

## 配置安装hadoop

### 一、需要的软件

- 1、虚拟机管理软件：VMware
- 2、系统：CentOS 7
- 3、ssh软件：Xshell 7
- 4、hadoop版本：2.7.7
- 5、Jdk版本：jdk8

### 二、安装配置Hadoop

注意：使用超级管理员root登录。

#### 1、配置静态网络，关闭防火墙

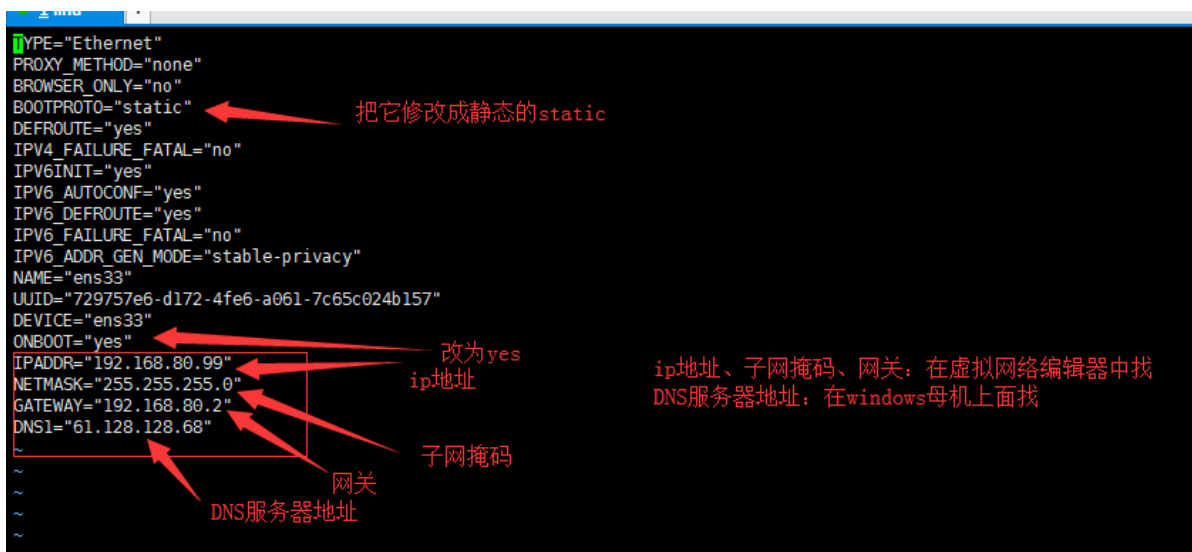
(1) 使用ping 命令检测网络是否连通    ping [www.baidu.com](http://www.baidu.com)    使用ctrl+c来停止命令

