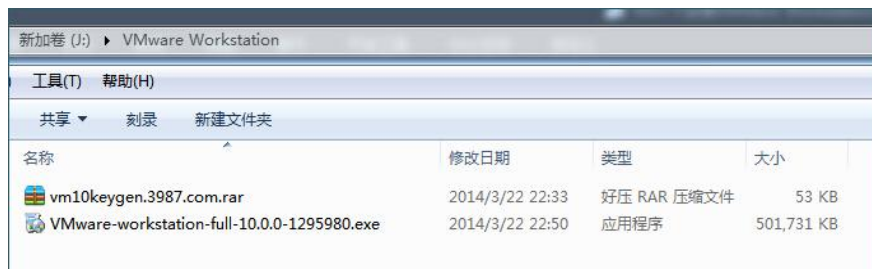


第 1 章 安装 VMWare Workstation 10

在软件包中找到“software\vmware”目录并进入该目录，如下所示：



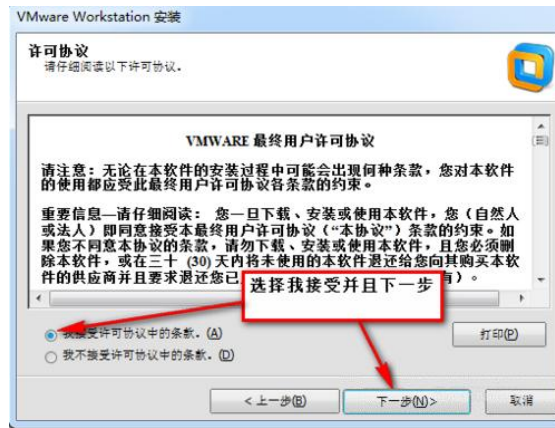
点击“VMware-workstation-full-10.0.0-1295980.exe”安装



