

Hadoop伪分布式/分布式平台搭建教程

- **声明：** 本搭建教程参考XMU的数据库实验室林子雨老师的搭建文档，附带了一点我们在安装时候遇到的问题和解决经验。
- **XMU安装指导文档网址：** <https://dblab.xmu.edu.cn/blog/2544/>

目录

▼ Hadoop伪分布式/分布式平台搭建教程

- 目录

▼ 1.Linux的安装

- 1.1 Linux的选择

▼ 1.2 Linux的安装

- 1.2.1 双系统安装

- 1.2.2 虚拟机安装

▼ 1.2.3 让你的虚拟机更好用：VMWare WorkStation PRO剪切板共享、共享文件夹等增强功能的实现

- 物理机和虚拟机的剪切板共享

- 共享文件夹

▼ 2.Hadoop单机/伪分布式集群的安装

- 2.1创建Hadoop用户

- 2.2更新apt

▼ 2.3安装ssh、配置ssh免密码登录

▼ 问题解决：正确配置ssh无密码登录之后还是不能免密登录

- 解决思路1：防火墙是否关闭

- 解决思路2：修改ssh配置文件

- 解决思路3：修改文件夹权限（ssh对.ssh文件夹的权限有比较严格的要求）

- 2.4安装Java环境

▼ 2.5安装和配置Hadoop

- 小技巧：使用Windows自带的SFTP实现物理机和虚拟机之间的文件传输

▼ 3.Hadoop分布式集群的安装

- 3.1网络配置

- 3.2无密码登录节点

- 3.3配置PATH变量

▼ 3.4配置集群/分布式环境并分发到各个节点

- 更改桥接模式

- 问题解决：在初次格式化namenode的时候不慎多次格式化使得主从节点不能正常启动

1.Linux的安装

1.1 Linux的选择

- 本实验建议采用 Ubuntu LTS 16.04 或 18.04
- 但是考虑到使用的相关包版本较低的情况，为了兼容性（如果是搭建分布式的话，那么参与搭建的几台机器版本要尽可能一致，以免出现各种奇奇怪怪的情况），首推 Ubuntu LTS 16.04，从 Ubuntu 中国官网下载，也可以寻找并下载 Kylin 版本。

1.2 Linux的安装

- 主要分为双系统和虚拟机安装

量力而行，如果已经装了双系统，那么就用双系统，否则建议使用虚拟机（毕竟谁也不想因为操作失误导致Ubuntu把原先的Windows抹除掉，虚拟机安装则相对安全一点）。

1.2.1 双系统安装

想要尝试双系统的，请自行上网搜索双系统安装教程。

1.2.2 虚拟机安装

- 关于 Virtual Box 与 VMWare WorkStation PRO的抉择

这两个软件，我们组当中均有装过，但是结合其他人的使用反馈和个人使用体验来看，更加推荐选择 VMWare WorkStation PRO，功能更强更好用，也不会有奇奇怪怪的小问题。

在本文当中，将着重介绍使用 VMWare WorkStation PRO 的安装，如果想要安装 Virtual Box 的，可以前往 XMU的数据实验室（开头已给出），参考他们的文档安装。

- VMWare WorkStation PRO 的安装

鉴于现在 VMWare 官网下载PRO版本需要登录并先购买才能下载，所以我在夸克网盘共享了 VMWare WorkStation PRO 17 的安装包供你们下载（网盘里面还包括了本次实验要下载的所有资源）。不要下载 VMWare Player，这个功能很少，用起来不方便。（主要资源来自XMU林子雨老师的教程，但是 ncbi-blast 和 mafft 是我们实验所用到的包）

链接：<https://pan.quark.cn/s/1204eac2e022>

提取码：qJeF

安装完毕之后，可以从网上寻找许可证代码，尝试激活 VMWare WorkStation PRO 17（多尝试几个，总归能找到一个对的）

- Ubuntu 16.04 LTS 的安装（可以自己上网搜索相关教程，这里我提供一个比较简单的叙述）

- 打开 VMWare Work Station PRO 17，文件-新建虚拟机，进入虚拟机新建向导。

- 选择 自定义安装，硬件兼容性按照默认
- 安装客户机操作系统选择 稍后安装操作系统
- 选择客户机操作系统，类别选 Linux，版本选 Ubuntu 64位
- 命名虚拟机，自己命名，然后位置看一眼，不建议装C盘（虚拟机占空间较大，可能会导致C盘爆满的情况，如果不想当C盘战士，尽量装别的盘，当然，没有分盘的人随意）

