# Hadoop集群的搭建及配置

为了更好地学习Hadoop，需要学习者搭建Hadoop集群环境。Hadoop集群环境可以分为单机版环境，伪分布式环境和完全分布式环境。单机版环境是指在一台单机上运行Hadoop，没有分布式文件系统，而是直接读取本地操作系统的文件系统。伪分布式环境可以看做一个节点的集群，而完全分布式环境是多个节点的集群。本章较为细致地演示Hadoop完全分布式环境的搭建过程。为了能更好地学习Hadoop并基于Hadoop做大数据开发，建议个人笔记本硬件最低配置为内存至少8G，硬盘容量至少50G，CPU处理器为Intel i3以上的处理器。在搭建Hadoop完全分布式集群之前，需要安装必要的软件，主要的软件和版本如表 2‑1所示。本书使用的集群拓扑如图 2‑1所示。

表 2‑1 Hadoop相关软件及版本

|  |  |  |  |
| --- | --- | --- | --- |
| 软件 | 版本 | 安装包 | 备注 |
| Linux OS | CentOS6.8 | CentOS-6.8-x86\_64-bin-DVD1.iso | 64位 |
| JDK | 1.7+ | jdk-7u80-linux-x64.rpm |  |
| VMware54 | 10 | VMware-workstation-full-10.0.15255226.zip |  |
| Hadoop | 2.6.4 | hadoop-2.6.4.tar.gz |  |
| Eclipse | 4.5.1 | eclipse-jee-mars-1-win32-x86\_64.zip |  |
| Eclipse Hadoop插件 | 2.6.0 | hadoop-eclipse-plugin-2.6.0 |  |
| SSH连接工具 | 5 | Xme5.exe |  |



图 2‑1集群拓扑

## 安装配置虚拟机

### 创建虚拟机

VMware Workstation是一款功能强大的虚拟机软件，在不影响本机操作系统的情况下，用户可以在虚拟机中同时运行不同版本的操作系统。安装Wmware Workstation的过程比较简单，双击下载的VMware安装软件VMware-workstation-full-10.0.1，选择安装的目录，然后下一步，继续安装，然后输入产品序列号，安装完成后即可安装CentOS6.8操作系统。

接下来在安装好的VMware上安装CentOS6.8系统。安装步骤如下所示。

1. 打开安装好的VMware虚拟机，选择“创建新的虚拟机”，如图 2‑2所示。



图 2‑2新建虚拟机

1. 单击“创建新的虚拟机”之后进入到新建虚拟机向导，选择“典型（推荐）（T）”模式，再单击“下一步”，如图 2‑3所示。



图 2‑3新建虚拟机向导

1. 安装客户机操作系统，选择“稍后安装操作系统”，然后单击“下一步”，如图 2‑4所示。



图 2‑4安装客户机操作系统

1. 选择客户机操作系统，这里选择的是“Linux(L)”，版本是CentOS 64位，选择好之后直接单击“下一步”，如图 2‑5所示。



图 2‑5选择客户机操作系统

1. 命名虚拟机，首先在E盘创建一个以VMware命名的文件夹，并在该文件夹下建立一个以master命名的文件夹，虚拟机的名称为“master”，位置为E:\VMware\master，如图 2‑6所示。

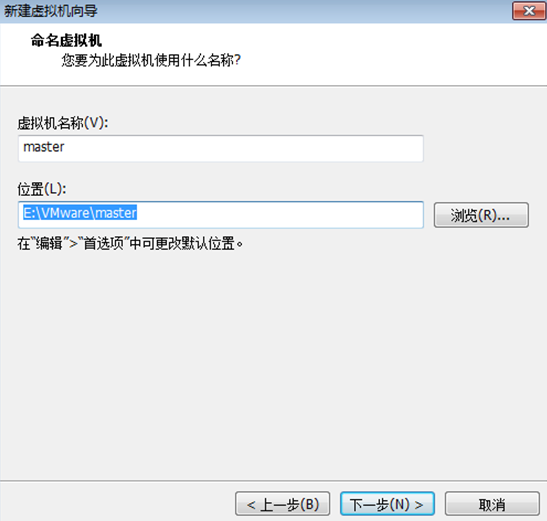


图 2‑6命名虚拟机

1. 指定磁盘量，指定最大磁盘大小为“20G”，选择将“将虚拟磁盘拆分成多个文件（M）”，单击“下一步”，如图 2‑7所示。

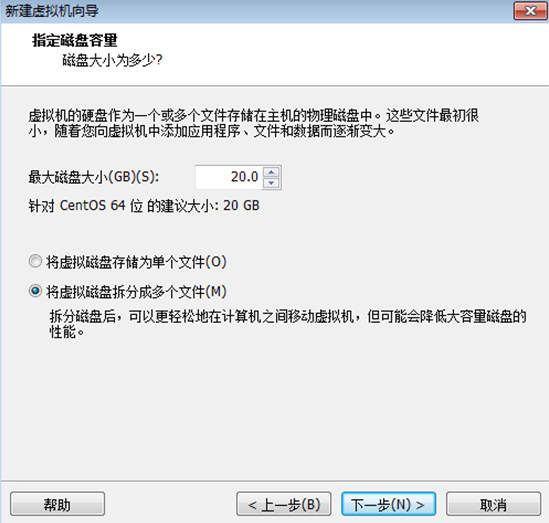


图 2‑7指定磁盘容量

1. 准备创建虚拟机，单击“自定义硬件”，进入硬件设置界面，单击“新CD/DVD(IDE)”，在右边的连接选项下选择ISO映像文件，并引进CentOS-6.8-x86\_64-bin-DVD1.iso的位置，如图 2‑8所示，最后单击“完成”。

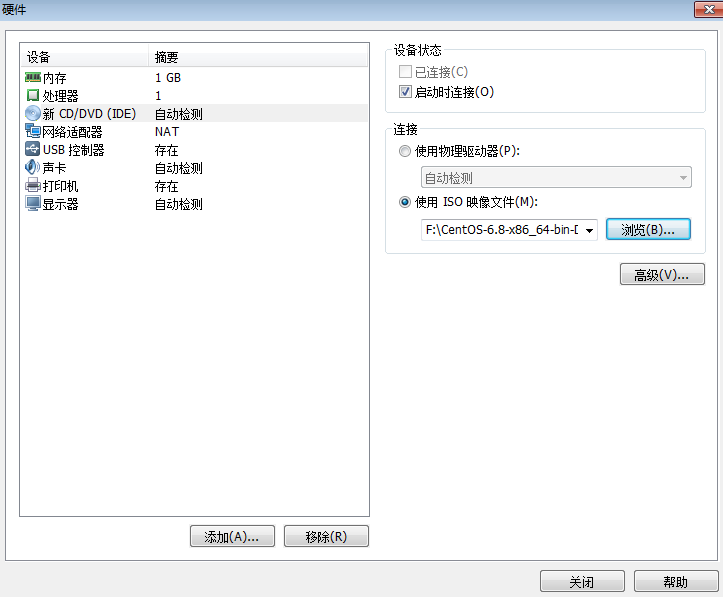


图 2‑8自定义硬件

1. 打开虚拟机，选择虚拟机“master”，单击“开启此虚拟机”，如图 2‑9所示。

图 2‑9开启虚拟机

1. 打开虚拟机之后看到CentOS 6.8的安装界面，按下Tab键可以选择接下来的操作，也可以不选择，56s之后系统会自动进入下一步，这里我们选择第一项，如图 2‑10所示，图中每个选项的解释如表 2‑3所示。

表 2‑3安装CentOS6.8系统的选项说明

|  |  |
| --- | --- |
| 选项 | 说明 |
| Install or upgrade an existing system | 安装或升级现有的系统 |
| Install system with basic video driver | 安装过程中采用基本的显卡驱动 |
| Rescue installed system | 进入系统修复模式 |
| Boot from local driver | 退出安装从磁盘启动 |
| Memory test | 内存检测 |