等待安装软件检测和解压以后，出现如下界面，直接单击下一步即可。

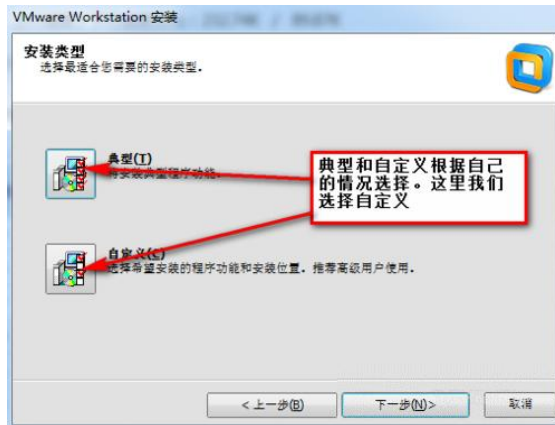


选择我同意选项，直接下一步。



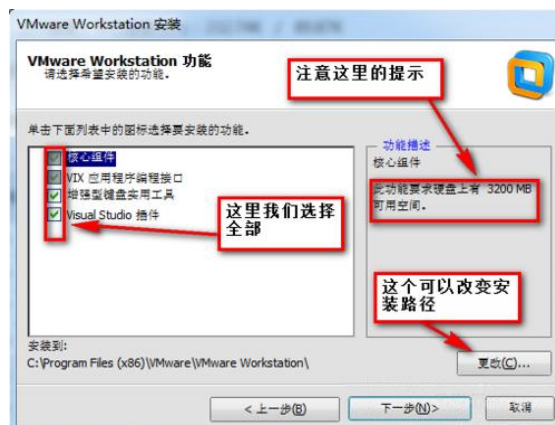
4

典型安装和自定义安装，可根据自己的情况酌情选择。这里我们选择自定义安装。



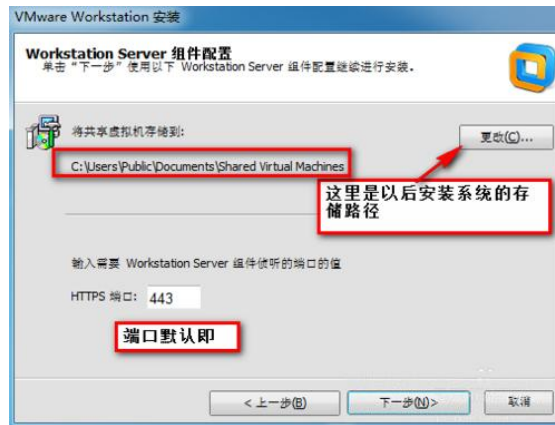
5

选择自定义以后，根据自己的情况选择自己需要的功能。这里我们选择全部。



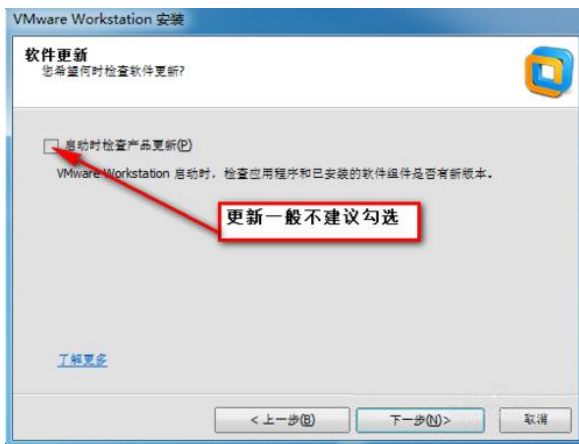
6

我们可以更改软件的安装路径，端口默认即可。



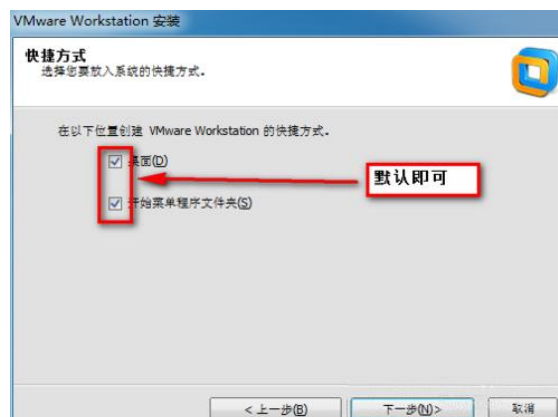
7

出现如下图的选择框框，一般不建议勾选。



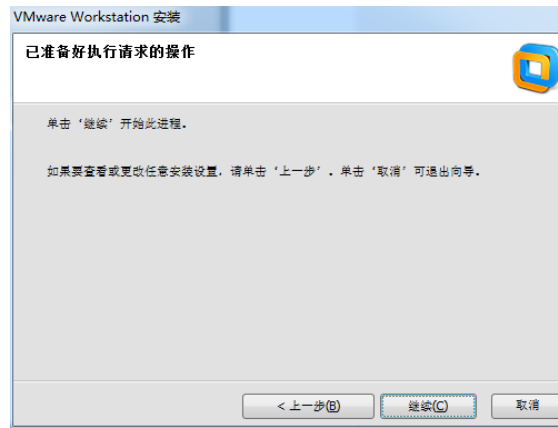
8

出现下图提示选择默认的即可。

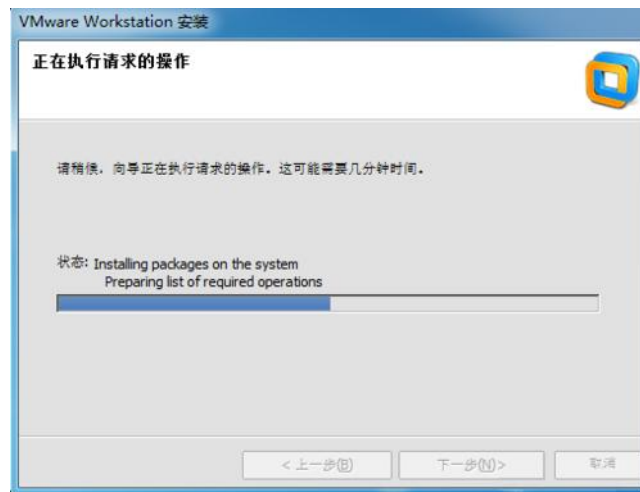


