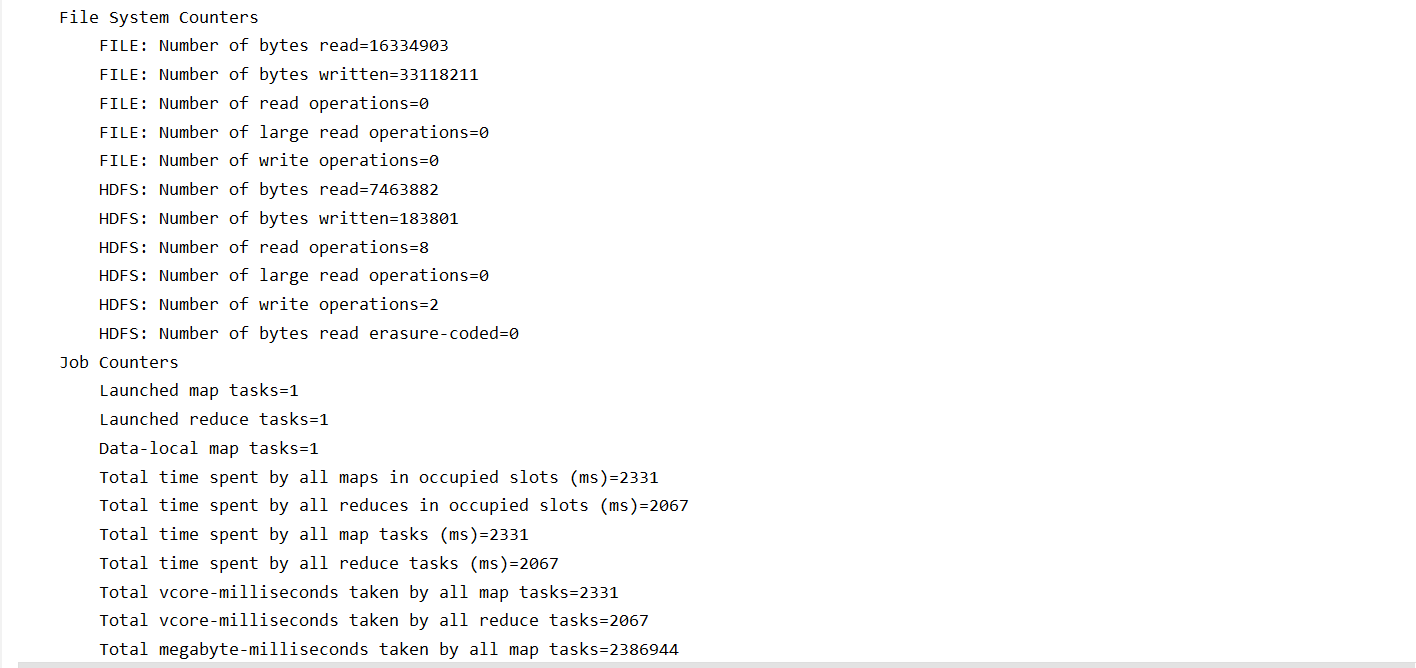


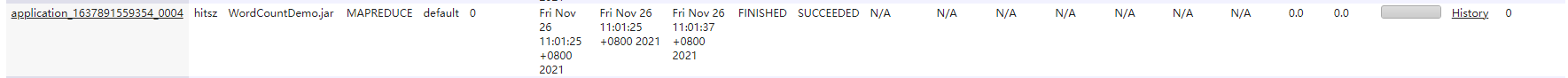




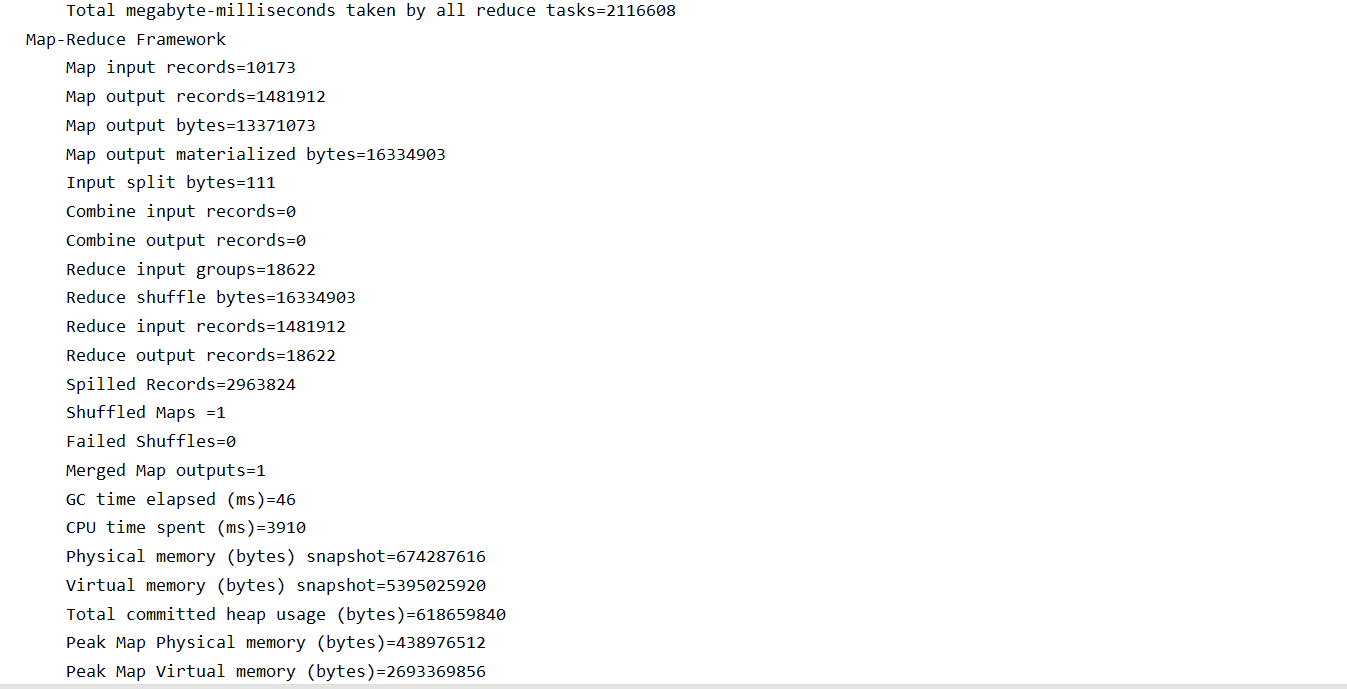


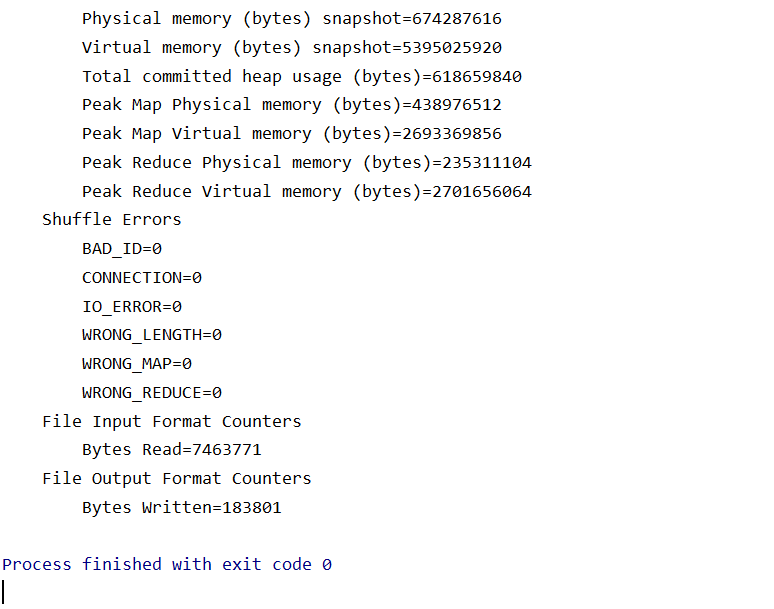












**2，3部分每个人根据自己作业二的内容进行完成。**

由于机房环境重启会还原系统，如实验课上未完成集群的搭建，可以课后自己小组用个人电脑搭建；如实验课上已完成集群的搭建但是尚未完成2，3部分，请向助教申请集群，或者自行搭建，不影响实验评分。

# 实验环境

* 1. Ubuntu 18.04
  2. Hadoop 3.2.2
  3. Java 1.8

# 实验步骤

真实分布式的Hadoop集群搭建过程较为复杂，请同学们仔细阅读该章节，注意突出显示的文本。

在本实验指导书中，Hadoop集群计算机预设如表4-1所示，集群包含3台计算机，在此基础上实现分布式Hadoop集群的搭建。

Hadoop集群的配置过程中有许多部分可以进行自定义，但是首次配置建议尽量与本指导书保持一致。

表 4-1 教程用Hadoop集群计算机资源概要

|  |  |  |
| --- | --- | --- |
| 编号 | 计算机名 | IP地址 |
| 678-21 | hitsz-master | 10.248.5.21 |
| 678-22 | hitsz-slave1 | 10.248.5.22 |
| 678-23 | hitsz-slave2 | 10.248.5.23 |

IP地址每台机器都不同，请根据实际情况进行修改。

配置过程中涉及到多条linux命令行操作，示例如下：

文本

描述已自动生成

注意事项

* + - 1. 不要用自己的虚拟机或mac来建集群。
      2. 注意命令行用户名，lenovo 这个用户无论什么时候都不能用，如果重开终端，输入 su – hitsz 或su – root切换到hitsz或者root用户。
      3. 客户端JDK不能高于1.8（Java8），建议安装了高版本JDK如11,15,17的同学再安一下1.8版本（群里有安装包和指导操作）或者借其他同学的客户端完成2，3部分。