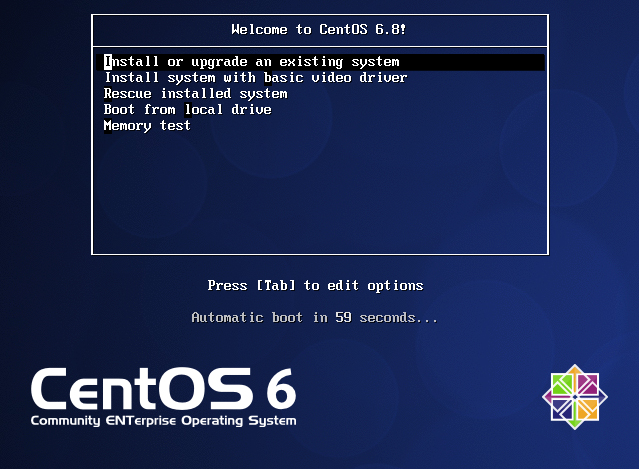


图 2‑10 CentOS6.8安装界面

1. Disc Found，选择“OK”测试你的安装媒体，即DVD光盘，如图 2‑11所示，我们选择“Skip”跳过，进去下一个界面之后直接单击“Next”。



图 2‑11 Disc Found

1. 选择安装的语言，选择“English(English)”，如图 2‑12所示。

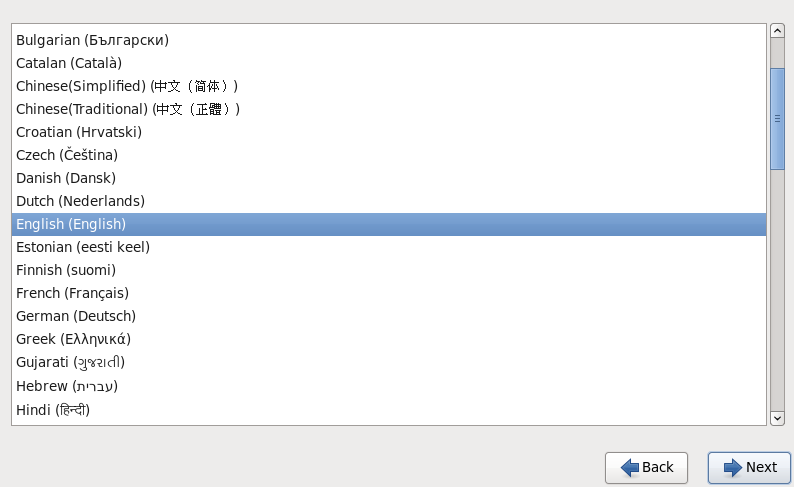


图 2‑12选择安装语言

1. 选择键盘，选择默认的“U.S.English”，如图 2‑13所示。

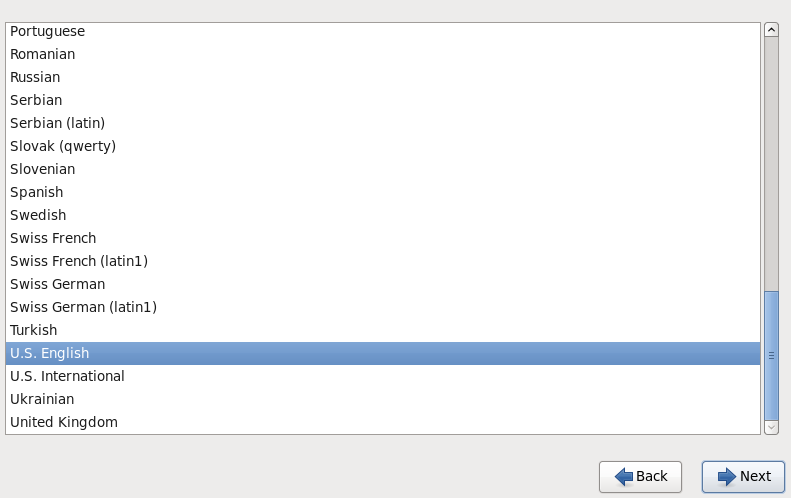


图 2‑13选择键盘

1. 选择安装的存储设备，一般都是安装到本地硬盘，所以选择“Basic Storage Devices”，然后单击 “Next”，如图 2‑14所示。

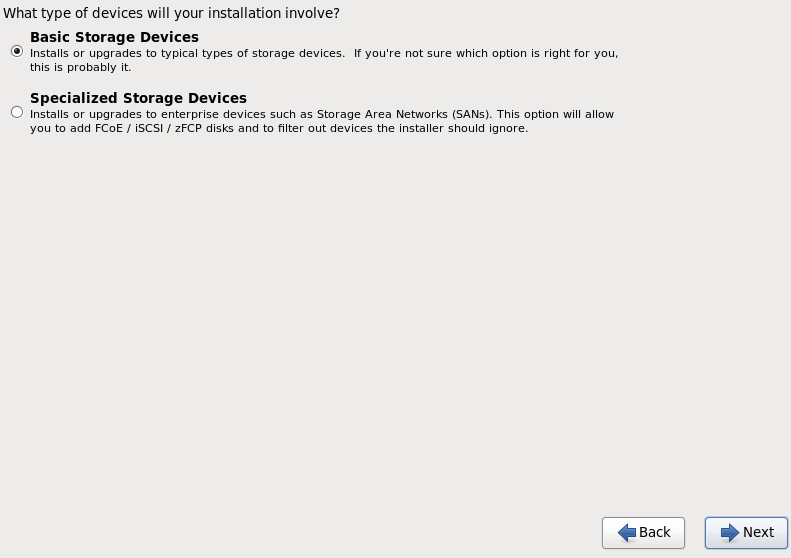


图 2‑14 选择安装的存储设备

1. 这时会弹出警告，提醒你是否删除检测到的这个硬盘的所有数据，直接单击“Yes,discard any data”，如图 2‑15所示。



图 2‑15提醒是否删除数据

1. 设置主机名为“master.centos.com”，单击“Next”，如图 2‑16所示。



图 2‑16设置主机名

1. 选择时区，选择“Asia/Shanghai”，单击“Next”，如图 2‑17所示。

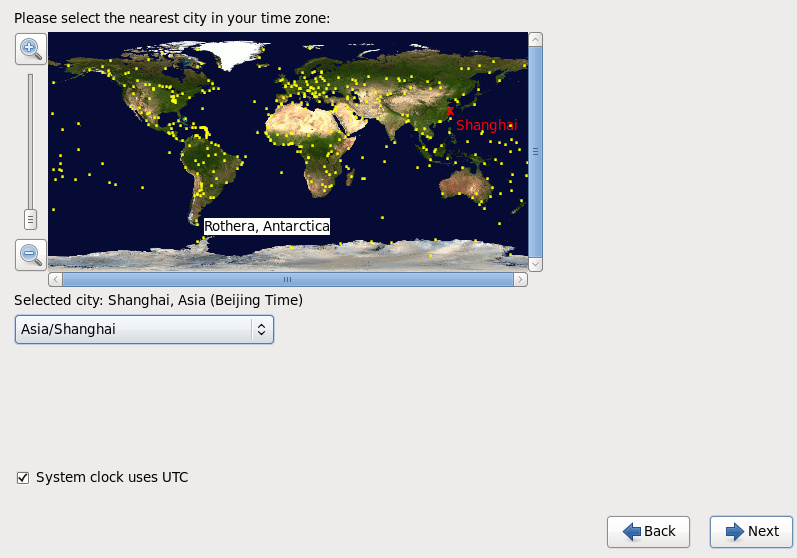


图 2‑17选择时区

1. 设置根目录(root)的密码，本书集群所有节点的根目录密码都设置为“123456”，如图 2‑18所示，如果密码设置得过于简单，则会弹出一个提示，可以选择“Use Anyway”，如图 2‑19所示。

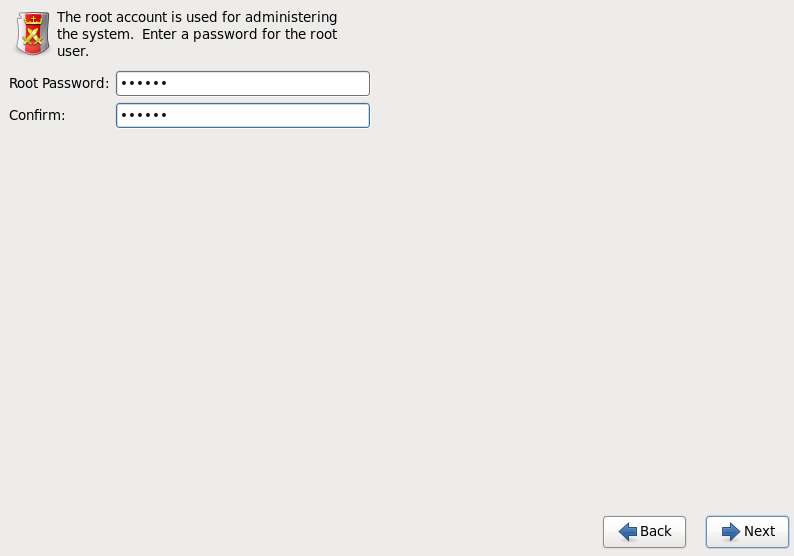


图 2‑18设置root密码

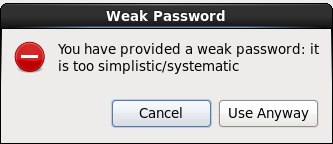


图 2‑19弱密码提示

1. 接下来是分区与安装的界面，选择“Use All Space”，如图 2‑20所示，单击下一步，会提示确认格式化并写入数据，选择“Write Changes to disks”，如图 2‑21所示。