- 处理器配置，建议 2-1 或者 2-2，看自己电脑配置情况
- 此虚拟机内存，如果想要畅快地使用，建议分配 4-6GB，当然多多益善，我物理机是 32GB，所以我给了 8GB
- 网络类型、I/O控制器类型、磁盘类型、选择磁盘均保持默认选择即可
- 指定磁盘容量，空间富裕的，可以给 30-40GB，保持建议大小也可以；不要勾选 立刻分配所有磁盘空间（如果勾选了，那么是立刻从你磁盘上面划走相应大小的空间给虚拟机，这很占地方）；选择 将虚拟磁盘拆分成多个文件
- 指定磁盘文件，保持默认选择，完成虚拟机的创建
- 创建完成之后，先不要启动虚拟机，点击编辑虚拟机设置，点击 CD/DVD，勾选 使用ISO映像文件，找到虚拟机映像文件（可以去 Ubuntu 官网下载，也可以用我分享的链接里面下载），然后启动虚拟机，进入虚拟机安装程序。
 - 选择语言-中文（简体），点击 安装Ubuntu
 - 准备安装Ubuntu，比较建议勾选 安装Ubuntu时下载更新 以及 为图形或无线硬件，以及MP3和其他媒体安装第三方插件
 - 选择 清除整个磁盘并安装Ubuntu，注意，这会清除之前你选择的安装位置的文件夹中的所有内容，所以记得确认安装位置所在文件夹有无重要文件
 - 地方选择 Shanghai
 - 键盘布局选择 英语（美国） - 英语（美国）
 - 关于用户，正常创建就行，别忘记密码就行，注意最好用英文。后续搭建 Hadoop 的时候还会再创建一个新用户
 - 等待安装完成
- 系统安装完成之后，会提示你重新启动虚拟机，这个时候不要点击内部提供的重启按钮，而是找到 VMWare WorkStation PRO 上面横条的暂停按钮，点旁边的小三角，选择 关闭客户机
- 待客户机关闭之后，回到编辑虚拟机设置，点击 CD/DVD，改回 使用物理驱动器，（这一步类似于装完游戏之后把装游戏安装包的光盘弹出）
- 开启虚拟机后，可能会跳出

“无法连接虚拟设备 sata0:1，因为主机上没有相应的设备。您要在每次开启此虚拟机时都尝试连接此虚拟设备吗？”

的窗口，这个不用管，点击 是 即可，虚拟机装好后的初次启动会有点慢，需要耐心等待一会

1.2.3 让你的虚拟机更好用：VMWare WorkStation PRO剪切板共享、共享文件夹等增强功能的实现

物理机和虚拟机的剪切板共享

这可以让你直接从物理机复制文本到虚拟机当中

CTRL + ALT + T 打开终端

```
#安装open-vm-tools以使虚拟机自动调整桌面
sudo apt-get install open-vm-tools
#安装open-vm-tools-desktop以使虚拟机和宿主机能够互相拖拽文件，复制粘贴，不可直接拖拽到桌面
sudo apt-get install open-vm-tools-desktop
#安装后重启计算机方可生效
```

共享文件夹

此部分内容可以自己上网找文档实现，在本次实验当中，物理机和虚拟机之间的文件的传送通过SFTP实现，用不到共享文件夹。

2.Hadoop单机/伪分布式集群的安装

2.1创建Hadoop用户

- 创建新的hadoop用户

```
sudo useradd -m hadoop -s /bin/bash
#这条命令创建了可以登陆的 hadoop 用户，并使用 /bin/bash 作为 shell。
```

需要注意的几个点：

- **sudo命令:** 在接下来的操作当中会大量使用到sudo命令。sudo是Ubuntu中一种权限管理机制，管理员可以授权给一些普通用户去执行一些需要root权限执行的操作。当使用sudo命令时，会要求输入当前用户的密码。
- **密码:** 在Linux的终端中输入密码，终端是不会显示任何你当前输入的密码，也不会提示你已经输入了多少字符密码。而在Windows系统中,输入密码一般都会以 * 表示你输入的密码字符
- **输入法中英文切换:** Ubuntu中终端输入的命令一般都是使用英文输入。Linux中英文的切换方式是使用键盘 `shift` 键来切换，也可以点击顶部菜单的输入法按钮进行切换。Ubuntu有自带的中文输入法。
- **Ubuntu终端复制粘贴快捷键:** 在Ubuntu终端窗口中，复制粘贴的快捷键需要加上 `shift`，即粘贴是 `ctrl + shift + v`，粘贴也是如此。而原先的 `ctrl + c` 在终端中的作用是发送中断信号，让Shell中当前正在执行的前台作业终止执行（这在HIT CSAPP当中已涉及到）。

```
sudo passwd hadoop # 设置hadoop的密码
sudo adduser hadoop sudo # 为 hadoop 用户增加管理员权限，方便部署，避免一些对新手来说比较棘手的权限问题
```

- 最后注销当前用户，返回登陆界面。在登陆界面中选择刚创建的 hadoop 用户进行登陆。

2.2更新apt

- 用 hadoop 用户登录后，需要先更新一下 apt，后续使用 apt 安装软件，如果没更新可能有一些软件安装不了。执行如下命令：

```
sudo apt-get update
```

若出现如下 "Hash校验和不符" 的提示，可通过更改软件源来解决。若没有该问题，则不需要更改。从软件源下载某些软件的过程中，可能由于网络方面的原因出现没法下载的情况，那么建议更改软件源(使用国内镜像源，如阿里、清华等，更改后需要再次执行更新apt的指令)。