## 4.1 预下载文件与准备工作

本次实验需要用到的文件有：

1. Hadoop3.22 压缩包
2. Jdk1.8.0\_202 压缩包
3. 实验指导书
4. xsync脚本
5. 用于wordcount的文本文件

（1）-（4）会预先提供，请同学们将（1）-（4）文件拷贝到U盘上，实验时需拷贝到集群计算机上。点击左侧文件管理器的图标，即可进行拷贝，建议直接拷贝到主目录下的文件夹，方便查找。

注意：由于操作系统为Ubuntu 16.04, 请使用格式为FAT32的U盘，请勿使用其他格式的U盘或者默认为exfat格式的移动硬盘。否则系统将不能识别该存储设备。

此外，需要携带完成了作业二的计算机，存储有（5）用于wordcount的文本文件，用于完成2，3部分。

## 4.2 登入网络

【以下操作三台计算机均要进行】

点击左上方的火狐浏览器，键入校园网网址 10.248.98.2 ，输入校园网用户名密码完成机器的联网。打开百度，确认联网成功。

## 4.3 基础配置

【以下操作三台计算机均要进行】

1. 切换到root用户，需要键入root用户的密码（密码保存在计算机桌面上）。

lenovo@687-21:~$ **su root**

1. apt更新。

root@687-21:/home/Lenovo/# **apt-get update**

1. 查看防火墙状态。

root@687-21:/home/Lenovo/# **ufw status**

如果防火墙已经开启，关闭防火墙。防火墙会阻碍Hadoop集群内部的通信，在实际生产环境中，单个服务器的防火墙会关闭，通过设置集群整体对外防火墙来保证安全。

root@687-21:/home/Lenovo/# **ufw disable**

1. 修改主机名。此步并非必须，但是为了更好的区分各台机器，修改计算机主机名。

将第一台机器的用户名修改为hitsz-master；第二台机器修改为hitsz-slave1；第三台机器修改为hitsz-slave2。

root@687-21:/home/Lenovo/# **hostnamectl set-hostname hitsz-master**

1. 安装vim。

root@hitsz-master:/home/Lenovo/# **apt-get install vim**

如果遇到依赖缺失问题，请按照提示键入：

root@hitsz-master:/home/Lenovo/# **apt --fix-broken install**

再重新安装vim。

root@hitsz-master:/home/Lenovo/# **apt-get install vim**

## 4.4 创建用户

【以下操作三台计算机均要进行】

1. 新增用户hitsz，提示输入新的UNIX密码时需要键入hitsz的密码（如1234），请记住自己设置的密码。其余项一律默认即可。

root@hitsz-master:/home/Lenovo/# **adduser hitsz**

文本

描述已自动生成

1. 添加用户到sudo用户组使其具有sudo权限。

root@hitsz-master:/home/Lenovo/# **adduser hitsz sudo**

注意：为保证部署的顺利进行，降低复杂度，在本教程中，三台机器的新用户名需保持一致，如均为hitsz。

## 4.5 配置Hosts

Hosts是指域名和IP地址的映射，通过配置Hosts，能够将复杂难记的IP地址转换成简单易记的域名，方便配置。在此教程中，将第一台机器hitsz-master的IP映射为master , 将第二台机器hitsz-slave1的IP映射为slave1 ，第三台机器hitsz-slave2的IP映射为slave2。

【以下操作三台计算机均要进行】

1. 查看各台机器的IP地址

root@hitsz-master:/home/Lenovo/# **ifconfig**

文本

描述已自动生成

如果ifconfig命令不存在，需要先安装net-tools。

root@hitsz-master:/home/Lenovo/# **apt-get install net-tools**

收集到三台计算器的IP地址如下：

|  |  |  |
| --- | --- | --- |
| 计算机名 | IP地址 | 自定义域名 |
| hitsz-master | 10.248.5.21 | master |
| hitsz-slave1 | 10.248.5.22 | slave1 |
| hitsz-slave2 | 10.248.5.23 | Slave2 |