9

单击下一步，即可安装。



点击“继续”按钮

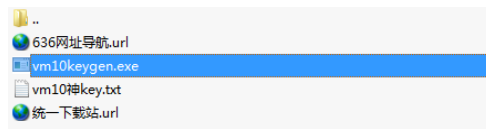


10

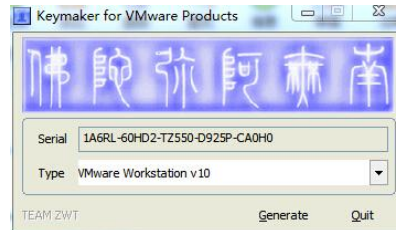
软件安装成功，如下图所示。



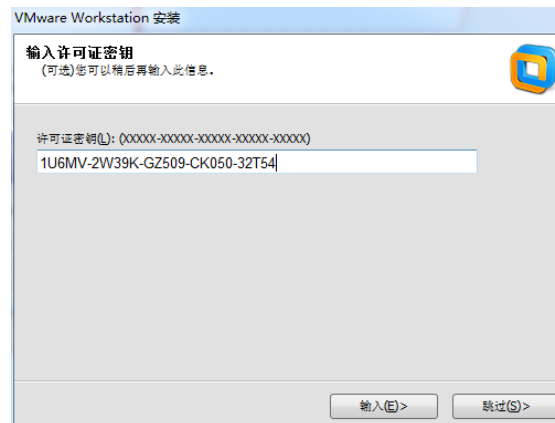
安装完成后，要求输入注册码，打开压缩文件中的算号器，



拷贝粘贴注册码



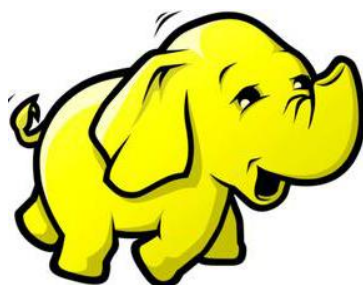
输入注册码:



点击输入后，出现:



点击完成，VMware 安装过程结束。



第 2 章

VMware 10 安装 CentOS 6

主要内容

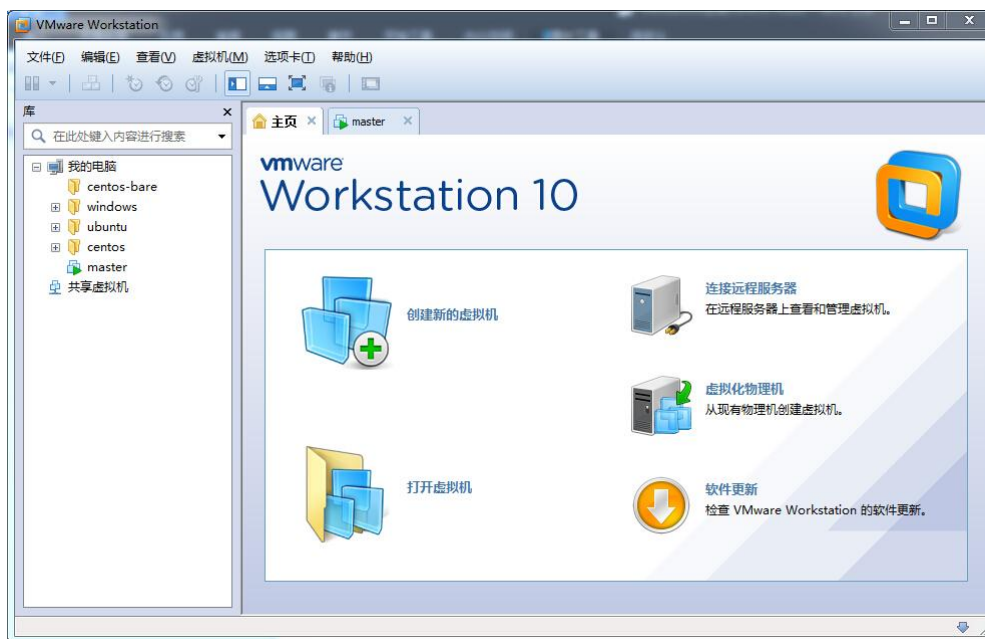
- CentOS 系统安装
- 安装中的关键问题
- 克隆 HadoopSlave
- 安装 SSH Secure Shell Client 传输软件