- 下载安装文本编辑器vim：

```
sudo apt-get install vim
```

2.3安装ssh、配置ssh免密码登录

集群、单节点模式都需要用到 SSH 登陆（类似于远程登陆，你可以登录某台 Linux 主机，并且在上面运行命令），Ubuntu 默认已安装了 SSH client，此外还需要安装 SSH server：

```
sudo apt-get install openssh-server
```

安装后可以尝试ssh登录到本机：

```
ssh localhost # 首次登录会有提示，输入yes就行了，之后按照提示输入hadoop用户的密码就行了
```

但这样登陆是需要每次输入密码的，我们需要配置成SSH无密码登陆。

- 首先退出刚才的ssh，回到原先的终端窗口，执行以下操作：

```
exit # 退出刚才的 ssh localhost
cd ~/.ssh/ # 若没有该目录，请先执行一次ssh localhost
ssh-keygen -t rsa # 会有提示，都按回车就可以
cat ./id_rsa.pub >> ./authorized_keys # 将公钥加入授权
```

- 这个时候再次执行 `ssh localhost` 命令，应该就可以无密码登录到本机了，如果还是要求输入密码，请转入到[问题解决:正确配置ssh无密码登录之后还是不能免密登录](#)

问题解决：正确配置ssh无密码登录之后还是不能免密登录

解决思路1：防火墙是否关闭

Ubuntu关闭防火墙及阻止防火墙自动启动（永久关闭）

```
sudo systemctl stop ufw
sudo systemctl disable ufw
```

再次尝试ssh无密码连接，如果还是不行，那么尝试 [解决思路2](#)

解决思路2：修改ssh配置文件

```
sudo vim /etc/ssh/sshd_config
```

在配置文件当中找到对应内容并修改，修改完成后内容如下：

```
.....
PermitRootLogin yes
StrictModes no
.....
RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile      %h/.ssh/authorized_keys
.....
PasswordAuthentication yes
.....
```

尝试重启ssh服务，再次尝试ssh无密码连接，如果还是不行，那么尝试 [解决思路3](#)

```
systemctl restart sshd.service
ssh Master
```

解决思路3：修改文件夹权限（ssh对.ssh文件夹的权限有比较严格的要求）

```
chmod 700 /home/username # username用你的当前用户名取代
chmod 700 ~/.ssh
chmod 600 ~/.ssh/authorized_keys
```

基本上到这一步都能解决问题，如果不能解决，建议重装系统或者向老师或助教寻求必要的帮助

2.4安装Java环境

推荐手动安装，因为这套实验环境版本不是最新的，手动安装可以更方便指定版本

Hadoop3.1.3 需要 JDK 版本在 1.8 及以上，需要按照下面步骤来手动安装 JDK1.8

- 从之前我给的夸克网盘链接当中下载 JDK1.8 的压缩包，从 Windows 本地通过SFTP传输进虚拟机当中（你也可以直接在虚拟机里面的浏览器里面下载，主要是我们小组有一个人下载好了，用u盘拷给每个人，所以是从Windows里面传进去的），如果不会SFTP传文件的请先跳转至[小技巧:使用windows自带的sftp实现物理机和虚拟机之间的文件传输](#)。
- 在本次实验当中，我们默认将JDK包上传到了/home/hadoop当中，在虚拟机当中执行如下命令，完成对压缩包的解压：

```
cd /usr/lib
sudo mkdir jvm # 创建/usr/lib/jvm目录用来存放JDK文件
cd ~ # 进入hadoop用户的主目录，刚才已经通过SFTP把JDK安装包jdk-8u162-linux-x64.tar.gz上传到该目录下
sudo tar -zxvf ./jdk-8u162-linux-x64.tar.gz -C /usr/lib/jvm # 把JDK文件解压到/usr/lib/jvm目录下
```

- JDK 文件解压缩以后，可以执行如下命令到 /usr/lib/jvm 目录查看一下

```
cd /usr/lib/jvm
ls
```

可以看到，在 /usr/lib/jvm 目录下有个 jdk1.8.0_162 目录。

- 下面继续执行如下命令，设置环境变量：

```
cd ~
vim ~/.bashrc
```

- 在 .bashrc 文件当中加入如下几行：

```
export JAVA_HOME=/usr/lib/jvm/jdk1.8.0_162
export JRE_HOME=${JAVA_HOME}/jre
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
export PATH=${JAVA_HOME}/bin:$PATH
```

- 执行以下命令，使环境配置生效：

```
source ~/.bashrc
```

- 验证 Java 是否成功安装

```
java -version #如果显示出对应的Java版本号，说明成功安装
```

2.5安装和配置Hadoop

- 执行以下命令，将Hadoop安装到 `/usr/local/` 路径下：

```
sudo tar -zxvf ~/下载/hadoop-3.1.3.tar.gz -C /usr/local
cd /usr/local/
sudo mv ./hadoop-3.1.3/ ./hadoop # 将文件夹名改为hadoop，便于后续操作，省的要记版本号
sudo chown -R hadoop ./hadoop # 修改文件权限
```