1. 修改host文件， 使用vim 进入编辑。

注意：此操作至关重要，请仔细检查，确保IP地址键入的正确性。

root@hitsz-master:/home/Lenovo/# **vim /etc/hosts**

在vim界面按 i 进入编辑，在第一行localhost下键入以下内容，注意每行中间使用的是Tab键进行分隔：

10.248.5.21 master

10.248.5.22 slave1

10.248.5.23 slave2

IP地址每台机器都不同，请根据实际情况进行修改。

最终效果如下：

文本

描述已自动生成

按Esc 键退出编辑，输入 :wq 命令保存退出。

1. 重启网络使hosts文件生效

root@hitsz-master:/home/Lenovo/# **/etc/init.d/networking restart**

重启需稍等片刻，直至看到ok。

## 4.6 安装SSH并配置SSH免密

【以下操作三台计算机均要进行】

1. 切换至新用户hitsz。在su命令后加“-”符号可以同时变更工作目录。

root@hitsz-master:/home/Lenovo/# **su - hitsz**

1. 安装ssh服务器。此处需要键入hitsz的用户密码。

hitsz@hitsz-master:~$ **sudo apt-get install openssh-server**

如果出现资源锁定的情况，联系助教，解决方案不在此赘述。

1. 修改ssh配置文件（注意是sshd\_config，不是ssh\_config）

(3)(4)一定要做，不然就是各种Permission Denied

hitsz@hitsz-master:~$ **sudo vim /etc/ssh/sshd\_config**

找到StrictModes项，按i键进入编辑，删除#号取消注释，并将属性值由yes修改为no。

找到PermitRootLogin 项，取消注释，并将属性值由prohibit-password修改为yes。

如果找不到，直接在任意位置添加以下内容：

PermitRootLogin yes

StrictModes no

按Esc退出编辑模式，输入 :wq 保存退出。

1. 使ssh配置文件修改生效（每次修改ssh配置文件后都需要重新执行该命令）

hitsz@hitsz-master:~$ **sudo service sshd reload**

1. 免密登录的原理

图片包含 图示

描述已自动生成

下面以hitsz-master为例演示ssh免密登录的操作过程。ssh免密配置不是为了hadoop集群间的数据通信，而是为了传输指令。Hadoop集群的主节点通过ssh协议控制从节点的启停。如果不配置ssh免密登录，则需要频繁地输入密码。配置免密登录也有利于后续的集群分发过程。

1. 生成密钥对

hitsz@hitsz-master:~$ **ssh-keygen -t rsa**

然后敲（三个回车），就会生成两个文件id\_rsa（私钥）、id\_rsa.pub（公钥）,该密钥属于hitsz-master下的hitsz用户，存储在/home/hitsz/.ssh目录下。

文本

描述已自动生成

1. 使用ssh-copy-id将公钥拷贝到要免密登录的目标机器上，对于hitsz-master来说，需要配置到hitsz-slave1, hitsz-slave2以及自身的免密登录。该免密登录的用户名均为hitsz。

请注意区分主机名和域名，此处应该使用域名如slave1！！！！！！！！！！！！

hitsz@hitsz-master:~$ **ssh-copy-id slave1**

hitsz@hitsz-master:~$ **ssh-copy-id slave2**

hitsz@hitsz-master:~$ **ssh-copy-id master**

每一次拷贝都需要输入一次yes，并在之后输入一次目标机器hitsz用户的密码。

文本

描述已自动生成

注意：

还需要在hitsz-slave1机器上配置到hitsz-master、hitsz-slave2以及自身hitsz-slave1的免密登录。

还需要在hitsz-slave2机器上配置到hitsz-master、hitsz-slave1以及自身hitsz-slave2的免密登录。

【以下（8）-（11）操作仅需在hitsz-master上执行】

额外的还需要配置root用户下，hitsz-master到hitsz-slave1、hitsz-slave2以及自身hitsz-master的免密登录。

1. 切换到root用户

hitsz@hitsz-master:~$ **su - root**

1. 生成密钥对

root@hitsz-master:~# **ssh-keygen -t rsa**

然后敲（三个回车），同样会生成两个文件id\_rsa（私钥）、id\_rsa.pub（公钥）,该密钥属于hitsz-master下的root用户，存储在/root/.ssh目录下。

1. 配置免密登录

root@hitsz-master:~# **ssh-copy-id slave1**

root@hitsz-master:~# **ssh-copy-id slave2**

root@hitsz-master:~# **ssh-copy-id master**

每一次拷贝都需要输入一次yes，并在之后输入一次目标机器root用户的密码（user1234）。

1. ssh免密登录验证

可以使用ssh登录来验证配置是否成功。通过命令行@后接主机名可以快速判断是否登录成功。

root@hitsz-master:~# **ssh slave1**

注意：验证后一定要输入exit 注销ssh登录。

假设当前已经ssh免密登录到hitsz-slave1上

root@hitsz-slave1:~# **exit**

## 4.7 创建目录

【以下操作三台计算机均要进行】

1. 切换到hitsz用户

root@hitsz-master:~# **su - hitsz**

1. 创建目录/opt/module和/opt/software，前者是软件的安装目录，后者用来存放杂项文件。

hitsz@hitsz-master:~$ **sudo mkdir /opt/module**

hitsz@hitsz-master:~$ **sudo mkdir /opt/software**

1. 将新目录所属的用户和用户组均改为hitsz

hitsz@hitsz-master:~$ **sudo chown hitsz:hitsz /opt/module**

hitsz@hitsz-master:~$ **sudo chown hitsz:hitsz /opt/software**

1. 可以移动到opt目录下使用ll命令查看是否修改成功

hitsz@hitsz-master:~$ **cd /opt**

hitsz@hitsz-master:/opt$ **ll**

## 4.8 安装jdk

【以下操作在hitsz-master上进行】

1. 在文件管理器中找到拷贝到计算机上的jdk安装包（最好不要在中文路径下，因为机器没有安装输入法，可在文件管理器中将安装包移动到主目录下），右键，点击属性即可查看该文件的实际位置。