第2章 VMware 10 安装 CentOS 6

2.1 CentOS 系统安装

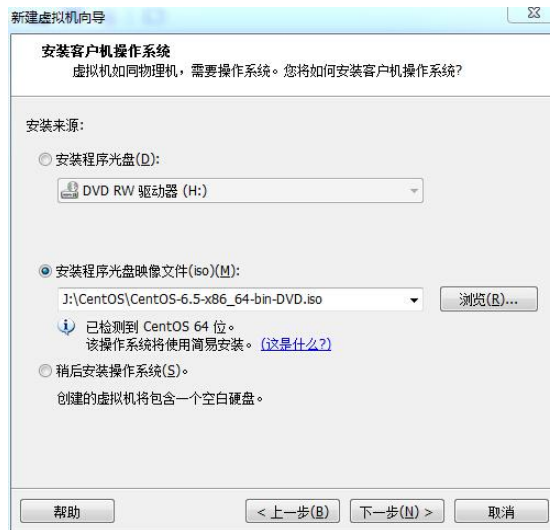
打开 VMware Workstation 10



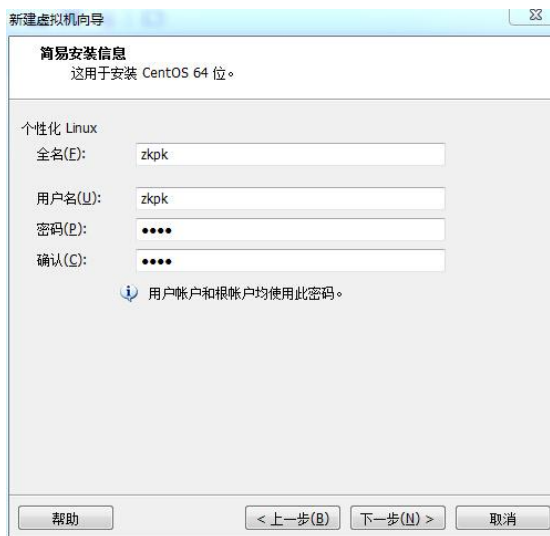
点击文件->新建虚拟机



选择典型（推荐）（T）选项，点击“下一步（N）>”

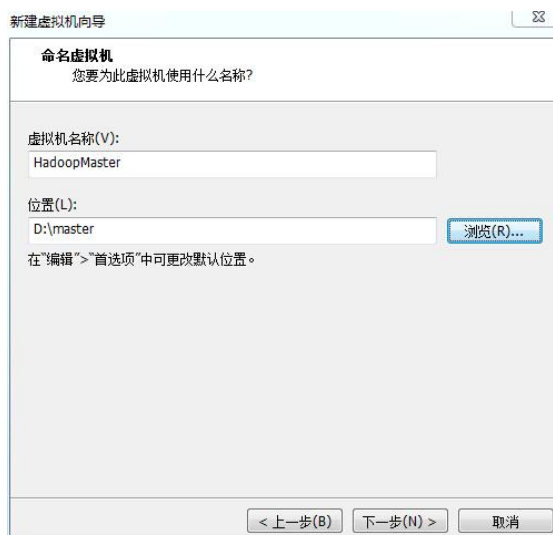


选择“安装程序光盘映像文件(iso)(M)”，选择指定的 CentOS 系统的.iso 文件，点击“下一步(N)>”



填写下面的信息，点击“下一步(N)>”，

全名: zkpk 用户名: zkpk 密码: zkpk 确认: zkpk



虚拟机名称 (V): HadoopMaster，选择安装位置，点击“下一步(N)>”