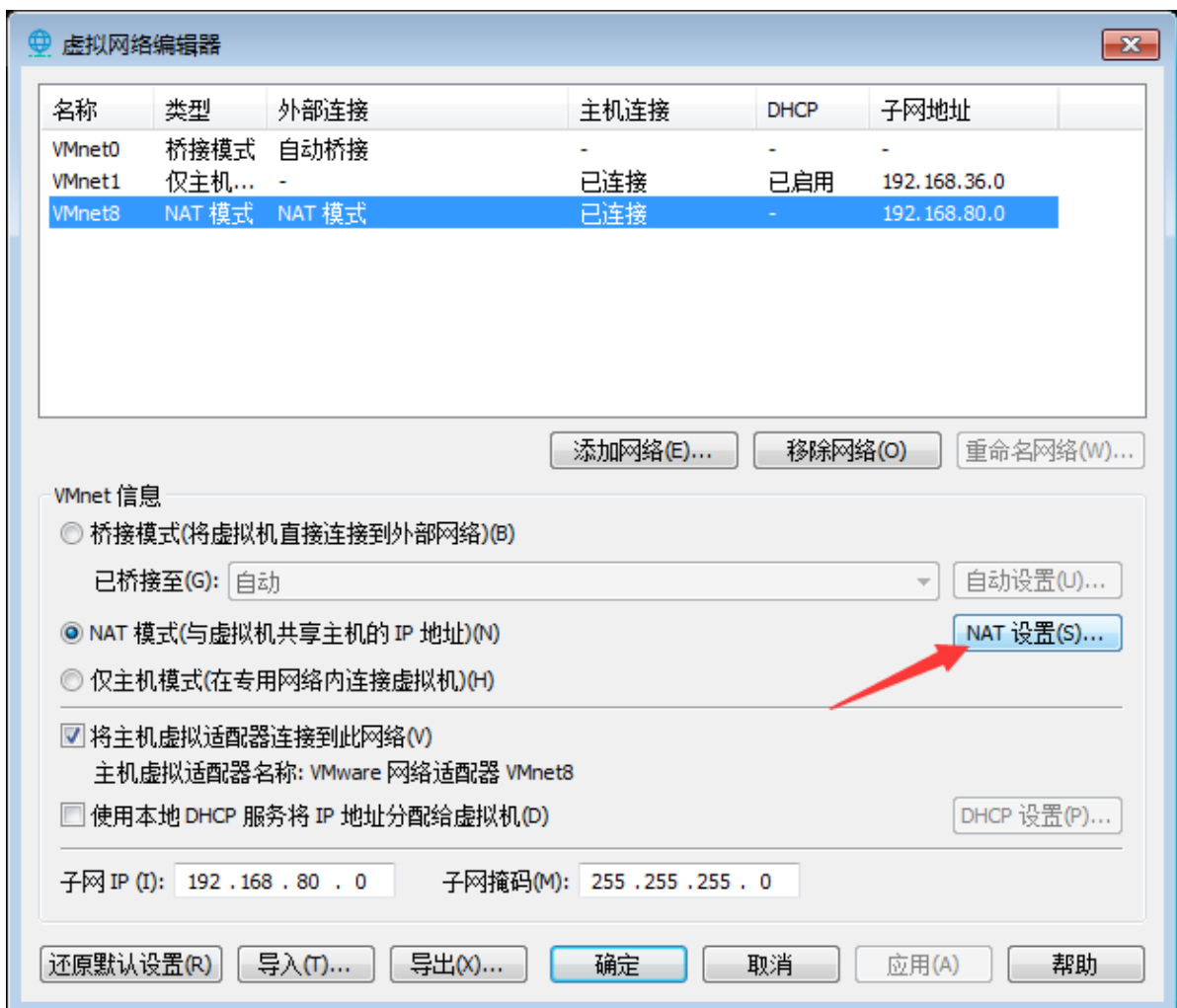
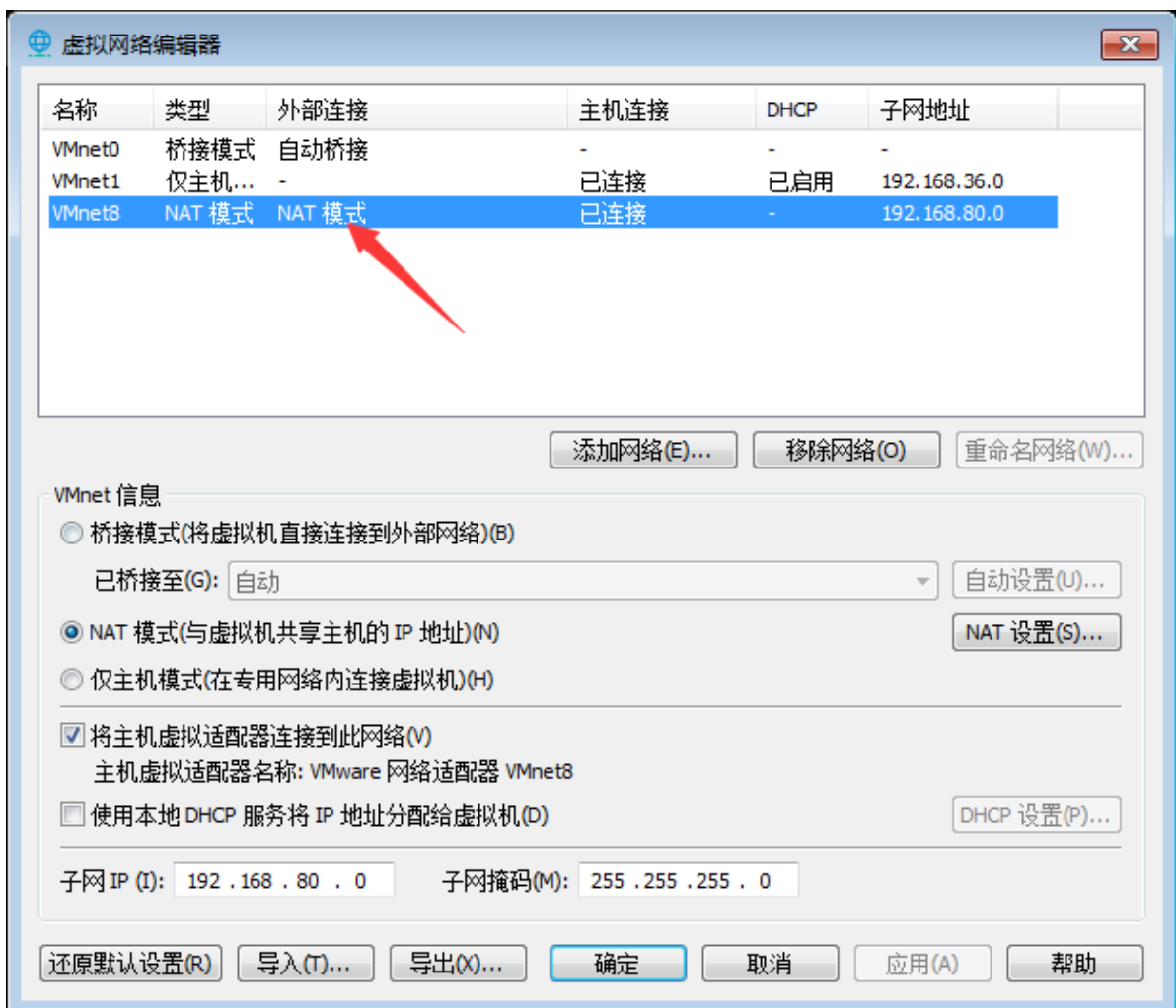
如果网络没有打开，打开网络：

```
service network restart
```