图形用户界面, 应用程序

描述已自动生成

1. 将该压缩包移动到hitsz用户的目录下，比如可以移动到/opt/software，jdk安装包的名称复杂，可以使用Tab键进行自动补全。假设安装包完整位置为/home/lenovo/software/

hitsz@hitsz-master:/opt$ **sudo mv /home/lenovo/software/jdk-8u202-linux-x64.tar.gz /opt/software**

1. 进入移动后的目录，解压jdk到/opt/module

hitsz@hitsz-master:/opt$ **cd /opt/software**

hitsz@hitsz-master:/opt/software $ **tar -zxvf jdk-8u202-linux-x64.tar.gz -C /opt/module**

1. 进入解压后的目录，重命名文件夹为jdk18。该名字可以自定义，此举的目的主要是简化文件夹名称。

hitsz@hitsz-master:/opt/software$ **cd /opt/module**

hitsz@hitsz-master:/opt/module$ **mv jdk1.8.\_202 jdk18**

1. 编辑全局环境变量文件，添加java环境变量

hitsz@hitsz-master:/opt/module$ **sudo vim /etc/profile.d/my\_env.sh**

按下i进入编辑模式，添加以下内容：

#JAVA\_HOME

export JAVA\_HOME=/opt/module/jdk18

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

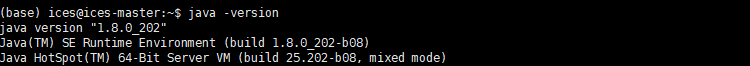
按下Esc后输入 :wq 保存退出。

1. 使环境变量修改生效

hitsz@hitsz-master:/opt/module$ **source /etc/profile**

1. 验证是否配置成功

hitsz@hitsz-master:/opt/module$ **java -version**



出现版本号1.8.0\_202即为正确。

## 4.9 安装hadoop

【以下操作在hitsz-master上进行】

以下步骤与4.8 非常相似。因此不过多赘述。

1. 在文件管理器中找到拷贝到计算机上的hadoop安装包，右键，点击属性即可查看该文件的实际位置。在该教程中，完整位置为/home/lenovo/software/
2. 将该压缩包移动到hitsz用户的目录下，比如可以移动到/opt/software。

hitsz@hitsz-master:/opt$ **sudo mv /home/lenovo/software/hadoop-3.2.2.tar.gz /opt/software**

1. 进入移动后的目录，解压hadoop到/opt/module

hitsz@hitsz-master:/opt$ **cd /opt/software**

hitsz@hitsz-master:/opt/software $ **tar -zxvf hadoop-3.2.2.tar.gz -C /opt/module**

1. 进入解压后的目录，重命名文件夹为hadoop。该名字可以自定义，此举的目的主要是简化文件夹名称。

hitsz@hitsz-master:/opt/software$ **cd /opt/module**

hitsz@hitsz-master:/opt/module$ **mv hadoop-3.2.2 hadoop**

1. 编辑全局环境变量文件，添加hadoop环境变量

hitsz@hitsz-master:/opt/module$ **sudo vim /etc/profile.d/my\_env.sh**

按下i进入编辑模式，在java的环境变量下方添加以下内容：

#HADOOP\_HOME

export HADOOP\_HOME=/opt/module/hadoop

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

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

按下Esc后输入 :wq 保存退出。

1. 使环境变量修改生效

hitsz@hitsz-master:/opt/module$ **source /etc/profile**

1. 验证是否配置成功，注意version前面没有“-”。

hitsz@hitsz-master:/opt/module$ **hadoop version**

文本

描述已自动生成

出现以上内容即为安装成功。

## 4.10 hadoop配置

【以下操作在hitsz-master上进行】

1. 集群规划

一个典型的集群规划如下：

|  |  |  |  |
| --- | --- | --- | --- |
|  | hitsz-master | hitsz-slave1 | hitsz-slave2 |
| HDFS | NameNode  DataNode | DataNode | SecondaryNameNode  DataNode |
| YARN | NodeManager | ResourceManager  NodeManager | NodeManager |

表格中的值代表机器在集群中需要承担的角色。

1. 修改配置文件

进入到配置文件所在位置，通过环境变量可以快速的定位hadoop安装目录。

hitsz@hitsz-master:/opt/module$ **cd $HADOOP\_HOME/etc/hadoop**

**5个配置文件都不是空文件，如果有空的，那就是文件名敲错了**

**前4个配置文件都要先删除内容，再粘贴，不是直接加在后面**

* + - 1. **修改核心配置文件**

hitsz@hitsz-master:/opt/module/Hadoop/etc/hadoop$ **vim core-site.xml**

文件内容如下，进入编辑模式删除原有内容在粘贴以下内容即可。Xml文件无需对齐，但需要保证标签的完整性，请务必复制完整。由于排版原因，指导书中的一个配置文件可能出现分页的情况，请务必注意。