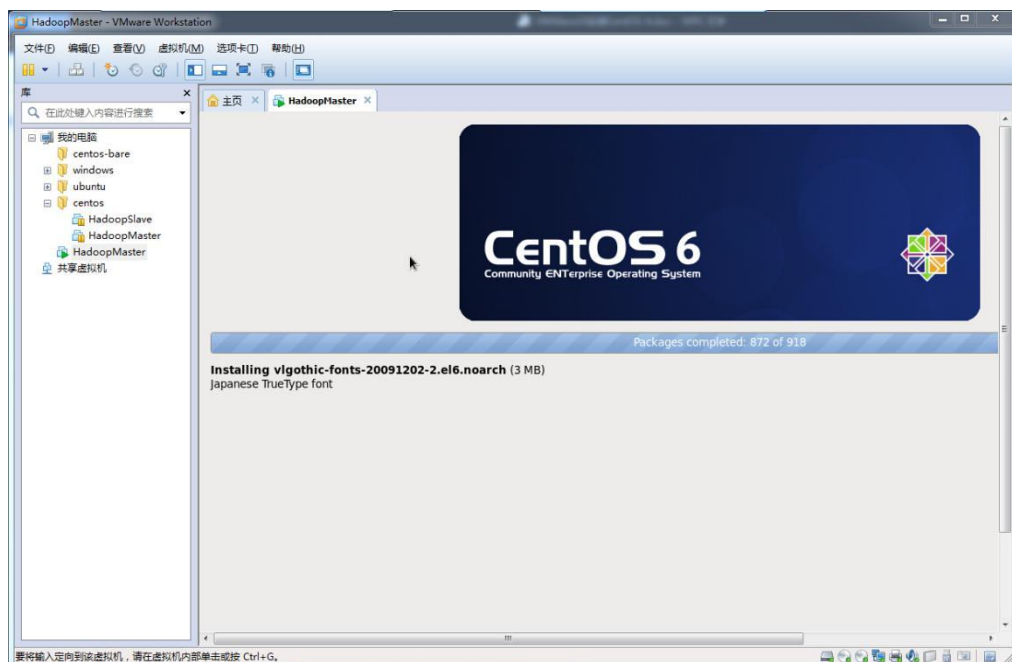


这里的磁盘大小不要直接使用默认值，要调大该值，设置为 30.0

使用默认，点击“下一步 (N) >”

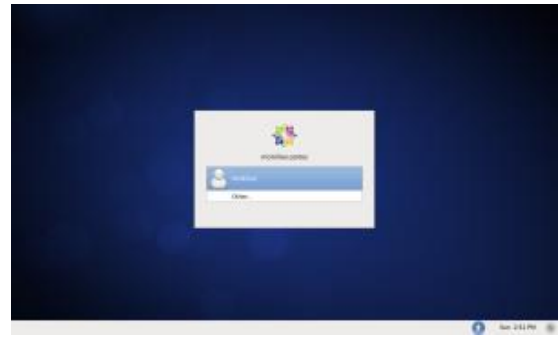


正常情况下，安装 CentOS6 进入下面的界面：





直接等待安装完成，系统自动重启



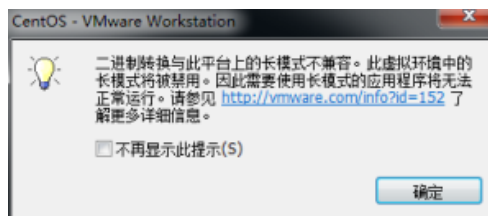
输入密码 zkpk 登录进系统



至此，CentOS 系统安装完毕。

2.2 安装中的关键问题

如果出现下面的界面，说明 BIOS 中没有打开 VT-x 功能，所以就不能用 VT-x 进行加速。

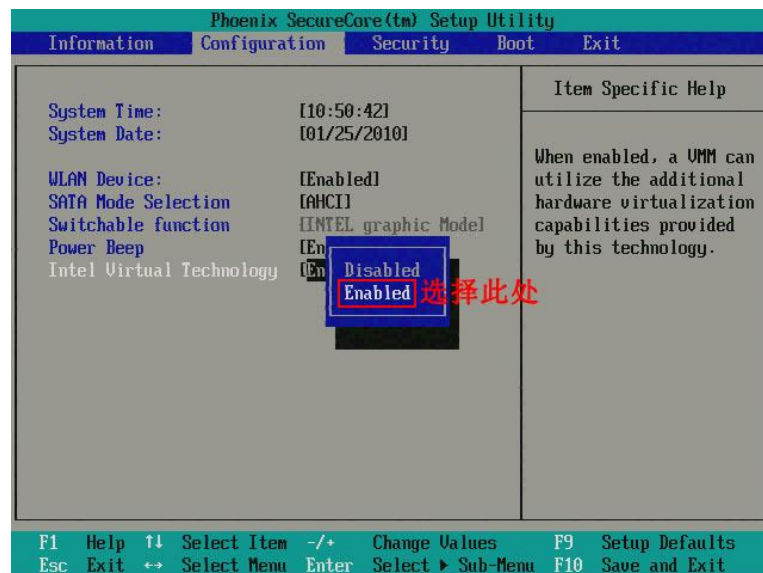


打开 BIOS 中的 VT-x 功能的操作如下：

首先在开机自检 Logo 处按 F2 热键（不同品牌的电脑进入 BIOS 的热键不同，有的电脑是 F1\F8\F12）进入 BIOS，选择 Configuration 选项，选择 Intel Virtual Technology 并回车，如下图：