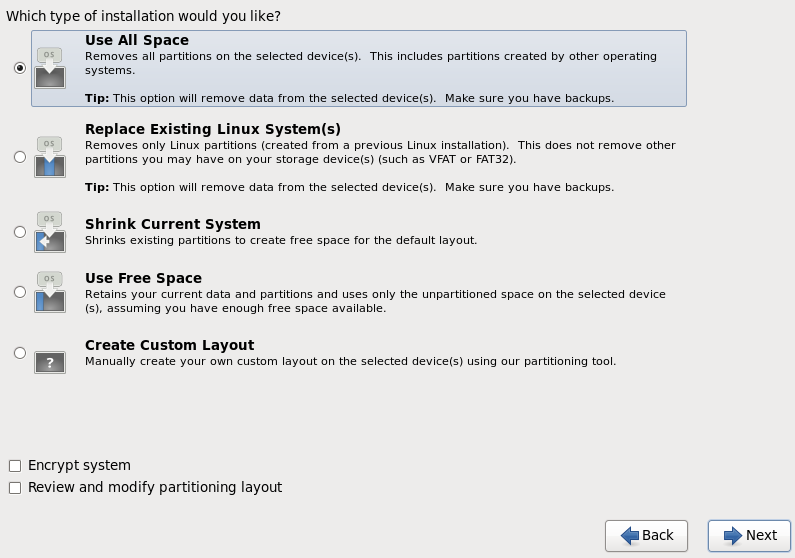


图 2‑20分区与安装

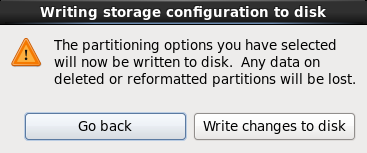


图 2‑21提示确认格式化与写入数据

1. 选择安装Centos组件，选择“Minimal”，如图 2‑22所示，单击“Next”开始安装，如图 2‑23所示。

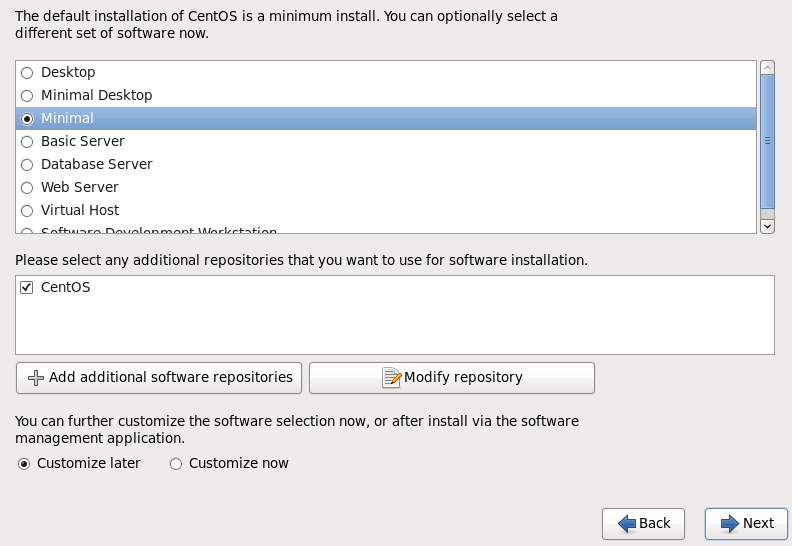


图 2‑22选择安装Centos组件



图 2‑23开始安装

1. 安装完成，单击“Reboot”进入Cent0S系统，如图 2‑24所示。

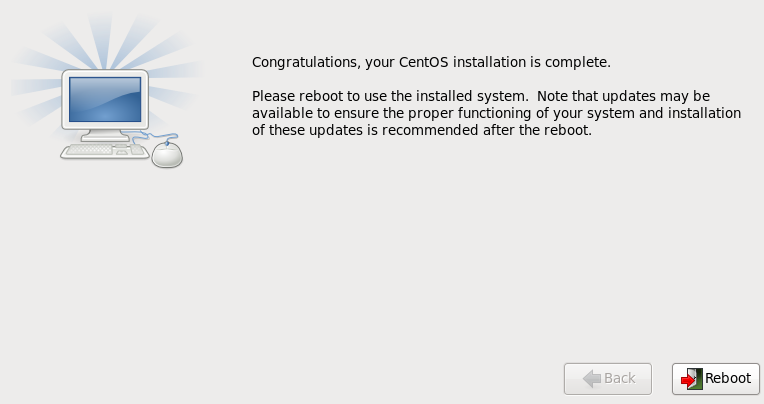


图 2‑24安装完成

1. 登陆，输入用户名“root”以及密码“123456”，如图 2‑25所示。

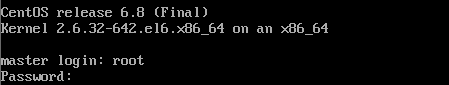


图 2‑25登录

### 设置固定IP

本书使用的集群有4台虚拟机，每台虚拟机均使用NAT模式接入网络，为其分配IP,并保证每台虚拟机的IP处于同一子网内。所以需要配置固定虚拟机的IP，这里以虚拟机master为例，详细介绍虚拟机固定IP的步骤，主要步骤如下。

1. 重启服务，输入“service network restart”，如表 2‑4所示。

表 2‑4重启服务

|  |
| --- |
| [root@master ~]# service network restart  Shutting down loopback interface: [ OK ]  Bringing up loopback interface: [ OK ] |

1. 修改配置文件/etc/sysconfig/network-scripts/ifcfg-eth0。

执行命令vi /etc/sysconfig/network-scripts/ifcfg-eth0，修改之前文件内容如代码 2‑1所示。

代码 2‑1 ifcfg-eth0文件修改前的内容

|  |
| --- |
| DEVICE=eth0  HWADDR=00:0C:29:5E:6C:55  TYPE=Ethernet  UUID=e5423901-b152-4f05-a6d4-68769c681046  ONBOOT=no  NM\_CONTROLLED=yes  BOOTPROTO=dhcp |

其中，DEVICE是指设备名，HWADDR是指网卡地址，ONTBOOT是设置系统启动时是否激活网卡，BOOTPROTO的值可以设置为dhcp，none，bootp或static，每个值的解释如表 2‑5所示。

表 2‑5 设置BOOTPROTO的值

|  |  |
| --- | --- |
| dhcp | 设置网卡绑定的时候通过DHCP协议的方法来获得地址 |
| none | 设置网卡绑定的时候不使用任何协议 |
| bootp | 设置网卡绑定的时候使用BOOTP协议 |
| static | 设置网卡绑定的时候使用静态协议 |

将该文件的ONBOOT=no改为ONBOOT=yes，BOOTPROTO=dhcp改为BOOTPROTO=static，并添加IP地址IPADDR，子网掩码NETMASK，网关GATEWAY以及DNS1等信息，修改后的文件内容如代码 2‑2所示。

代码 2‑2修改ifcfg-eth0文件

|  |
| --- |
| DEVICE=eth0  HWADDR=00:0C:29:5E:6C:55  TYPE=Ethernet  UUID=e5423901-b152-4f05-a6d4-68769c681046  ONBOOT=yes  NM\_CONTROLLED=yes  BOOTPROTO=static  #添加以下内容  IPADDR=192.168.128.130  NETMASK=255.255.255.0  GATEWAY=192.168.128.2  DNS1=192.168.128.2 |

1. 再次重启服务并查看IP，如表 2‑6所示，从图中可以看出IP地址已经是我们设置的192.168.128.130了，说明配置固定IP成功。

表 2‑6重启服务并查看ip

|  |
| --- |
| [root@master ~]# service network restart  Shutting down loopback interface: [ OK ]  Bringing up loopback interface: [ OK ]  Bringing up interface eth0: Determining if ip address 192.168.128.130 is already in use for device eth0... [ OK ]  [root@master ~]# ifconfig  eth0 Link encap:Ethernet HWaddr 00:0C:29:87:29:92  inet addr:192.168.128.130 Bcast:192.168.128.255 Mask:255.255.255.0  inet6 addr: fe80::20c:29ff:fe87:2992/64 Scope:Link  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  RX packets:209751 errors:0 dropped:0 overruns:0 frame:0  TX packets:110480 errors:0 dropped:0 overruns:0 carrier:0  collisions:0 txqueuelen:1000  RX bytes:42455267 (40.4 MiB) TX bytes:16300905 (15.5 MiB)  lo Link encap:Local Loopback  inet addr:127.0.0.1 Mask:255.0.0.0  inet6 addr: ::1/128 Scope:Host  UP LOOPBACK RUNNING MTU:65536 Metric:1  RX packets:10873 errors:0 dropped:0 overruns:0 frame:0  TX packets:10873 errors:0 dropped:0 overruns:0 carrier:0  collisions:0 txqueuelen:0  RX bytes:1261490 (1.2 MiB) TX bytes:1261490 (1.2 MiB) |

### 远程连接虚拟机

1. 双击程序图标Xshell，打开Xshell，如图 2‑26所示。



图 2‑26双击Xshell

1. 单击文件——新建，建立会话，如图 2‑27所示。



图 2‑27新建会话

1. 配置新建会话属性，在常规的名称一栏填入“master”，当然，这个名称可以随便取，在主机一栏中填入master的IP，即192.168.128.130，如图 2‑28单击左边的“用户身份验证”，在右边填写用户名和密码，master的用户名是root，密码是123456，如图 2‑29所示，单击“确定”，创建会话完成。

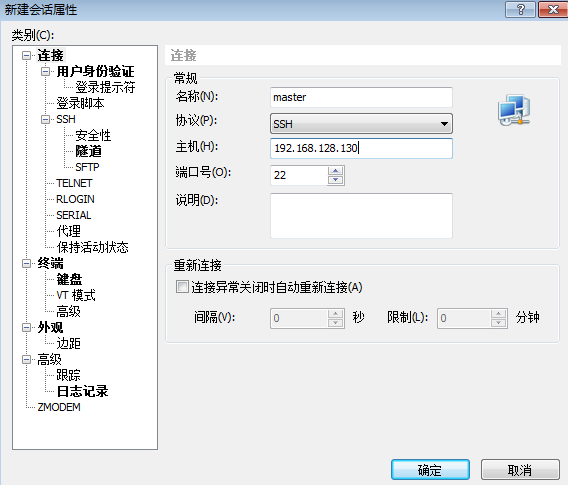


图 2‑28创建会话属性1

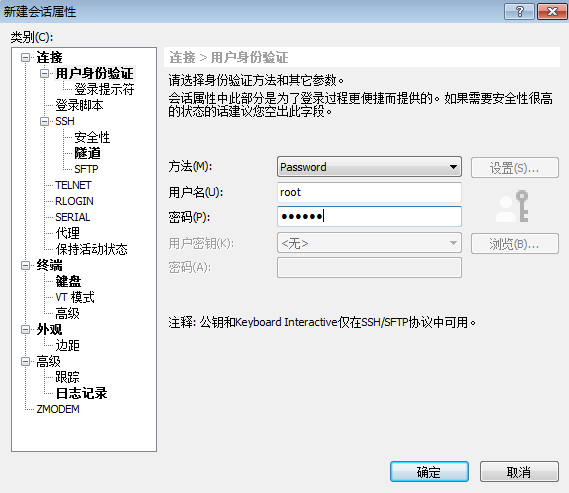


图 2‑29创建会话属性2

1. 连接会话，打开会话窗口，选中会话“master”，单击“连接”，如图 2‑30这时会弹出SSH安全警告，单击“接受并保存”即可连接上master。如图 2‑31所示。

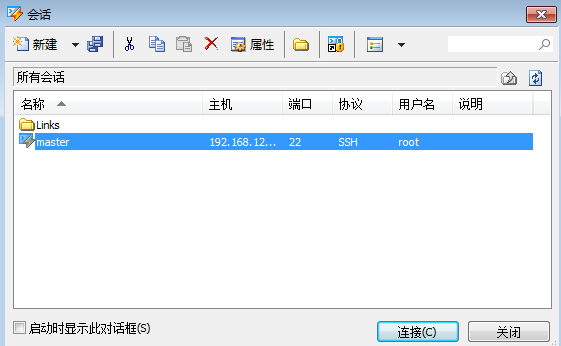


图 2‑30连接会话

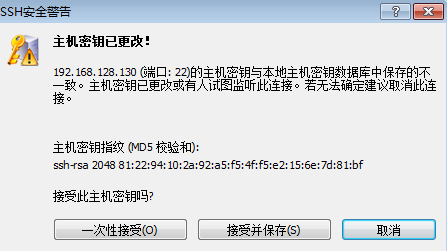


图 2‑31 SSH安全警告

### 虚拟机离线安装软件

1. 执行命令cd /etc/yum.repos.d进入/etc/yum.repos.d目录。
2. 查看yum.repos.d目录下的文件，会看到CentOS-Base.repo,CentOS-Debuginfo.repo，CentOS-fasttrack.repo,CentOS-Vault.repo,Centos-Media.repo 5个文件，其中Centos-Media.repo是yum本地源的配置文件。配置本地yum源，需要将除yum本地源以外的其他yum源禁用，可将CentOS-Base.repo,CentOS-Debuginfo.repo,CentOS-fasttrack.repo,CentOS-Vault.repo改为CentOS-Base.repo.bak,CentOS-Debuginfo.repo.bak,CentOS-fasttrack.repo.bak,CentOS-Vault.repo.bak，也可以删除CentOS-Base.repo,CentOS-Debuginfo.repo CentOS-fasttrack.repo,CentOS-Vault.repo这4个文件。