- Hadoop 解压后即可使用。输入如下命令来检查 Hadoop 是否可用，成功则会显示 Hadoop 版本信息：

```
cd /usr/local/hadoop
./bin/hadoop version
```

Hadoop的单机配置：

Hadoop的单机模式无需配置，装了就可以运行。

Hadoop的伪分布式配置：

Hadoop的配置文件位于 `/usr/local/hadoop/etc/hadoop/` 中，伪分布式需要修改2个配置文件 `core-site.xml` 和 `hdfs-site.xml`：

- 文件 `core-site.xml` 修改为如下内容：

```
<configuration>
  <property>
    <name>hadoop.tmp.dir</name>
    <value>file:/usr/local/hadoop/tmp</value>
    <description>Abase for other temporary directories.</description>
  </property>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://localhost:9000</value>
  </property>
</configuration>
```

- 文件 `hdfs-site.xml` 修改为如下内容：


```
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:/usr/local/hadoop/tmp/dfs/name</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:/usr/local/hadoop/tmp/dfs/data</value>
  </property>
</configuration>
```

注意:

Hadoop 的运行方式是由配置文件决定的（运行 Hadoop 时会读取配置文件），因此如果需要从伪分布式模式切换回非分布式模式，需要删除 core-site.xml 中的配置项。

此外，伪分布式虽然只需要配置 fs.defaultFS 和 dfs.replication 就可以运行（官方教程如此），不过若没有配置 hadoop.tmp.dir 参数，则默认使用的临时目录为 /tmp/hadoop-hadoop，而这个目录在重启时有可能被系统清理掉，导致必须重新执行 format 才行。所以我们进行了设置，同时也指定 dfs.namenode.name.dir 和 dfs.datanode.data.dir，否则在接下来的步骤中可能会出错。

- 配置完成后，执行namenode的格式化：

```
cd /usr/local/hadoop
./bin/hdfs namenode -format
```

- 启动和停止Hadoop：

```
cd /usr/local/hadoop
./sbin/start-dfs.sh      # 启动
./sbin/stop-dfs.sh       # 停止
```

注意:

因为没有在环境变量里面配置Hadoop的相关路径，所以以上命令都需要进入到Hadoop的文件夹bin和sbin(bin指的是binary，意为存放二进制可执行文件的文件夹)执行相应的程序/脚本，如果需要在任意路径当中都可以直接执行Hadoop的有关命令，可以在环境变量中(`vim ~/.bashrc`)加入如下内容：

```
export PATH=$PATH:/usr/local/hadoop/bin:/usr/local/hadoop/sbin
```

添加完成后保存退出，执行命令 “`source ~/.bashrc`” 让配置生效。

小技巧：使用Windows自带的SFTP实现物理机和虚拟机之间的文件传输

- 在Windows中 `win + r`，输入 `cmd` 唤起终端，使用如下指令连接用户名为 `hadoop` 虚拟机：

```
sftp hadoop@<这里用虚拟机的ip取代> # 关于ip的获取，在虚拟机终端当中使用ifconfig指令即可获取；连接时会要求你输入密码
```

- 常用指令大全：

```
pwd # 查看当前远程工作目录（也就是虚拟机的）
ls # 列出当前远程工作目录下的文件列表
cd # 切换远程工作目录
```

```
lpwd # 查看当前本地工作目录（也就是物理机的）
lls # 列出当前本地工作目录下的文件列表
lcd # 切换本地工作目录
```

```
put <文件名> #从本地上传指定文件到远程
get <文件名> #从远程获取指定文件到本地
```

3.Hadoop分布式集群的安装

Hadoop分布式集群的配置流程大致如下（前三条和单机/伪分布式操作类似）：

- 步骤1：选定一台机器作为 Master节点；
- 步骤2：在Master节点上创建hadoop用户、安装SSH服务端、安装Java环境；
- 步骤3：在Master节点上安装Hadoop，并完成配置；
- 步骤4：在其他Slave节点上创建hadoop用户、安装SSH服务端、安装Java环境；
- 步骤5：将Master节点上的“/usr/local/hadoop”目录复制到其他Slave节点上；
- 步骤6：在Master节点上开启Hadoop；

其实可以先参照单机/伪分布式的安装指导，做到完成 Java 的安装那一步，再来做这个

还有，可以自己通过vmware安装多个虚拟机，使用NAT模式，这样每个虚拟机都会获得一个虚拟的ip地址，虚拟机之间可以实现相互通讯，可以以这种方式安装分布式的Hadoop集群（不过这样本质上算力没有发生变化，因为这些虚拟机节点都使用同一台机器的硬件，但是可以感受一下分布式）

3.1网络配置

- 首先创建名为hadoop的用户（参考单机/伪分布式的安装）
- 将虚拟机的网络连接模式改为桥接模式（虚拟机设置 - 网络适配器 -勾选 桥接模式）[更改桥接模式](#)
- 修改主机名（Master， Slave1，， 修改完成记得用 `reboot` 指令重启一下虚拟机，反正一定要在在测试节点连通性之前重启一次）

```
sudo vim /etc/hostname
```