(2) 修改ip地址，设置为静态网络。

```
cd /etc
cd sysconfig
cd network-scripts
ll
#在network-scripts中，选择编辑ifcfg-ens33这个文件修改ip地址信息。
vim ifcfg-ens33
```





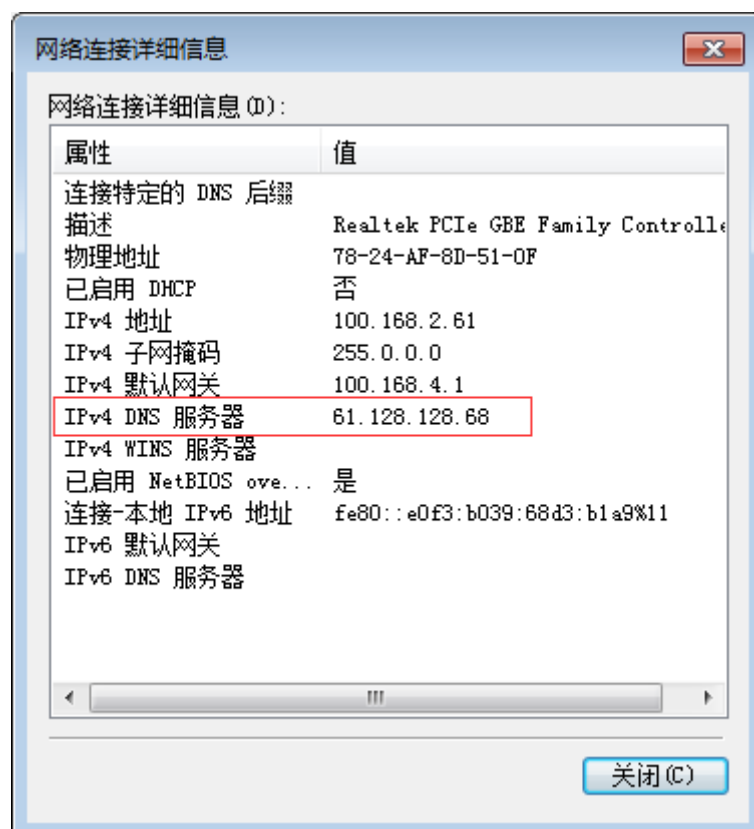


DNS服务器地址是当前windows的DNS服务器地址。



## 查看基本网络信息并设置连接





(3) 输入i 进入insert编辑

(4) 把BOOTPROTO="hdcp" 改为BOOTPROTO="static"静态网络ip

(5) 设置ONBOOT="yes"