执行重命名命令示例：mv CentOS-Base.repo CentOS-Base.repo.bak

执行删除命令：rm –rf CentOS-Base.repo CentOS-Debuginfo.repo CentOS-fasttrack.repo CentOS-Vault.repo

1. 执行命令vi Centos-Media.repo打开文件，修改其内容，修改前的内容如代码 2‑3所示。

代码 2‑3 Centos-Media.repo修改前的内容

|  |
| --- |
| [c6-media]  name=CentOS-$releasever - Media  baseurl=file:///media/CentOS/  file:///media/cdrom/  file:///media/cdrecorder/  gpgcheck=1  enabled=0  gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6 |

将baseurl的值修改为file:///meida/，gpgcheck的值改为0，enabled的值改为1，修改后的内容如代码 2‑4所示。

代码 2‑4 Centos-Media.repo修改后的内容

|  |
| --- |
| [c6-media]  name=CentOS-$releasever - Media  baseurl=file:///media/  gpgcheck=0  enabled=1  gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6 |

1. 挂载

执行命令：mount /dev/dvd /media，如图 2‑32所示，出现了“mount:you must specify the filesystem type”的信息提示，说明没有挂载成功，解决方案为右键单击虚拟机master——选择设置——CD/DVD——设备状态——在已连接前面打钩，如图 2‑33所示。



图 2‑32挂载失败

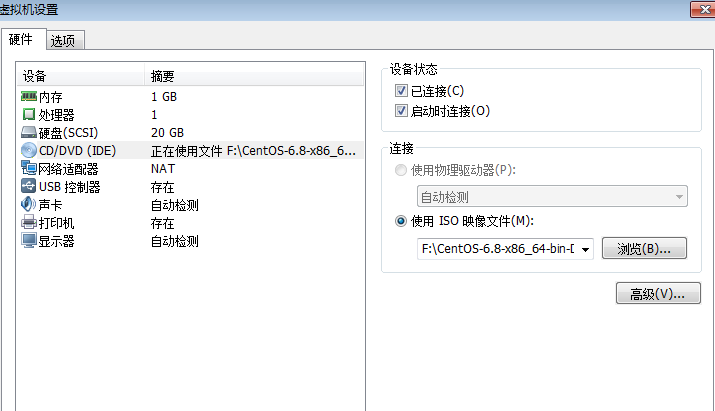


图 2‑33挂载失败解决方案

再次执行挂载命令，出现“mount:block device /dev/sr0 is write-protected,mounting read-only”信息提示，说明挂载成功，如图 2‑34所示。



图 2‑34挂载成功

1. 更新yum源。

执行命令：yum clean all，出现如图 2‑35所示的信息，说明更新yum源成功。

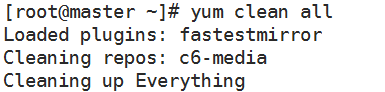


图 2‑35更新yum源

1. 使用yum安装软件，这里我们安装vim，zip，openssh-server，openssh-clients，每个软件的说明如表 2‑9所示。

表 2‑9软件说明

|  |  |
| --- | --- |
| 软件 | 说明 |
| vim | 类似于vi的文本编辑器 |
| zip | 压缩文件命令 |
| openssh-server | 主要是作为一个服务运行在后台，如果这个服务开启，我们就可以用一些远程连接工具来连接centOS |
| openssh-client | 类似于XShell，可以作为一个客户端连接上openssh-server |

安装软件命令：yum install –y vim zip openssh-server openssh-clients，安装过程中自动搜索并安装需要安装的软件所必须的依赖包，如图 2‑36所示。安装完成会提示你已安装相关的软件，如图 2‑37所示。

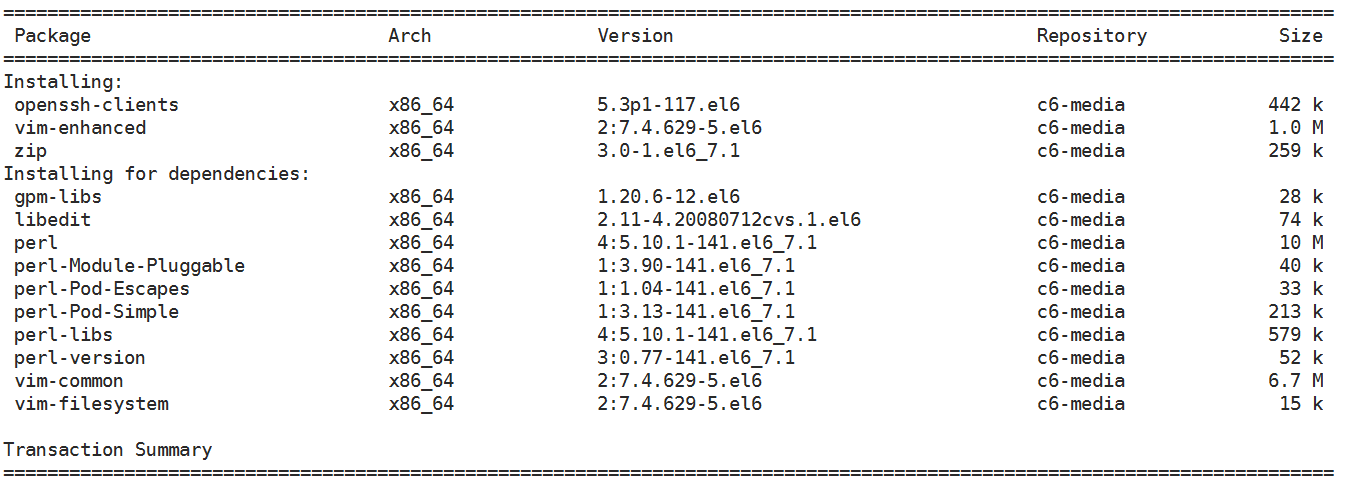


图 2‑36安装依赖包

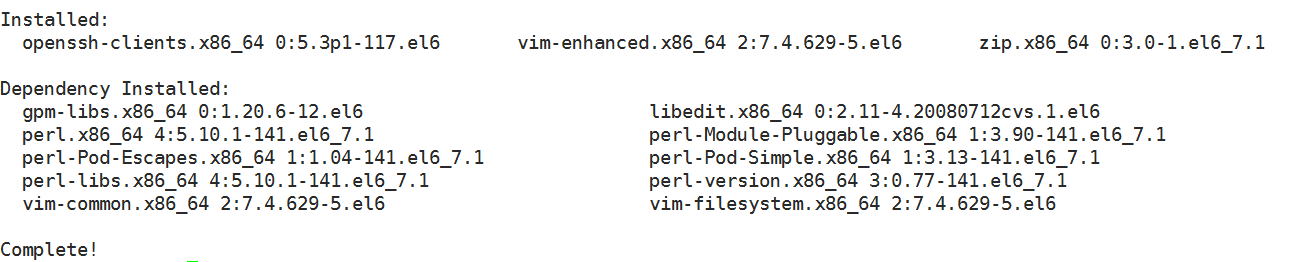


图 2‑37 yum安装软件完成

## 安装Java

### 在Windows下安装Java

JDK是 Java语言的软件开发工具包，主要用于移动设备、嵌入式设备上的Java应用程序。JDK是整个Java开发的核心，它包含了Java的运行环境，Java工具和Java基础的类库。本书后续章节的Hadoop开发是基于Java语言开发的，所以需要在Windows下安装JDK，具体安装步骤如下所示。

1. 双击jdk-7u80-windows-x64.exe，如图 2‑38所示，单击“下一步”。

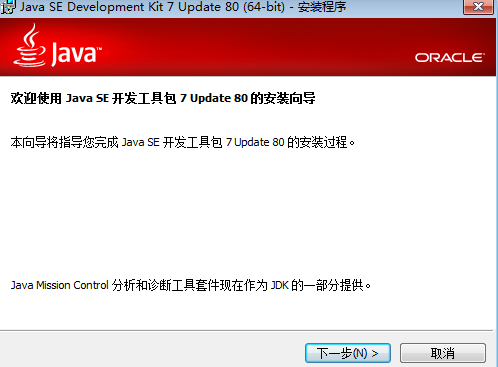


图 2‑38 安装jdk

1. 更改安装目录，单击“更改”,自己选定一个目录，如图 2‑39所示，然后等待JDK安装完成即可。



图 2‑39选择JDK的安装目录

1. 安装完JDK时，会弹出安装jre7的提示窗口，我们这里根据自己的需要更改jre的安装目录，需要注意的是jdk和jre的安装目录最好在同一个文件夹下，比如都在C:\Program Files\java\下，如图 2‑40所示，单击“下一步”，进入到jre的安装。jre安装完成之后单击关闭按钮即可完成JDK的安装。