将光标移动至 Enabled 处，并回车确定，如下图：



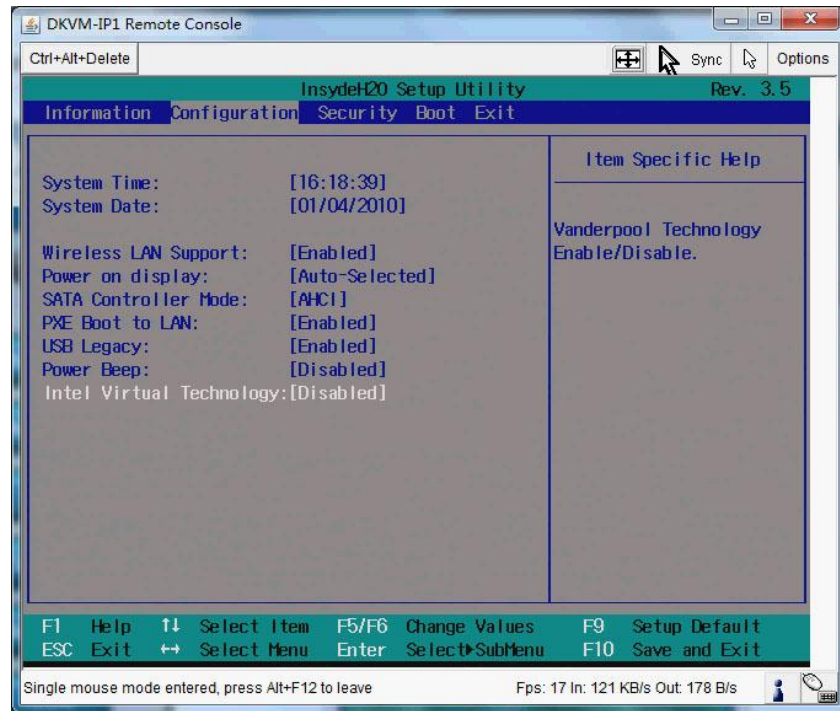
此时该选项将变为 Enabled，最后按 F10 热键保存并退出即可开启 VT 功能，如下图：