- 增加映射关系

```
sudo vim /etc/hosts
```

- 测试节点连通性

```
ping Master
```

#在从节点的终端输入，尝试连接主节点，ctrl+c终止ping命令

```
ping Slave1
```

#在主节点的终端输入，尝试连接从节点，ctrl+c终止ping命令

3.2无密码登录节点

- 首先生成Master节点的公钥，在Master节点上面执行以下命令：

```
cd ~/.ssh  
rm ./id_rsa* # 删除以前的rsa密钥  
ssh-keygen -t rsa # 一直按回车就行了
```

- 尝试ssh无密码登录到Master节点，在Master节点上面执行以下命令：

```
cat ./id_rsa.pub >> ./authorized_keys # 将公钥加入authorized_keys文件当中  
ssh Master
```

- 首次登录需要输入yes确认将Master添加到文件known_hosts当中，此时已经不需要输入密码，如果要求输入密码，请转到[问题解决:正确配置ssh无密码登录之后还是不能免密登录](#)
- 如果在Master节点上能ssh免密码登录Master后，将公钥传输到Slave1.....等其他节点上

```
scp ~/.ssh/id_rsa.pub hadoop@Slave1:/home/hadoop/ # 执行时要求输入对应节点的密码（因为此时还不能免密登录到
```

- 传输完公钥后，在对应节点上执行以下命令：

```
mkdir ~/.ssh  
cat ~/id_rsa.pub >> ~/.ssh/authorized_keys  
rm ~/id_rsa.pub
```

- 在Master节点上测试ssh免密登录其他节点：

```
ssh Slave1
```

3.3配置PATH变量

- 在Master节点上执行以下命令：

```
sudo vim ~/.bashrc
```

- 加入以下内容：

```
export PATH=$PATH:/usr/local/hadoop/bin:/usr/local/hadoop/sbin
```

- 保存退出后执行以下命令使配置生效：

```
source ~/.bashrc
```

3.4配置集群/分布式环境并分发到各个节点

- 配置集群/分布式模式需要修改 `/usr/local/hadoop/etc/hadoop` 目录下的配置文件，包括 `workers`、`core-site.xml`、`hdfs-site.xml`、`mapred-site.xml`、`yarn-site.xml` 共5个文件。
 - (1)修改文件workers：
本实验将Master节点作为名称节点使用，故将workers文件中原来的localhost删除，添加如下内容（如果Master节点要作为数据节点一部分使用，那么无需删除）：

```
Slave1
..... # 将所有子节点名称加入，之间用回车换行隔开
```

- (2)修改文件core-site.xml

```
<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://Master:9000</value>
  </property>
  <property>
    <name>hadoop.tmp.dir</name>
    <value>file:/usr/local/hadoop/tmp</value>
    <description>Abase for other temporary directories.</description>
  </property>
</configuration>
```

- (3)修改文件hdfs-site.xml

其中value的值代表冗余因子，表示文件备份数量，一般默认为3，不过也可以参考所使用的具体节点数来设定

```
<configuration>
  <property>
    <name>dfs.namenode.secondary.http-address</name>
    <value>Master:50090</value>
  </property>
  <property>
    <name>dfs.replication</name>
    <value>3</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:/usr/local/hadoop/tmp/dfs/name</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:/usr/local/hadoop/tmp/dfs/data</value>
  </property>
</configuration>
```

- (4)修改文件mapred-site.xml

在目录下有一个 `mapred-site.xml.template`，修改文件名称为 `mapred-site.xml`，然后把 `mapred-site.xml` 文件配置成如下内容：(但是当时我们在配置的时候，已经没有template的后缀了)

```

<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
  <property>
    <name>mapreduce.jobhistory.address</name>
    <value>Master:10020</value>
  </property>
  <property>
    <name>mapreduce.jobhistory.webapp.address</name>
    <value>Master:19888</value>
  </property>
  <property>
    <name>yarn.app.mapreduce.am.env</name>
    <value>HADOOP_MAPRED_HOME=/usr/local/hadoop</value>
  </property>
  <property>
    <name>mapreduce.map.env</name>
    <value>HADOOP_MAPRED_HOME=/usr/local/hadoop</value>
  </property>
  <property>
    <name>mapreduce.reduce.env</name>
    <value>HADOOP_MAPRED_HOME=/usr/local/hadoop</value>
  </property>
</configuration>

```

◦ (5)修改文件 yarn-site.xml

```

<configuration>
  <property>
    <name>yarn.resourcemanager.hostname</name>
    <value>Master</value>
  </property>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
</configuration>

```

- 上述5个文件全部配置完成以后，把 Master 节点上的 /usr/local/hadoop 文件夹复制到各个节点上。首先在 Master 节点上执行如下命令：

```
cd /usr/local
sudo rm -r ./hadoop/tmp
sudo rm -r ./hadoop/logs/*
tar -zcf ~/hadoop.master.tar.gz ./hadoop
cd ~
scp ./hadoop.master.tar.gz Slave1:/home/hadoop
..... # 复制到每个节点上面
```