图 2‑40选择jre的安装目录

1. 配置环境变量。
2. 右键单击计算机——属性——高级系统设置——环境变量，出现环境变量设置窗口，如图 2‑41所示。

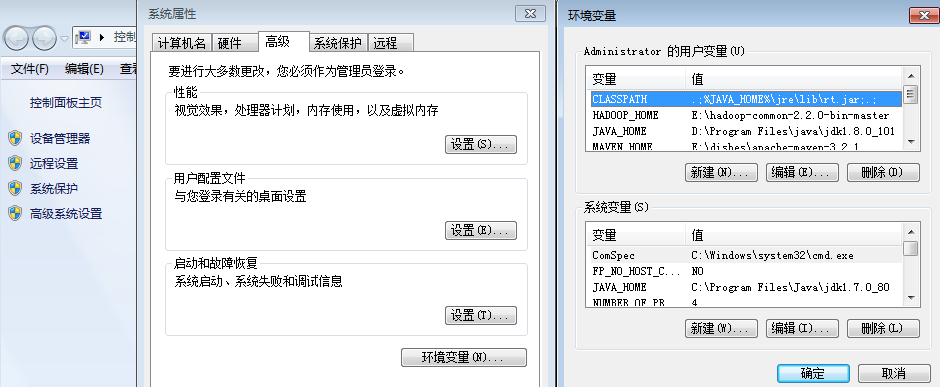


图 2‑41环境变量设置窗口

1. 新建 JAVA\_HOME变量，变量值填写JDK安装路径，如图 2‑42所示。

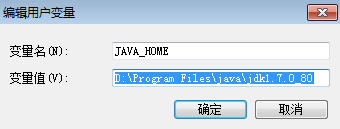


图 2‑42新建JAVA\_HOME变量

1. 新建 CLASSPATH 变量，变量值输入 .;%JAVA\_HOME%\jre\lib\rt.jar;.;如图 2‑43所示。

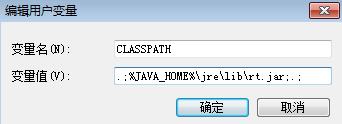


图 2‑43新建变量CLASSPATH

1. 找到 Path 变量，在变量值最后输入;%JAVA\_HOME%\bin;（注意在最前面有个;号），如图 2‑44所示。

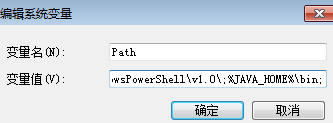


图 2‑44修改Path变量

1. 测试环境变量是否配置成功。双击电脑开始——所有程序——附件——命令提示符，输入 java —version，出现如表 2‑10所示的信息，说明安装配置成功。

表 2‑10验证JDK安装成功

|  |
| --- |
| C:\Users\Administrator>java –version  java version “1.7.0\_80”  Java<TM> SE Runtime Environment <build .7.0\_80-b15>  Java HotSport<TM> 64-Bit Server VM <build 24.80-b11,mixed mode> |

### 在Linux下安装Java

在Linux下安装JDK的步骤如下所示。

1. 上传JDK安装包，使用CTRL+ALT+F弹出文件传输框，把jdk-7u80-linux-x64.rpm上传到/opt上，如图 2‑45所示。

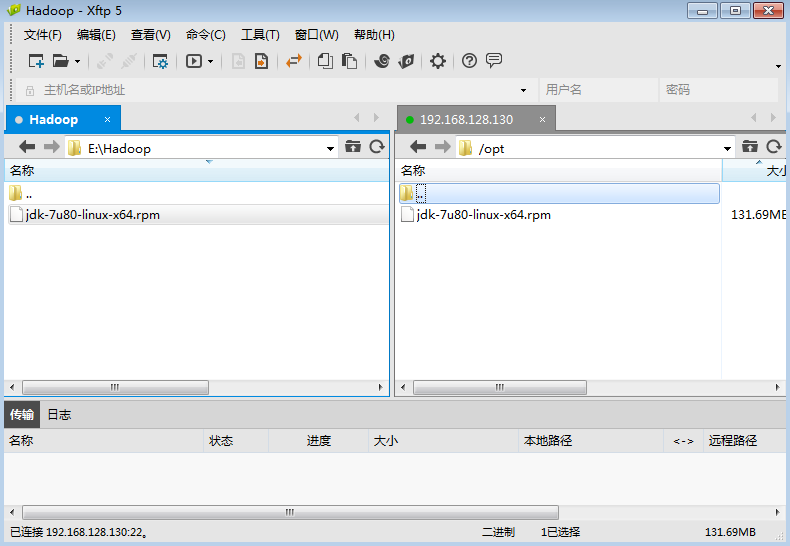


图 2‑45上传JDK安装包

1. 进入/opt目录，执行命令“rpm –ivh jdk-7u80-linux-x64.rpm”安装JDK，如图 2‑46所示。

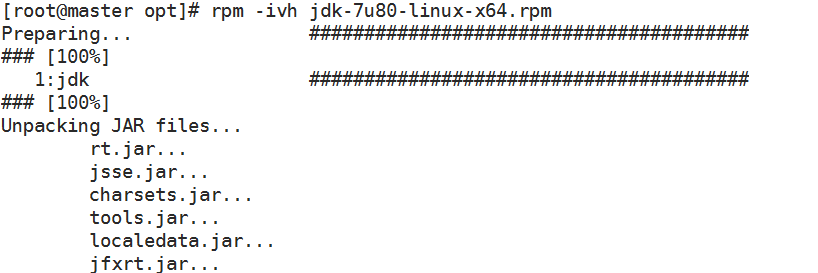


图 2‑46安装JDK

1. 验证JDK是否配置成功，执行命令“java -version”，如图 2‑47所示说明JDK配置成功。

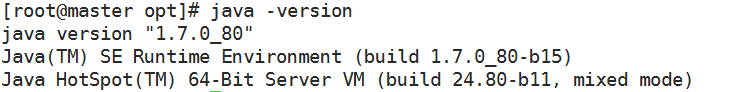


图 2‑47验证JDK配置成功

## 搭建Hadoop完全分布式集群

### 修改配置文件

在任务2.1已经创建建配置了虚拟机master，接下来在虚拟机master上做Hadoop的相关配置。首先需要把Hadoop安装包hadoop-2.6.4.tart.gz上传的虚拟机master的/opt目录下，然后执行命令tar –zxf hadoop-2.6.4.tar.gz –C /usr/local将Hadoop安装包解压到虚拟机master的/usr/local目录下。

Hadoop配置涉及到的文件有7个：core-site.xml、hadoop-env.sh、yarn-env.sh、mapred-site.xml、yarn-site.xml、slaves、hdfs-site.xml。这些文件都在目录/usr/local/hadoop-2.6.4/etc/hadoop/目录下，首先要进入到该目录下，文件具体修改内容如下所示。

1. 修改core-site.xml文件，这个是Hadoop的核心配置文件，这里需要配置两个属性，fs.defaultFS配置了Hadoop的HDFS系统的命名，位置为主机的8020端口，这里需要注意替换hdfs://master:8020中斜体的master，改名字为NameNode所在机器的机器名；hadoop.tmp.dir配置了Hadoop的临时文件的位置。添加内容如代码 2‑5所示。

代码 2‑5修改core-site.xml

|  |
| --- |
| <configuration>  <property>  <name>fs.defaultFS</name>  <value>hdfs://master:8020</value>  </property>  <property>  <name>hadoop.tmp.dir</name>  <value>/var/log/hadoop/tmp</value>  </property>  </configuration> |

1. 修改hadoop-env.sh文件，该文件是Hadoop运行基本环境的配置，需要修改为JDK的实际位置。故在该文件中修改JAVA\_HOME值为本机安装位置，如代码 2‑6所示。

代码2‑6修改hadoop-env.sh

|  |
| --- |
| export JAVA\_HOME=/usr/jdk1.8.0\_144 |

1. 修改yarn-env.sh，该文件是YARN框架运行环境的配置，同样需要修改JDK所在位置。在该文件中修改JAVA\_HOME值为本机安装位置，如代码 2‑7所示。

代码 2‑7修改yarn-env.sh文件

|  |
| --- |
| export JAVA\_HOME=/usr/jdk1.8.0\_144 |

1. 修改mapred-site.xml，这个是MapReduce的相关配置，由于Hadoop2.x使用了YARN框架，所以必须在mapreduce.framework.name属性下配置yarn。mapreduce.jobhistory.address和mapreduce.jobhistoryserver.webapp.address是配置JobHistoryserver的相关配置，即运行MapReduce任务的日志相关服务，这里同样需要注意修改“master”为实际服务所在机器的机器名。

mapred-site.xml文件是从mapred-site.xml.template文件复制得到的，复制命令为cp mapred-site.xml.template mapred-site.xml，修改内容如代码2‑8所示。

代码2‑8修改mapred-site.xml

|  |
| --- |
| <configuration>  <property>  <name>mapreduce.framework.name</name>  <value>yarn</value>  </property>  <!-- jobhistory properties -->  <property>  <name>mapreduce.jobhistory.address</name>  <value>master:10020</value>  </property>  <property>  <name>mapreduce.jobhistory.webapp.address</name>  <value>master:19888</value>  </property>  </configuration> |

1. 修改yarn-site.xml文件，该文件为YARN框架的配置，在最开始命名了一个yarn.resourcemanager.hostname的变量，这样在后面YARN的相关配置中就可以直接引用该变量了，其他配置保持不变即可。修改内容如代码2‑9所示。

代码2‑9修改yarn-site.xml

|  |
| --- |
| <configuration>  <!-- Site specific YARN configuration properties -->  <property>  <name>yarn.resourcemanager.hostname</name>  <value>master</value>  </property>  <property>  <name>yarn.resourcemanager.address</name>  <value>${yarn.resourcemanager.hostname}:8032</value>  </property>  <property>  <name>yarn.resourcemanager.scheduler.address</name>  <value>${yarn.resourcemanager.hostname}:8030</value>  </property>  <property>  <name>yarn.resourcemanager.webapp.address</name>  <value>${yarn.resourcemanager.hostname}:8088</value>  </property>  <property>  <name>yarn.resourcemanager.webapp.https.address</name>  <value>${yarn.resourcemanager.hostname}:8090</value>  </property>  <property>  <name>yarn.resourcemanager.resource-tracker.address</name>  <value>${yarn.resourcemanager.hostname}:8031</value>  </property>  <property>  <name>yarn.resourcemanager.admin.address</name>  <value>${yarn.resourcemanager.hostname}:8033</value>  </property>  <property>  <name>yarn.nodemanager.local-dirs</name>  <value>/data/hadoop/yarn/local</value>  </property>  <property>  <name>yarn.log-aggregation-enable</name>  <value>true</value>  </property>  <property>  <name>yarn.nodemanager.remote-app-log-dir</name>  <value>/data/tmp/logs</value>  </property>  <property>  <name>yarn.log.server.url</name>  <value>http://master:19888/jobhistory/logs/</value>  <description>URL for job history server</description>  </property>  <property>  <name>yarn.nodemanager.vmem-check-enabled</name>  <value>false</value>  </property>  <property>  <name>yarn.nodemanager.aux-services</name>  <value>mapreduce\_shuffle</value>  </property>  <property>  <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>  <value>org.apache.hadoop.mapred.ShuffleHandler</value>  </property>  </configuration> |