点击vmware的【编辑】>【虚拟网络编辑器】>选择【VMnet8网络】>选择【NAT】按钮查看VMnet8 相关网络信息，包括子网ip、子网掩码、网关。

( 6 ) 添加master IP地址为：**IPADDR="192.168.50.3"**

( ip地址需要跟虚拟网络编辑器中的ip地址在同一个网段中 )

注意:s1和s2的ip地址也要修改，三个系统的ip地址前三位一样。最后一位不一样。

( 7 ) 添加子网掩码：**NETMASK="255.255.255.0"**

( 8 ) 添加网关：**GATEWAY="192.168.50.2"**

( 9 ) 添加自己宿主机的DNS: **DNS1="192.168.1.1"**

公共DNS:180.76.76.76

( 10 ) 保存退出：esc + :wq

( 11 ) 重启网络：

```
service network restart
```

( 12 ) 测试网络有没有连通

```
ping www.baidu.com
```

## 2、关闭防火墙

```
#关闭防火墙
systemctl stop firewalld
#禁用防火墙
systemctl disable firewalld
#查看防火墙状态
systemctl status firewalld
```

## 3.配置主机名

```
#给3台虚拟机设置主机名分别为master、s1和s2。
#在第一台机器操作
hostnamectl set-hostname master
#在第二台机器操作
hostnamectl set-hostname s1
#在第三台机器操作
hostnamectl set-hostname s2
```

设置完毕后需重启虚拟机：reboot

## 4.编辑hosts文件

hosts 配置文件是用来把主机名字映射到IP地址的方法，这种方法比较简单

( 1 ) 编辑hosts文件：

```
vi /etc/hosts
```

( 2 ) 进入编辑模式 i

( 3 ) 在最后一行添加

```
192.168.50.3 master
```

192.168.50.4 s1

192.168.50.5 s2

备注：这里是把三个Linux的ip地址保存到三个虚拟机去，相当于我们自己在手机里面存别人的电话号码一样的道理，这样相互之间就知道对应ip地址的机器是哪一台。**所以这个操作也是要在三台虚拟机都要进行的。**给ip地址起名字，几个机器需要互相连通，这样在连接几台机器的时候只需要使用机器名就行，不需要使用ip地址。

(4) 在s1和s2进行相同的操作。

```
#127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
#::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.80.99 master
192.168.80.100 s1
192.168.80.101 s2
```

(5) 使用ping命令，看是否能够进行相互的连通。

在master里面连通s1和s2。

```
ping s1
ping s2
```

## 5.文件传送xftp

## 6.配置SSH免密登录

在xshell中,Shift+insert键表示粘贴

把hadoop2.7.7压缩文件和jdk压缩文件传送到master中即可，s1和s2不需要传送。

**(1) 在各个机器 ( master、s1、s2 ) 家目录执行ssh-keygen -b 1024 -t rsa，输入2次回车，输入y/yes再继续回车**

```
ssh-keygen -b 1024 -t rsa
```

```
The key fingerprint is:
SHA256:+p/RA+bhacRdmrEJkRKtNQTEF1EsQAFo4NzXh3FAp\
The key's randomart image is:
+---[RSA 1024]-----+
|  .000=B@0=B.      |
|  oE.0+ =+.*..     |
|  0.. + .0... .    |
|      . . 0 B      |
|      S * *         |
|      . = =         |
|      * 0           |
|      . . 0 .       |
|      ..0           |
+----[SHA256]-----+
```