- 在每个对应的节点上都进行如下操作：

```
sudo rm -r /usr/local/hadoop
sudo tar -zxvf ~/hadoop.master.tar.gz -C /usr/local
sudo chown -R hadoop /usr/local/hadoop
```

- 首次启动Hadoop集群时，需要先在Master节点执行名称节点的格式化（但是后续再次启动不需要，多次格式化会导致集群不能正常启动，主从节点之间不能相互识别和通讯，如果不慎多次格式化，请看[问题解决：在初次格式化namenode的时候不慎多次格式化使得主从节点不能正常启动](#)）

```
hdfs namenode -format
```

- 现在启动hadoop，在Master节点上执行如下命令：

```
start-dfs.sh
start-yarn.sh
mr-jobhistory-daemon.sh start historyserver
```

- 通过命令jps可查看各节点启动进程。正确启动后，在Master节点上可看到NameNode、ResourceManager进程等，在Slave节点上可看到DataNode和NodeManager进程等。在Master节点上通过命令“hdfs dfsadmin -report”查看数据节点是否正常启动（只要 Live Datanode 不为0，就说明正常启动，而完全正确启动所有节点的情况应当为 Live Datanode 数=之前设置的 value 值）。
- 也可以在浏览器中输入 <http://master:9870/>，查看各节点的状态，比较直观。
- hadoop集群的停止，只需要在Master节点上面输入以下指令：

```
stop-dfs.sh
stop-yarn.sh
mr-jobhistory-daemon.sh stop historyserver
```

或者

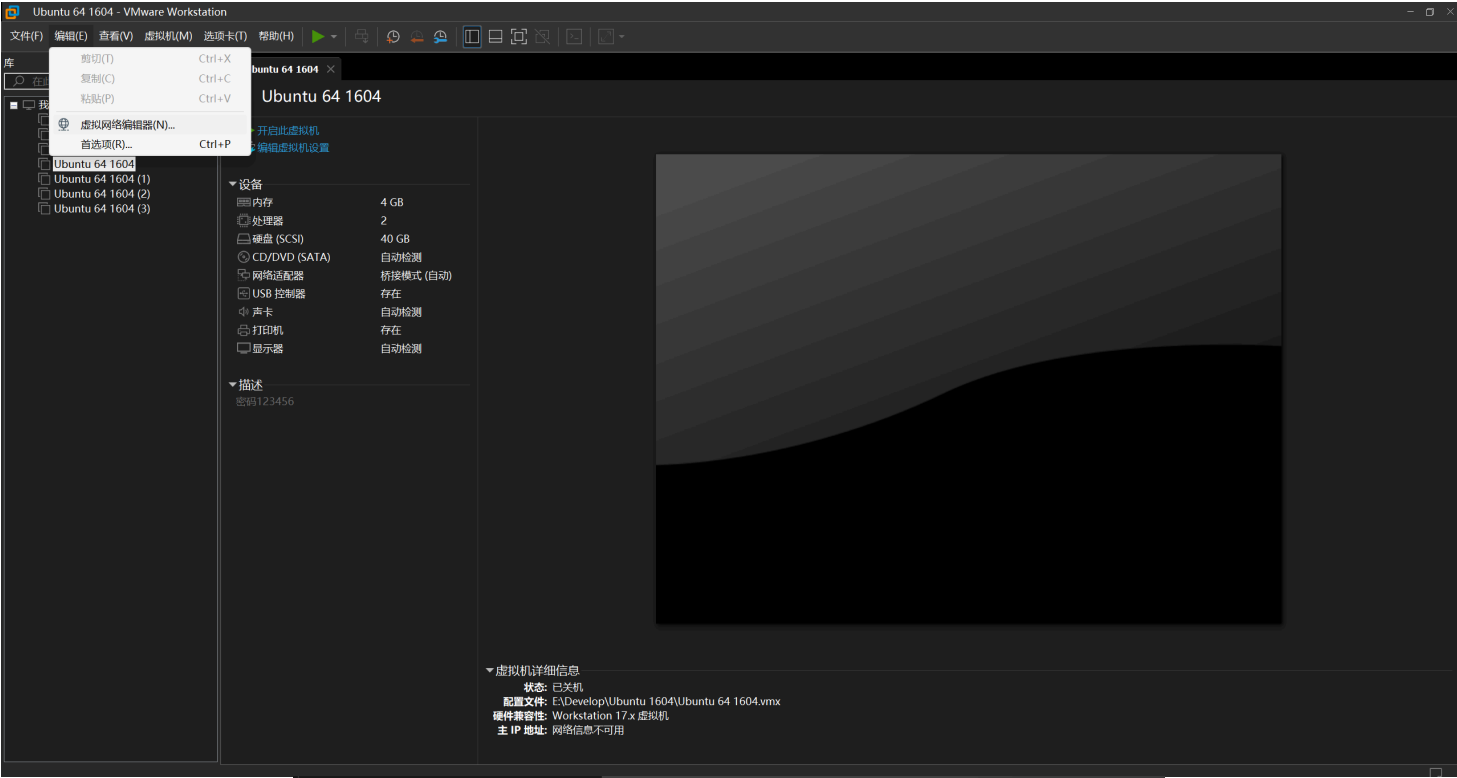
```
stop-all.sh
mr-jobhistory-daemon.sh stop historyserver
```