1. 修改slaves文件，该文件里面保存有slave节点的信息，在slaves文件里面添加如代码 2‑10所示的内容。

代码 2‑10修改slaves文件

|  |
| --- |
| slave1  slave2  slave3 |

1. 修改hdfs-site.xml文件，这个是HDFS相关的配置文件，dfs.namenode.name.dir和dfs.datanode.data.dir分别指定了NameNode元数据和DataNode数据存储位置。dfs.namenode.secondary.http-address配置的是SecondaryNameNode的地址，同样需要注意修改“master”为实际SecondaryNameNode地址。dfs.replication配置了文件块的副本数，默认就是3个，所以这里也可以不配置。修改内容如代码 2‑11所示。

代码 2‑11修改hdfs-site.xml文件

|  |
| --- |
| <configuration>  <property>  <name>dfs.namenode.name.dir</name>  <value>file:///data/hadoop/hdfs/name</value>  </property>  <property>  <name>dfs.datanode.data.dir</name>  <value>file:///data/hadoop/hdfs/data</value>  </property>  <property>  <name>dfs.namenode.secondary.http-address</name>  <value>master:50090</value>  </property>  <property>  <name>dfs.replication</name>  <value>3</value>  </property>  </configuration> |

### 克隆虚拟机

在虚拟机master上配置好Hadoop之后，将虚拟机master克隆到slave1，slave2，slave3。在虚拟机master的安装目录E:\VMware下建立三个文件slave1，slave2，slave3。下面以克隆master到slave1为例，详细介绍虚拟机的克隆步骤。

1. 右键单击虚拟机master——管理——克隆，来到欢迎使用克隆虚拟机向导的界面，直接单击“下一步”。
2. 选择克隆源，这里选择“虚拟机中的当前状态”，如图 2‑48所示。

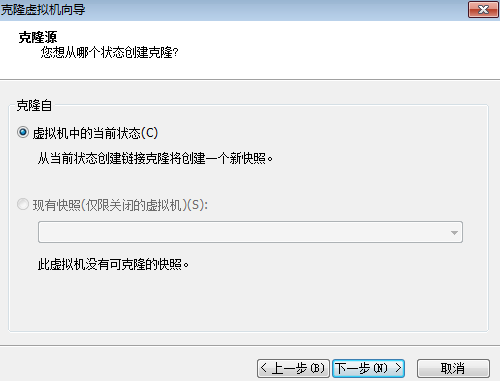


图 2‑48选择克隆源

1. 选择“创建完整克隆”并单击“下一步”，如图 2‑49所示。

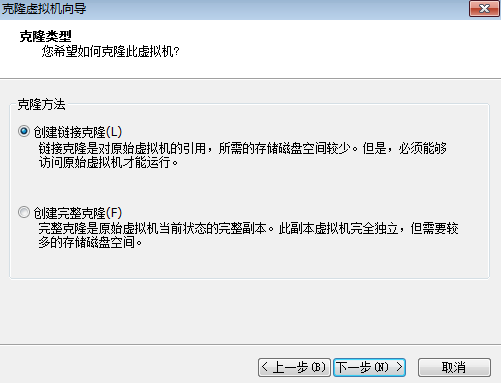


图 2‑49选择克隆类型

1. 设置新虚拟机的名称，新虚拟机名称为slave1，选择安装位置为E:\VMware\slave1，如图 2‑50所示，单击“完成”，虚拟机开始克隆，最后单击“关闭”，如图 2‑51克隆虚拟机完成。

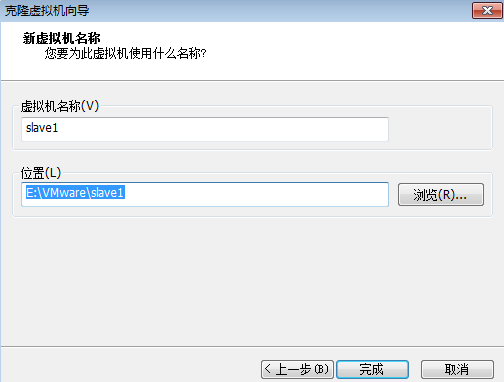


图 2‑50新建虚拟机名称

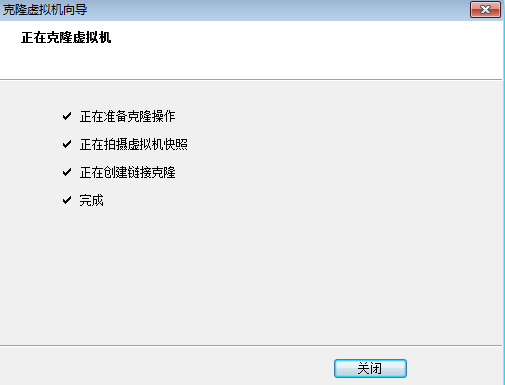


图 2‑51克隆虚拟机完成

1. 开启虚拟机slave1，修改相关配置。
2. 执行命令rm –rf /etc/udev/rules.d/70-persistent-net.rules删除/etc/udev/rules.d/70-persistent-net.rules文件。
3. 执行命令ifconfig -a查看HWADDR，如图 2‑52所示，记录下新虚拟机的HWADDR为00:0C:29:6A:DB:27（不同机器此值是不一样，即笔者实验的值和读者的是不一样的）。

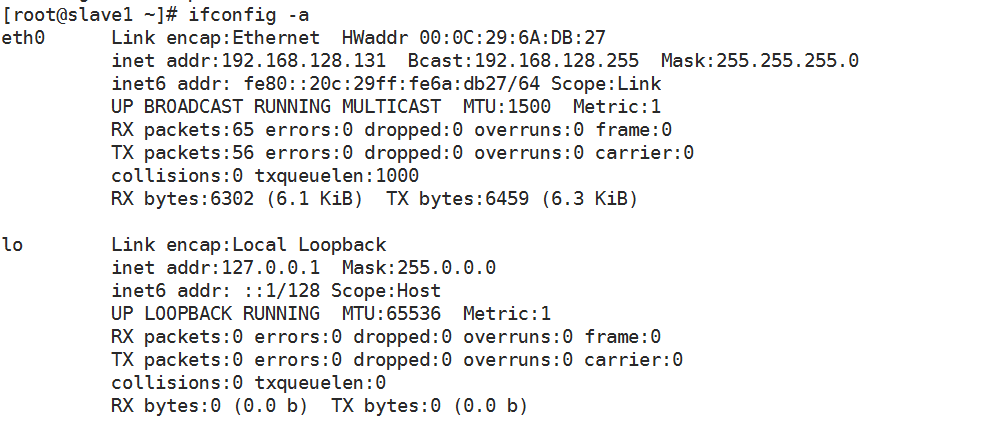


图 2‑52 查看新虚拟机的HWADDR

1. 修改/etc/sysconfig/network-scriptd/ifcfg-eth0文件，修改HWADDR，IPADDR以及注释掉或删除UUID开头的一行内容，并修改HWADDR的内容为实际的地址，修改内容如代码 2‑12所示。

代码2‑12修改slave1的ifcfg-eth0文件

|  |
| --- |
| DEVICE=eth0  HWADDR=00:0C:29:6A:DB:27  TYPE=Ethernet  #UUID=de344ae9-6449-4fdc-8155-f188b4fc7362  ONBOOT=yes  NM\_CONTROLLED=yes  BOOTPROTO=static  IPADDR=192.168.128.131  NETMASK=255.255.255.0  GATEWAY=192.168.128.2  DNS1=192.168.128.2 |

1. 修改机器名，执行vim /etc/sysconfig/network打开文件，修改机器名为slave1.centos.com，如代码2‑13所示。

代码 2‑13修改机器名

|  |
| --- |
| NETWORKING=yes  HOSTNAME=slave1.centos.com |

1. 重启，执行reboot重新启动虚拟机。
2. 验证slave1是否配置成功。在master下ping slave1，如图 2‑53所示，说明slave1配置成功。

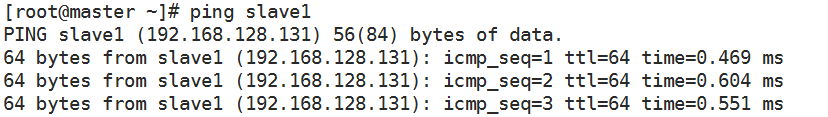


图 2‑53在master下ping slave1

1. 重复1-5的相关步骤，克隆master到slave2，slave3并修改slave2，slave3的相关配置。

### 配置SSH免密码登录

SSH(Secure Shell的缩写)是建立在TCP/TP协议的应用层和传输层基础上的安全协议。SSH保障了远程登录和网络传输服务的安全性，起到防止信息泄露等作用，通过SSH可以对文件进行加密处理，SSH也可以运行于多平台。配置SSH无密码登录的步骤如下所示，以下步骤都是在主节点master上操作。

1. 使用ssh-keygen产生公钥与私钥对。

输入命令：ssh-keygen -t rsa，接着按三次enter键，如图 2‑54所示，生成私钥密钥id\_rsa和公有密钥id\_rsa.pub两个文件。ssh-keygen是用来生成RSA类型的密钥以及管理该密钥，参数“-t”用于指定要创建的SSH密钥的类型为RSA。