(2) ls -all :查看所有文件和文件夹

(3) 进入 .ssh目录中

```
cd .ssh
```

(4) 查看目录

id\_rsa:私钥 id\_rsa.pub:公钥

- 如何能在master中对s1和s2进行免密登录？需要把master的公钥放到s1和s2的authorized\_keys文件里

(5) 查看master的公钥

```
cat id_rsa.pub
```

(6) 在master的.ssh目录中执行

```
ssh-copy-id s1  
ssh-copy-id s2  
ssh-copy-id master
```

(7) 在s1的.ssh目录中执行

```
ssh-copy-id s2
```

(8) 在s2的.ssh目录中执行

```
ssh-copy-id s1
```

(9) 配置权限

在master、s1和s2的.ssh目录中执行该命令

```
chmod 600 authorized_keys
```

(10) 相互验证能否免密登录

```
ssh s1  
ssh s2  
#exit表示退出登录
```

## 7.设置时间同步

```
crontab -e  
0 1 * * * /usr/sbin/ntpdate cn.pool.ntp.org
```

## 8.解压jdk包和Hadoop包并安装jdk

(1) 解压压缩文件：解压hadoop文件和jdk文件，输入命令时可以用tab键补全

```
tar -zxvf hadoop-2.7.7.tar.gz  
tar -zxvf jdk-8u171-linux-x64.tar.gz
```

(2) 安装jdk，设置环境变量

两个环境变量配置文件是父子关系，后者会继承前者的环境变量配置。

/etc/profile	所有用户的环境变量（全局，父）
.bash_profile	当前用户的环境变量（局部，子）

#设置环境变量

```
vim /etc/profile
```

#在末尾追加环境变量

#如果你自己解压到其他的位置，那么就根据你自己的位置去写参数。

```
export JAVA_HOME=/root/jdk1.8.0_171
```

```
export PATH=$JAVA_HOME/bin:$PATH
```

#解释：输入路径就是为了方便找到bin目录的java。

#使环境变量生效

```
source /etc/profile
```

#可以查看java的版本

```
[root@master ~]# java -version
```

```
java version "1.8.0_171"
```

```
Java(TM) SE Runtime Environment (build 1.8.0_171-b11)
```

```
Java HotSpot(TM) 64-Bit Server VM (build 25.171-b11, mixed mode)
```

- 到此, master中的java配置已经结束了。

(3) 在s1和s2中不用再去安装，直接传送就好了

```
[root@master ~]# scp -r jdk1.8.0_171 root@s1:~/
```

```
[root@master ~]# scp -r jdk1.8.0_171 root@s2:~/
```

(4) 可以在s1和s2中查看传送过来的文件。

(5) 配置s1和s2的环境变量，参考步骤（2）。

## 9.配置Hadoop(7个文件)

在master上面配置好了直接将文件夹复制到从机上。

(1) 进入Hadoop的/etc目录下。

```
cd hadoop-2.7.7
```

```
cd etc
```

```
cd hadoop
```

```
ls
```

需要修改hadoop下的三个文件：yarn-env.sh、hadoop-env.sh、core-site.xml

(2) 修改hadoop-env.sh文件



```
vim hadoop-env.sh
#修改JAVA_HOME的路径
export JAVA_HOME=/root/jdk1.8.0_171
```

```
# Set Hadoop-specific environment variables here.

# The only required environment variable is JAVA_HOME. All others are
# optional. When running a distributed configuration it is best to
# set JAVA_HOME in this file, so that it is correctly defined on
# remote nodes.

# The java implementation to use.
export JAVA_HOME=/root/jdk1.8.0_171

# The jsvc implementation to use. Jsvc is required to run secure datanodes
# that bind to privileged ports to provide authentication of data transfer
# protocol. Jsvc is not required if SASL is configured for authentication of
# data transfer protocol using non-privileged ports.
```

保存退出。

( 3 ) 修改yarn-env.sh文件的JAVA\_HOME。

```
[root@master hadoop]# vim yarn-env.sh
#配置java_home
JAVA_HOME=/root/jdk1.8.0_171
```

```
export JAVA_HOME=/root/jdk1.8.0_171

# some Java parameters
# export JAVA_HOME=/home/y/libexec/jdk1.6.0/
if [ "$JAVA_HOME" != "" ]; then
    #echo "run java in $JAVA_HOME"
    JAVA_HOME=/root/jdk1.8.0_171
fi

if [ "$JAVA_HOME" = "" ]; then
```