注意配置文件中的master、slave1等名称均是指4.5中配置的域名。

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<!-- 指定NameNode的地址 -->

<property>

<name>fs.defaultFS</name>

<value>hdfs://master:8020</value>

</property>

<!-- 指定hadoop数据的存储目录 -->

<property>

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

<value>/opt/module/hadoop/data</value>

</property>

<!-- 配置HDFS网页登录使用的静态用户为hitsz -->

<property>

<name>hadoop.http.staticuser.user</name>

<value>hitsz</value>

</property>

</configuration>

* + - 1. **修改HDFS配置文件**

hitsz@hitsz-master:/opt/module/Hadoop/etc/hadoop$ **vim hdfs-site.xml**

文件内容如下：

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<!-- nn web端访问地址-->

<property>

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

<value>master:9870</value>

</property>

<!-- 2nn web端访问地址-->

<property>

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

<value>slave2:9868</value>

</property>

</configuration>

* + - 1. **修改yarn配置文件**

hitsz@hitsz-master:/opt/module/Hadoop/etc/hadoop$ **vim yarn-site.xml**

文件内容如下：

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<!-- 指定MR走shuffle -->

<property>

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

<value>mapreduce\_shuffle</value>

</property>

<!-- 指定ResourceManager的地址-->

<property>

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

<value>slave1</value>

</property>

<!-- 环境变量的继承 -->

<property>

<name>yarn.nodemanager.env-whitelist</name>

<value>JAVA\_HOME,HADOOP\_COMMON\_HOME,HADOOP\_HDFS\_HOME,HADOOP\_CONF\_DIR,CLASSPATH\_PREPEND\_DISTCACHE,HADOOP\_YARN\_HOME,HADOOP\_MAPRED\_HOME</value>

</property>

<!-- 开启日志聚集功能 -->

<property>

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

<value>true</value>

</property>

<!-- 设置日志聚集服务器地址 -->

<property>

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

<value>http://hadoop102:19888/jobhistory/logs</value>

</property>

<!-- 设置日志保留时间为7天 -->

<property>

<name>yarn.log-aggregation.retain-seconds</name>

<value>604800</value>

</property>

</configuration>

* + - 1. **修改MapReduce配置文件**

hitsz@hitsz-master:/opt/module/Hadoop/etc/hadoop$ **vim mapred-site.xml**

文件内容如下：

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<!-- 指定MapReduce程序运行在Yarn上 -->

<property>

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

<value>yarn</value>

</property>

<!-- 历史服务器端地址 -->

<property>

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

<value>master:10020</value>

</property>

<!-- 历史服务器web端地址 -->

<property>

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

<value>master:19888</value>

</property>

</configuration>

* + - 1. **修改Worker文件，配置DataNode节点**

hitsz@hitsz-master:/opt/module/Hadoop/etc/hadoop$ **vim workers**

**删除原有的localhost，加入以下内容**：

注意：该文件中添加的内容结尾不允许有空格，文件中不允许有空行。

master

slave1

slave2

* + - 1. **修改hadoop-env.sh**

hitsz@hitsz-master:/opt/module/Hadoop/etc/hadoop$ **vim hadoop-env.sh**

在任意位置加入以下内容，即再次显示声明JAVA\_HOME：

export JAVA\_HOME=/opt/module/jdk18

## 4.11 hadoop集群分发

【以下操作在hitsz-master上进行】

1. 在文件管理器中找到拷贝到计算机上的xsync脚本，右键，点击属性即可查看该文件的实际位置。在该教程中，完整位置为/home/ices/software/。
2. 将该脚本移动到/bin/目录下

hitsz@hitsz-master:/opt/module/Hadoop/etc/hadoop$ **sudo mv /home/ices/software/xsync /bin/**

1. 如果你自定义的域名与本教程不同，则需要修改xsync文件（红色字体部分），将域名进行替换。如果与本教程一致，则无需修改。

Xsync文件内容如下：

#!/bin/bash

#1. 判断参数个数

if [ $# -lt 1 ]

then

echo Not Enough Arguement!

exit;

fi

#2. 遍历集群所有机器

for host in master slave1 slave2

do

echo ==================== $host ====================

#3. 遍历所有目录，挨个发送

for file in $@

do

#4. 判断文件是否存在

if [ -e $file ]

then

#5. 获取父目录

pdir=$(cd -P $(dirname $file); pwd)

#6. 获取当前文件的名称

fname=$(basename $file)

ssh $host "mkdir -p $pdir"

rsync -av $pdir/$fname $host:$pdir

else

echo $file does not exists!