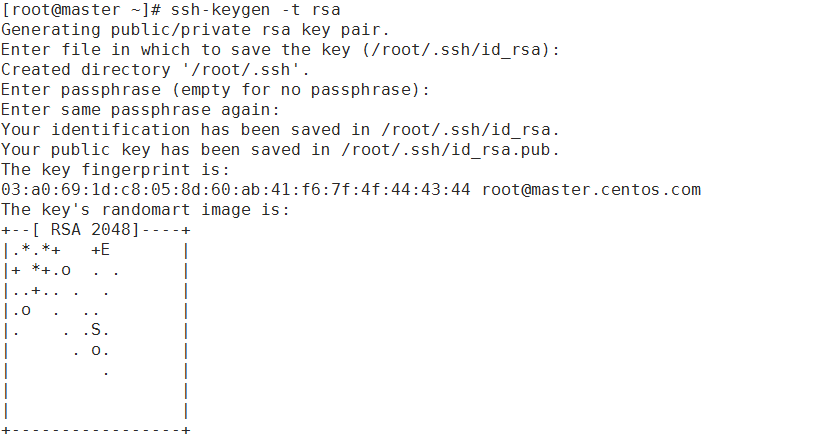


图 2‑54设置SSH免密码登录

1. 修改ip与hostsname映射，修改/etc/hosts文件，在文件中添加如代码2‑14所示的内容。

代码 2‑14修改/etc/hosts文件

|  |
| --- |
| 192.168.128.130 master master.centos.com  192.168.128.131 slave1 slave1.centos.com  192.168.128.132 slave2 slave2.centos.com  192.168.128.133 slave3 slave3.centos.com |

1. 用ssh-copy-id将公钥复制到远程机器中，执行如表 2‑11所示命令。

表 2‑11将公钥复制到远程机器中

|  |
| --- |
| ssh-copy-id -i /root/.ssh/id\_rsa.pub master//依次输入yes,123456(root用户的密码)  ssh-copy-id -i /root/.ssh/id\_rsa.pub slave1  ssh-copy-id -i /root/.ssh/id\_rsa.pub slave2  ssh-copy-id -i /root/.ssh/id\_rsa.pub slave3 |

1. 验证SSH是否能够免密钥登录。

在master下分别输入ssh slave1，ssh slave2，ssh slave3，如图 2‑55所示，说明配置SSH免密码登录成功。

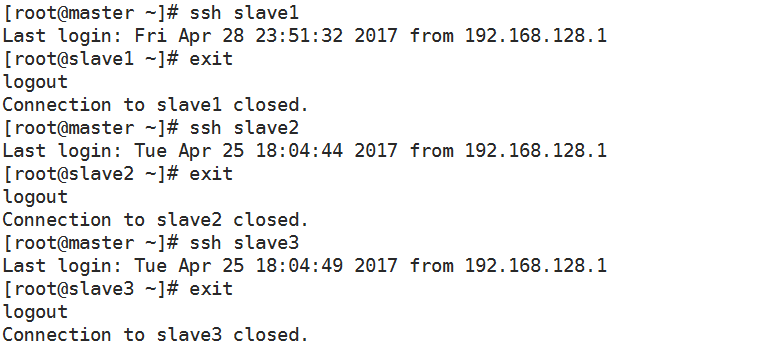


图 2‑55验证SSH无密钥登录

### 配置时间同步服务

NTP服务器是用来使计算机时间同步化的一种协议，它可以使计算机对其服务器或时钟源做同步化，提供高精准度的时间校正。Hadoop集群对时间要求很高，主节点与各从节点的时间都必须要同步。配置时间同步服务主要是为了进行集群间的时间同步。Hadoop集群配置时间同步服务的步骤如下所示。

1. 安装NTP服务。

在任务2.1的时候已经配置了yum源，可以直接使用yum来安装NTP服务，执行命令yum install –y ntp即可，若是最终出现了“Complete”信息，说明安装NTP服务成功。

1. 假设master节点为NTP服务主节点，那么其配置如下。

使用vim /etc/ntp.conf打开/etc/ntp.conf文件，注释掉server开头的行，并添加如代码 2‑15所示的内容。

代码 2‑15修改master主节点的ntp.conf文件

|  |
| --- |
| restrict 192.168.0.0 mask 255.255.255.0 nomodify notrap  server 127.127.1.0  fudge 127.127.1.0 stratum 10 |

1. 在slave1,slave2,slave2配置NTP，同样修改/etc/ntp.conf文件，注释掉server开头的行，并添加如代码清单 2‑16所示的内容。

代码2‑16修改子节点的ntp.conf文件

|  |
| --- |
| server master |

1. 执行命令service iptables stop & chkconfig iptables off永久性关闭防火墙，主节点和从节点都要关闭。
2. 启动NTP服务。
3. 在master节点执行：service ntpd start & chkconfig ntpd on，如图 2‑56所示，说明NTP服务启动成功。

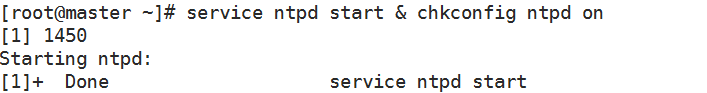


图 2‑56主节点启动NTP服务

1. 在slave1，slave2，slave3上执行命令ntpdate master即可同步时间，如图 2‑57所示。



图 2‑57子节点执行ntpdate master

1. 在slave1，slave2，slave3上分别执行service ntpd start & chkconfig ntpd on即可启动并永久启动NTP服务。如图 2‑58所示。

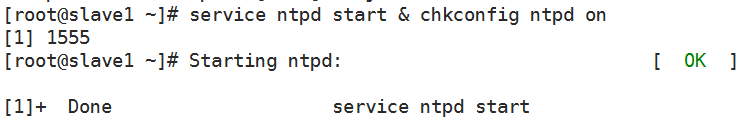


图 2‑58子节点启动ntp服务

### 启动关闭集群

做完Hadoop的所有配置后，即可执行格式化NameNode操作，该操作会在NameNode所在机器初始化一些HDFS的相关配置，并且该操作在集群搭建过程中只需执行一次，执行格式化之前可以先配置环境变量。

配置环境变量是在master，slave1，slave2，slave3上修改/etc/profile文件，在文件末尾修改如代码2‑17所示内容，文件修改完保存退出，执行source /etc/profile使配置生效。

代码2‑17设置环境变量

|  |
| --- |
| export HADOOP\_HOME=/usr/local/hadoop-2.6.4  export PATH=$HADOOP\_HOME/bin:$PATH:/usr/jdk1.8.0\_144/bin |

格式化只需执行命令hdfs namenode –format，若出现“Storage directory /data/hadoop/hdfs/name has been successfully formatted”提示，则格式化成功，如图 2‑59所示。

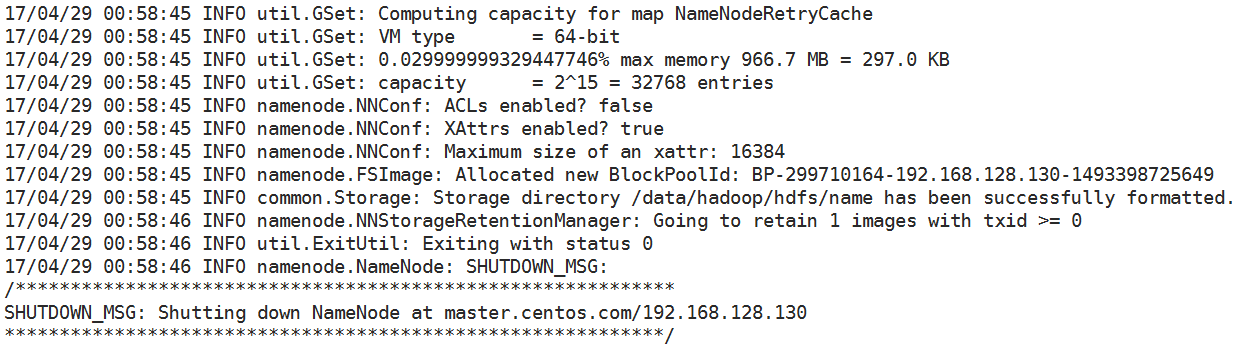


图 2‑59格式化成功提示

格式化完成之后即可启动集群，启动集群只需要在master节点，直接进入Hadoop安装目录，分别执行如表 2‑12所示的命令即可。

表 2‑12启动集群

|  |
| --- |
| cd $HADOOP\_HOME // 进入Hadoop安装目录  sbin/start-dfs.sh // 启动HDFS相关服务  sbin/start-yarn.sh // 启动YARN相关服务  sbin/mr-jobhistory-daemon.sh start historyserver // 启动日志相关服务 |

集群启动之后，在主节点master，子节点slave1，slave2，slave3分别执行jps，出现如图 2‑60所示的信息，说明集群启动没问题。

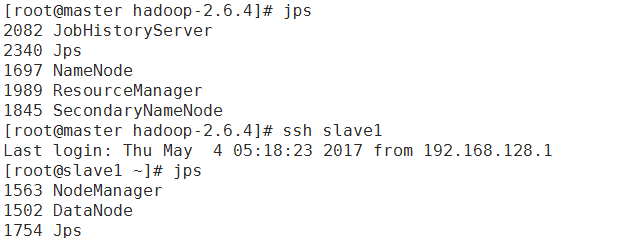


图 2‑60 jps查看进程

同理，关闭集群也只需要在master节点，直接进入Hadoop安装目录，分别执行如表 2‑13所示的命令即可。

表 2‑13关闭集群

|  |
| --- |
| cd $HADOOP\_HOME // 进入Hadoop安装目录  sbin/stop-yarn.sh // 关闭YARN相关服务  sbin/stop-dfs.sh // 关闭HDFS相关服务  sbin/mr-jobhistory-daemon.sh stop historyserver // 关闭日志相关服务 |

### 监控集群

Hadoop集群相关服务监控如表 2‑14所示。

表 2‑14 Hadoop集群相关服务

|  |  |  |
| --- | --- | --- |
| 服务 | Web 接口 | 默认端口 |
| NameNode | http://namenode\_host:port/ | 50070 |
| ResourceManager | http://resourcemanager\_host:port/ | 8088 |
| MapReduce JobHistory Server | http://jobhistoryserver\_host:port/ | 19888 |