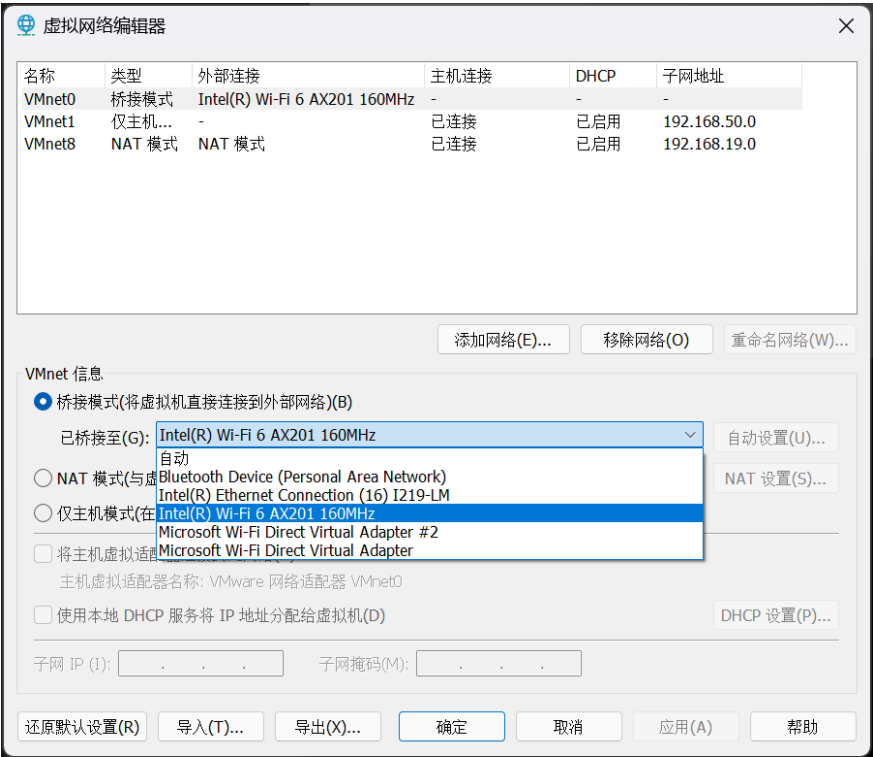
更改桥接模式

- 在这里我仅讲述我们在配置过程中的操作，网上类似的教程很多，方法不唯一也不一定对每台机子都适用，大家看着来。
在解决问题的过程中，我们发现，如果电脑连接的是校园网一类的登录才能上网/ip是动态分配的网络，那么切换桥接模式之后，有很大可能虚拟机是上不了网的，而且每次重新连接校园网，ip都会重新分配，导致每次记录节点名称和ip地址对应关系的文件都要重新修改一遍
- 建议开热点，让所有节点都连接同一个热点，这样既可以避免桥接模式带来的虚拟机无法上网，也可以避免因校园网重新连接就会重新分配ip地址需要不断修改相应配置文件的问题。我是通过使用我的平板的WLAN信号桥(平板是华为的，不知道其他牌子的有没有这个功能，本质上是WLAN共享，平板连的是校园网，但是平板分享出来的网每个人连上去之后ip就是固定的了，虚拟机也可以正常上网)来实现的。
- 虚拟机配置里面，选择网络模式为 桥接模式 ，并勾选 复制物理网络的连接状态