保存退出。

( 4 ) 修改core-site.xml文件

```
vim core-site.xml

<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://master:9000</value>
  </property>
  <property>
    <name>hadoop.tmp.dir</name>
    <value>/root/hadoopdata</value>
  </property>
</configuration>
```

```
mkdir hadoopdata
```

```

<!-- Put site-specific property overrides in this file. -->
<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://master:9000</value>
  </property>
  <property>
    <name>hadoop.tmp.dir</name>
    <value>/root/hadoopdata</value>
  </property>
</configuration>
~

```

配置主机地址

集群中临时数据存放的位置

解释说明：

fs.defaultFS中的FS适合两个以上的主机使用，单点的话最好使用.name

hdfs://master:9000:master为主机名,9000为端口号

hadoop.tmp.dir：hadoop运行过程中产生的一些临时中间数据。

/home/zkpk/hadoopdata：存放临时数据的文件夹。

#### (6) 配置hdfs-site.xml

##### 1) 编辑hdfs-site.xml文件

```
[root@master hadoop]# vim hdfs-site.xml
```

##### 2) 加入以下内容：

```

<property>
  <name>dfs.replication</name>
  <value>2</value>
</property>

```

补充说明:一个文件会复制两个副本分别在s1和s2下。

```

<configuration>
  <property>
    <name>dfs.replication</name>
    <value>2</value>
  </property>
</configuration>
~

```

#### (7) 编辑mapred-site.xml文件

需要把mapred-site.xml.template文件的名字改为mapred-site.xml

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

```
vim mapred-site.xml
#粘贴以下内容
<configuration>
    <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
</configuration>
```

- 由于上面修改的是mapred-site.xml.template，而不是mapred-site.xml，所以我们可以把mapred-site.xml.template文件名修改为mapred-site.xml。

```
[root@master hadoop]# mv mapred-site.xml.template mapred-site.xml
```

#### ( 8 ) 编辑yarn-site.xml文件

```
vim yarn-site.xml
```

```
<configuration>

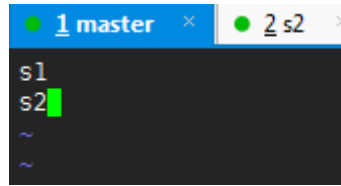
<!-- Site specific YARN configuration properties -->
    <property>
        <name>yarn.resourcemanager.hostname</name>
        <value>master</value>
    </property>
    <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
    </property>
</configuration>
```

```
-->
<configuration>
<!-- Site specific YARN configuration properties -->
<property>
    <name>yarn.resourcemanager.hostname</name>
    <value>master</value>
</property>
<property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
</property>
</configuration>
~
```

保存退出。

#### ( 9 ) 编辑 slaves文件

```
vim slaves
#填入两个从机名
s1
s2
```



( 10 ) 复制hadoop目录传到从机上

```
scp -r hadoop-2.7.7 s1:~/
scp -r hadoop-2.7.7 s2:~/
```

( 5 ) 在三台虚拟机中 /root下创建hadoopdata目录

( 11 ) 配置环境变量

**系统环境配置在master、s1和s2相同，以master 为例。**

进入根目录。

```
vim /etc/profile
#将下面的代码追加/etc/profile末尾:
export HADOOP_HOME=/root/hadoop-2.7.7
export PATH=$HADOOP_HOME/bin:$HADOOP_HOME/sbin:$PATH
```

使文件生效

```
source /etc/profile
```

s1和s2创建数据目录。

```
mkdir hadoopdata
```

## 10.hadoop格式化

格式化只能进行一次，在master上面进行。

```
hdfs namenode -format
```

## 11.启动hadoop

- 在master里面

```
start-all.sh
```

或者可以分两步启动

```
start-dfs.sh  
start-yarn.sh
```

- jps查看有没有启动成功

```
[root@master ~]# jps  
8866 Jps  
8551 NodeManager  
8441 ResourceManager  
8123 DataNode  
7852 NameNode  
8287 SecondaryNameNode
```

## 12.web ui查看hadoop集群

<http://master:50070>

<http://master:8088>