修改本地文件C:\Windows\System32\drivers\etc\hosts文件，在文件末尾添加如代码 2‑18所示的内容。

代码 2‑18修改本地hosts文件

|  |
| --- |
| 192.168.128.130 master master.centos.com  192.168.128.131 slave1 slave1.centos.com  192.168.128.132 slave2 slave2.centos.com  192.168.128.133 slave3 slave3.centos.com |

1. HDFS监控。

在浏览器的地址栏输入<http://master:50070>，回车即可看到HDFS的监控界面，如图 2‑61所示。

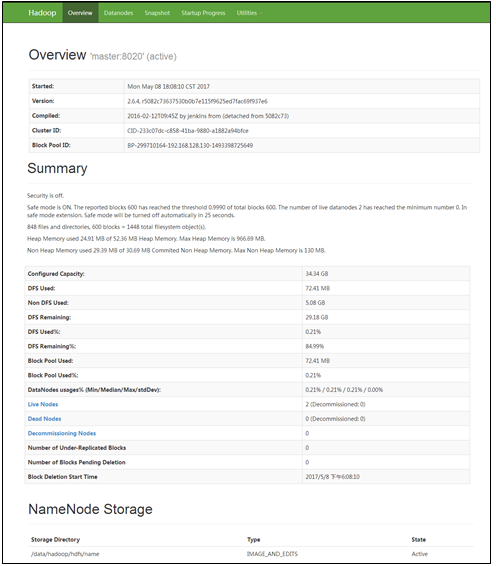


图 2‑61 HDFS监控1

如图 2‑62所示，单击“Utilities”——“Browse the file system”可以查看HDFS上的文件信息。

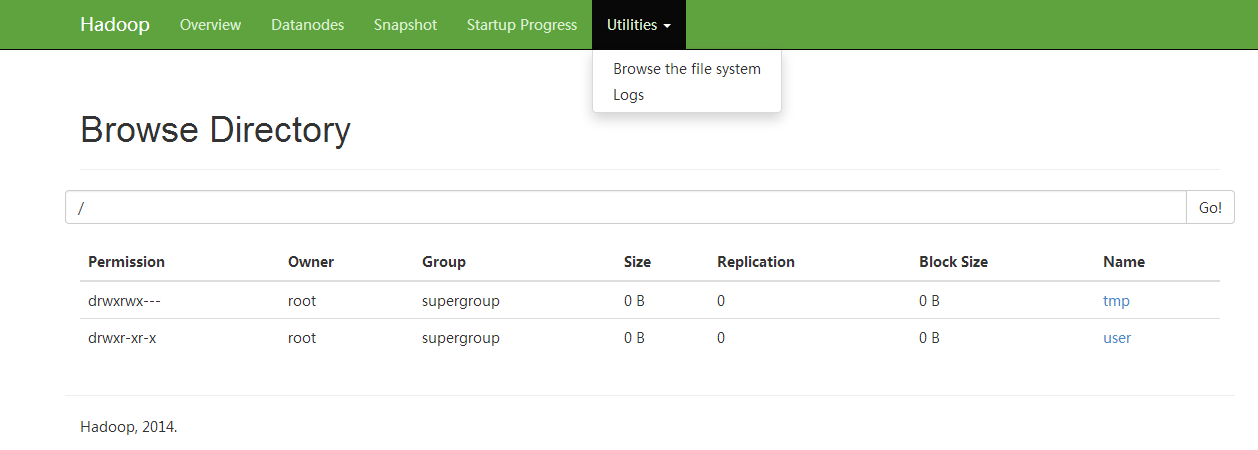


图 2‑62查看HDFS文件

在浏览器访问<http://master:50070/dfshealth.jsp>，得到如图 2‑62所示的页面，单击“Browse the filesystem”可以打开文件存储目录。如图 2‑63所示。而“NameNode Logs”为用户提供NameNode节点的log信息，如图 2‑64所示。

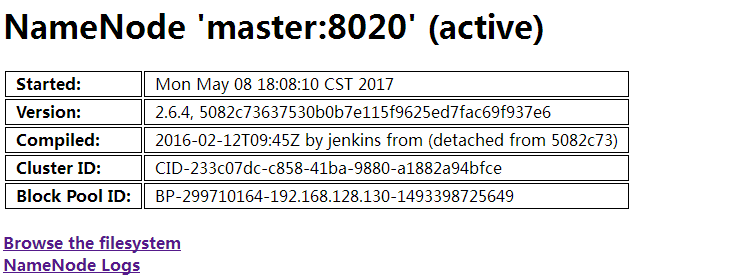


图 2‑63 HDFS监控2

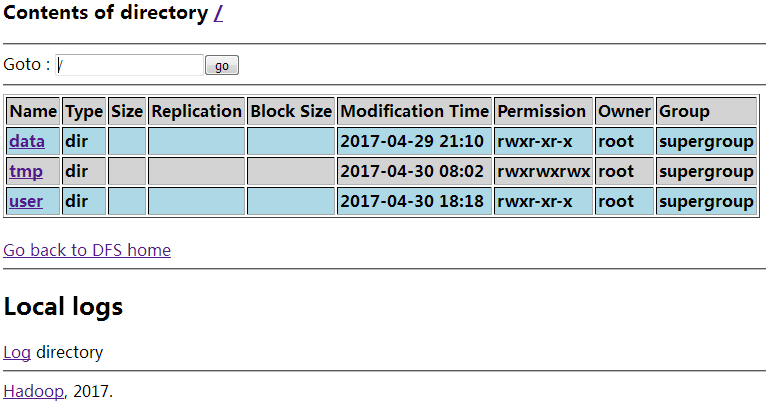


图 2‑64 HDFS文件存储目录

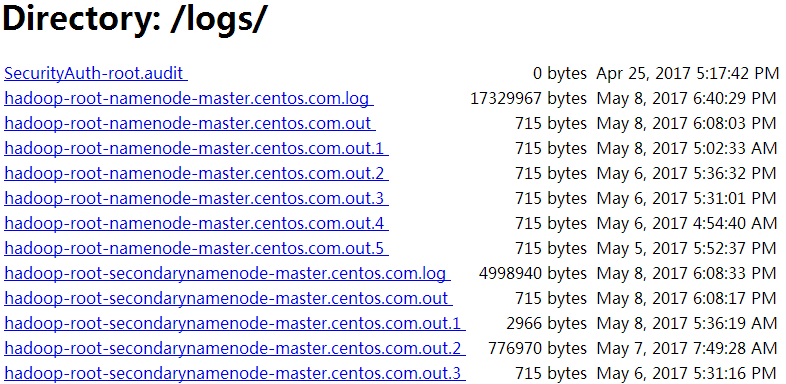


图 2‑65 NameNode节点的信息

1. YARN监控。

在浏览器的地址栏输入<http://master:8088>，即可看到YARN的监控界面，如图 2‑65所示。

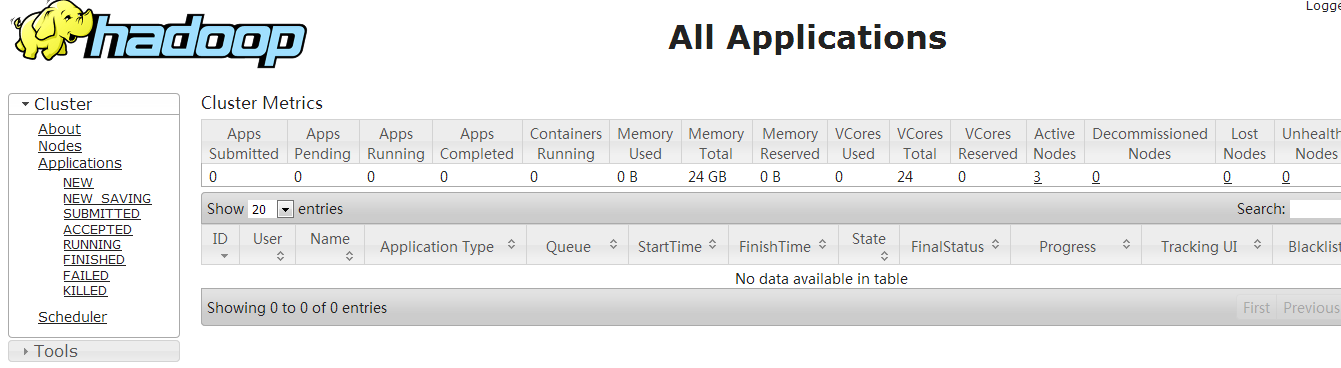


图 2‑66 YARN监控

1. 日志监控。

在浏览器的地址栏输入<http://master:19888>，即可看到日志监控界面，如图 2‑66所示。

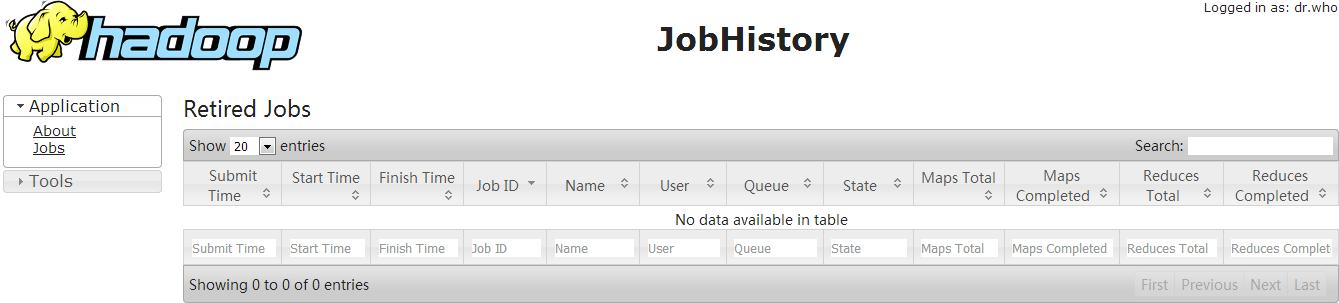


图 2‑67日志监控