fi

done

done

1. 赋予xsync执行权限

hitsz@hitsz-master:/bin$ **cd /bin/**

hitsz@hitsz-master:/bin$ **sudo chmod +x xsync**

1. 分发jdk和hadoop安装文件

直接同步整个/opt/module文件夹

hitsz@hitsz-master:/bin$  **xsync /opt/module**

特别注意：虽然xsync可以自动对比文件的差异，只同步机器间不同的文件，但是启动集群后，/opt/module/hadoop目录下会生成data和logs两个文件夹，这两个文件夹不同的机器会产生不同的文件，不可以使用xsync进行同步，后续如果修改了配置文件，同步配置文件或者配置文件所在文件夹即可。绝对不能直接同步整个/opt/module/或者整个hadoop安装目录。如下所示，假设修改了core-site.xml文件。

hitsz@hitsz-master:/opt/module/hadoop/etc/hadoop$  **xsync core-site.xml**

或：

hitsz@hitsz-master:/opt/module/Hadoop $ **xsync $HADOOP\_HOME/etc/hadoop**

1. 同步环境变量

hitsz@hitsz-master:/bin$  **sudo /bin/xsync /etc/profile.d/my\_env.sh**

注意，由于环境变量文件属于root用户，所以需要使用sudo, 而在使用sudo时，xsync需要带上绝对路径。

1. 让环境变量生效

【该操作需要在hitsz-slave1, hitsz-slave2上进行】

hitsz@hitsz-slave1:~ $  **source /etc/profile**

hitsz@hitsz-slave2:~ $  **source /etc/profile**

## 4.12 启动Hadoop集群

【以下操作在hitsz-master上进行】

1. **如果集群是第一次启动**，需要在hitsz-master节点格式化NameNode（注意：格式化NameNode，会产生新的集群id，导致NameNode和DataNode的集群id不一致，集群找不到已往数据。如果集群在运行过程中报错，需要重新格式化NameNode的话，一定要先停止namenode和datanode进程，并且要删除所有机器的data和logs目录，然后再进行格式化。）

hitsz@hitsz-master:/bin$  **cd $HADOOP\_HOME**

hitsz@hitsz-master:/opt/module/hadoop $  **hdfs namenode -format**

1. 启动hdfs

hitsz@hitsz-master:/opt/module/hadoop $ **sbin/start-dfs.sh**

1. 在配置了ResourceManager的节点（hitsz-slave1）上启动yarn

hitsz@hitsz-slave1:/bin$  **cd $HADOOP\_HOME**

hitsz@hitsz-slave1:/opt/module/hadoop $ **sbin/start-yarn.sh**

1. 启动历史记录服务器（hitsz-master）

hitsz@hitsz-master:/opt/module/hadoop $ **mapred --daemon start historyserver**

1. 验证是否正常启动，在每台机器使用jps命令查看

hitsz@hitsz-master:/opt/module/hadoop $ **jps**

每台机器的输出结果应该等同于4.9节集群规划中的内容。

查看HDFS WebUI界面，点击DataNode，截图此界面In Operation部分。

<http://master:9870>

图形用户界面

中度可信度描述已自动生成

## 4.13 配置Hadoop客户端

1. 将服务端安装目录下etc/hadoop子目录下的四个配置4个配置文件替换客户端安装目录下etc/hadoop子目录下的文件。4个文件分别为core-site.xml，hdfs-site.xml，yarn-site.xml，mapred-site.xml。ubuntu下**文件管理器->其他位置->计算机即可进入根目录。找到opt->module->hadoop->etc->Hadoop 即可复制。**
2. 配置Host文件

以windows客户端为例，macos系统请自行百度，在此不赘述。将4.5配置的3行host复制到C:\Windows\System32\drivers\etc\hosts文件内，需要管理员权限，**建议先保存到桌面，再从桌面复制到文件夹内进行覆盖**。在cmd内输入

C:\> **ipconfig /flushdns**

## 4.14 上传文本文件

以windows客户端为例，假设文本文件为str.txt，位于D盘下。

（1） 新建目录，如果目录为多级，需要加 -p 参数

D:\> **hadoop fs -mkdir -p /user/hitsz/input**

1. 上传文件

D:\> **hadoop fs -put D:\str.txt** **/user/hitsz/input**

同组的同学不要用同一个目录，可以取input1，input2，input3

如何确认是否成功：

打开HDFS webUI 浏览器输入master:9870

右上角 Utilities -> Browse the file system ->user -> hitsz ->input

如果不成功，在所有hdfs路径前面加上hdfs://master:8020，再次尝试

例如

D:\> **hadoop fs -put D:\str.txt hdfs://master:8020/user/hitsz/input**

## 4.15 执行WordCount程序

1. 将core-site.xml，hdfs-site.xml，yarn-site.xml，mapred-site.xml四个配置文件添加到resources目录下。
2. 在WordCountDriver文件中添加以下配置代码