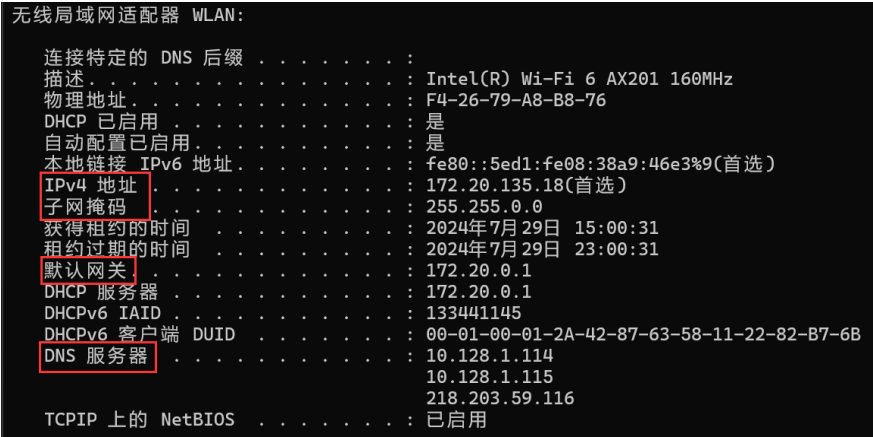


- 然后再vmware界面，点击左上角的 编辑 - 虚拟网络编辑器 ，点击 更改设置 ，然后选择你的物理机的网卡

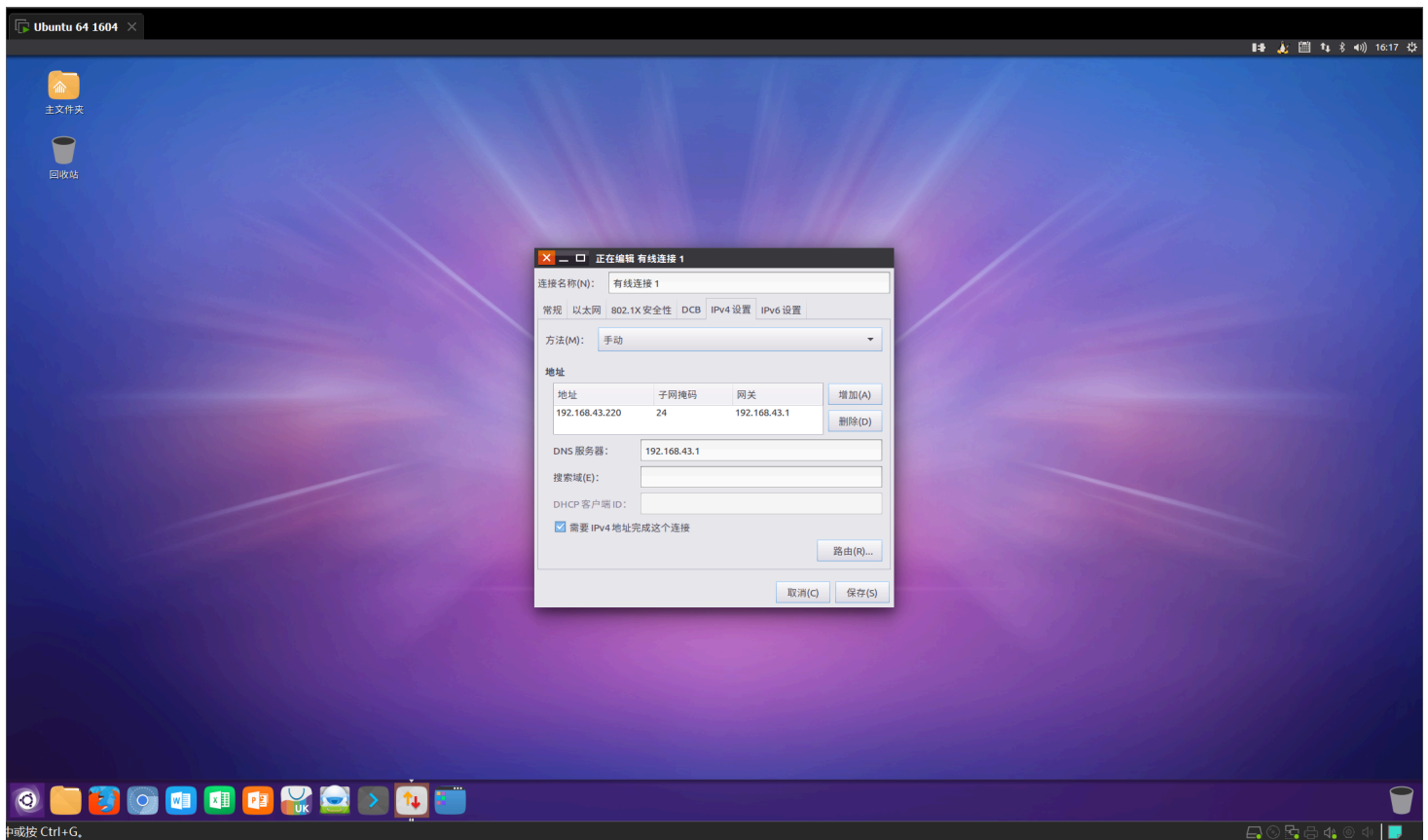




- 完成之后启动虚拟机，现在Windows的cmd当中使用命令 `ipconfig /all`，查看内容如下，记住这几个红框标出来的值：



- 在虚拟机里面进行网络编辑（可能是写这篇文章的时候连接的是校园网，所以DNS服务器有多个，开热点的话应该只有一个），按照图片来填就行了，方法勾选的是手动，ipv6设置我勾选的方法是忽略：



问题解决：在初次格式化namenode的时候不慎多次格式化使得主从节点不能正常启动

在配置hadoop文件 `core-site.xml` 的时候，有一组参数 `hadoop.tmp.dir`，它的值指定的是配置hadoop的临时目录，我们只需要在每个节点上将这个目录删除，然后重新格式化即可。

```
<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://Master:9000</value>
  </property>
  <property>
    <name>hadoop.tmp.dir</name>
    <value>file:/usr/local/hadoop/tmp</value>
    <description>Abase for other temporary directories.</description>
  </property>
</configuration>
```

执行以下指令以删除tmp目录，每个节点都要做：

```
cd /usr/local/hadoop
rm -rf tmp
```

然后重新初始化，问题解决

```
hdfs namenode -format
```