示例代码已经写了，直接删除三行代码前面的//就可以。

//设置允许跨平台运行

conf.set("mapreduce.app-submission.cross-platform", "true");

//设置用户名

System.setProperty("HADOOP\_USER\_NAME", "hitsz");

Job job = Job.getInstance(conf);

//关联jar包，此处应该填写自己的jar包位置+jar包名称+.jar。

job.setJar("E:\\workplace\\idea\\test\\hadoopdemo\\out\\artifacts\\wc\_jar\\hadoopdemo.jar");

Jar包的位置：File->Project Structure

图形用户界面, 文本, 应用程序

描述已自动生成

1. 修改运行配置，Edit Configuration -> Program arguments，填写输入输出路径，两个路径之间用空格分割。Output路径必须不存在。

hdfs://master:8020/user/hitsz/input 两个路径之间用空格连接hdfs://master:8020/user/hitsz/output01

1. 重新build项目生成jar包。

菜单Build->rebuild project，或者点击锤子。左侧项目结构会生成橙色的out目录。

1. 点击运行
2. 下载查看运行结果

D:\> **hadoop fs -get /user/hitsz/output01/part-r-00000 D:\**

图片包含 文本

描述已自动生成

## 4.16 Hadoop集群关闭

注意：该部分了解即可

如果存在问题需要关闭集群，按照启动的相反顺序依次进行关闭。

1. 关闭历史记录服务器（hitsz-master）

hitsz@hitsz-master:/opt/module/hadoop $ **mapred --daemon stop historyserver**

1. 在配置了ResourceManager的节点（hitsz-slave1）上关闭yarn

hitsz@hitsz-slave1:/bin$  **cd $HADOOP\_HOME**

hitsz@hitsz-slave1:/opt/module/hadoop $ **sbin/stop-yarn.sh**

1. 关闭hdfs（hitsz-master）

hitsz@hitsz-master:/opt/module/hadoop $ **sbin/stop-dfs.sh**

## 4.17 Hadoop集群WebUI界面

注意：该部分了解即可，WebUI可以在自己的电脑上访问

1. Web端查看HDFS的NameNode

浏览器中输入<http://master:9870>

图形用户界面

中度可信度描述已自动生成

点击Utilities下的Browse the file system 可以直接访问集群的HDFS系统。

图形用户界面, 应用程序

描述已自动生成

图形用户界面, 表格

描述已自动生成

1. Web端查看YARN的ResourceManager

浏览器中输入<http://slave1:8088>

图形用户界面, 应用程序, Word

描述已自动生成

1. Web端查看Mapreduce运行历史记录

浏览器中输入[http://master:19888:jobhistory](http://master:19888:jobhistory8)

图形用户界面, 应用程序, Word

描述已自动生成

# 常见错误和解决方案

1. 集群部分正常，部分无法启动

此错误主要有两个原因，配置文件出错以及Hosts文件出错。请仔细检查配置文件和hosts文件。

可以查看对应的日志来查找问题的所在。比如secondlynamenode没有启动， 在hitsz-slave2

上查看日志

hitsz@hitsz-master:~ $ **cd $HADOOP\_HOME/logs**

hitsz@hitsz-master:~ $ **vim hadoop-ices-secondarynamenode-ices-slave2.log**

1. hitsz-master可以正常启动，从属节点无法启动

考虑SSH配置和Hosts文件配置的问题。

1. 端口号被占用导致的无法启动

正常情况下本教程所使用的端口在机房环境下均不会出现端口占用的问题。在这种情况下，尝试关闭集群， 重启机器，再启动集群，必要时重新格式化namenode。如果问题依旧存在，基本为以下两个原因。

* + - 1. 配置文件出错，端口重复使用了。
      2. Hosts文件出错，将两个域名映射到了同一个IP地址。

1. Java.net.UnknownHostException: master:master

重新配置Hosts文件。同时主机名称和域名不要使用hadoop hadoop000等特殊名称。

1. 命令不存在的问题

检查环境变量的设置，以及修改环境变量后是否使用source命令使之生效。

1. vim 环境下输入命令无效

检查输入法，vim相关命令均需要在英文环境下输入。

1. vim 误删

按下Esc, 按u进行回退。

或者按下Esc，输入 :q! 退出，该操作会完全撤销这次的vim编辑。

1. jps不生效

全局变量hadoop java没有生效。解决办法：检查java环境变量是否设置正确，同时source /etc/profile文件。

1. jps发现进程已经没有，但是重新启动集群，提示进程已经开启。

原因是在Linux的根目录下/tmp目录中存在启动的进程临时文件，将集群相关进程删除掉，再重新启动集群。

1. DataNode和NameNode进程同时只能工作一个

图片包含 图示

描述已自动生成

如果无法判断问题的原因，一直无法解决问题，可以考虑删除 /opt/module目录， 删除环境变量my\_env.sh， 删除hosts文件，删除所有.ssh文件夹， 重启机器再进行重新的配置。