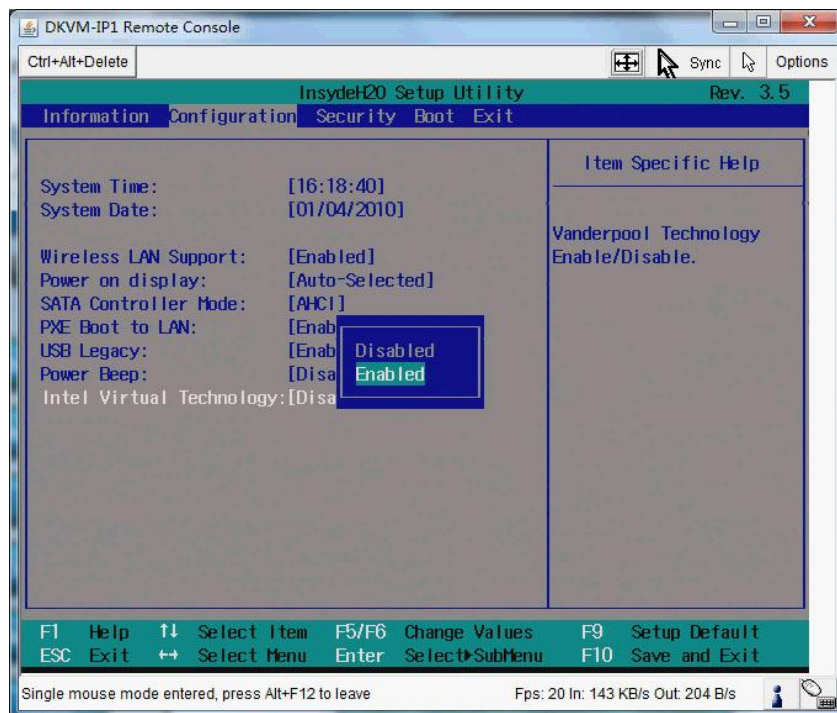


2、Insyde BIOS 机型的参考操作方法：（以 Lenovo 3000 G460 作为操作平台）

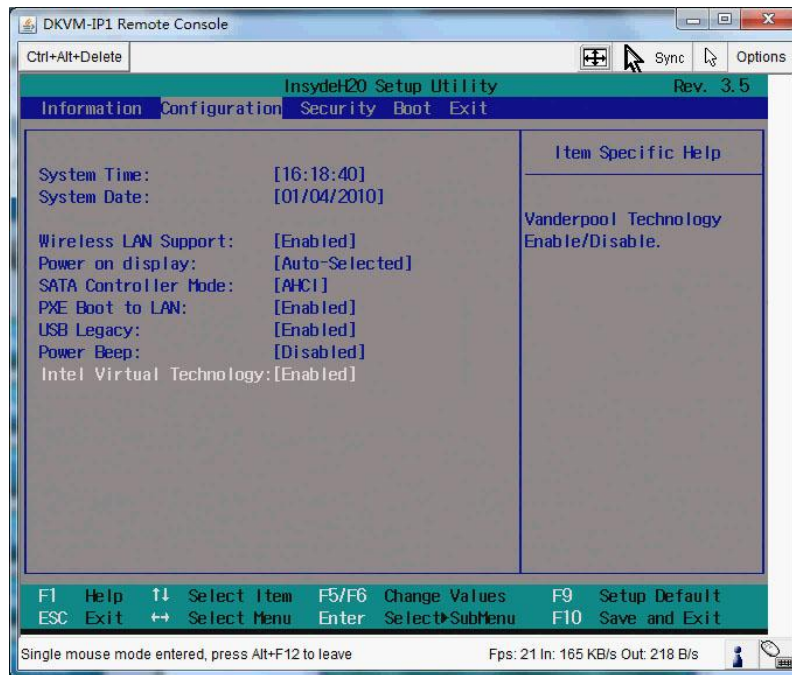
首先在开机自检 Logo 处按 F2 热键进入 BIOS，选择 Configuration 选项，选择 Intel Virtual Technology 并回车，如下图：



将光标移动至 Enabled 处，并回车确定，如下图：



此时该选项将变为 Enabled，最后按 F10 热键保存并退出即可开启 VT 功能，如下图：

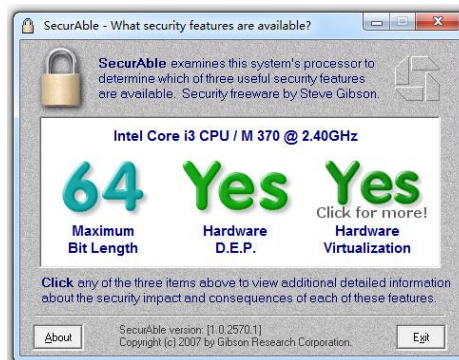


如果修改该 BIOS 选项之后，仍然出现提示 VT-x 没有打开的情况，需要重启电脑重试。如果仍然不可以，请使用下面的方式验证电脑的硬件配置。

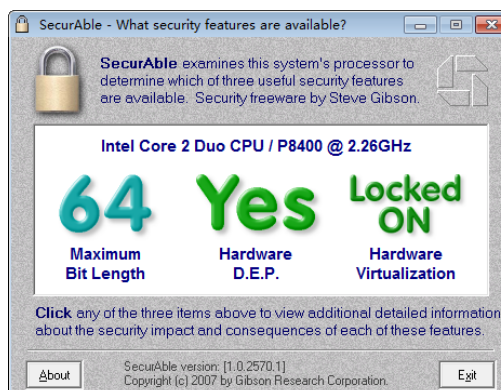
如何测试自己电脑是否支持虚拟化？

运行 SecurAble 软件，有三种情况。

1、如下图，说明支持 64 位系统，满足需求。

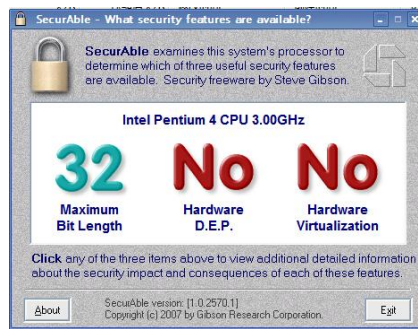


2、如下图，说明支持 64 位系统，但是虚拟化在 BIOS 中没有开启。需要在 BIOS 中开启相关选项。具体不同笔记本型号修改方法，请查询百度。





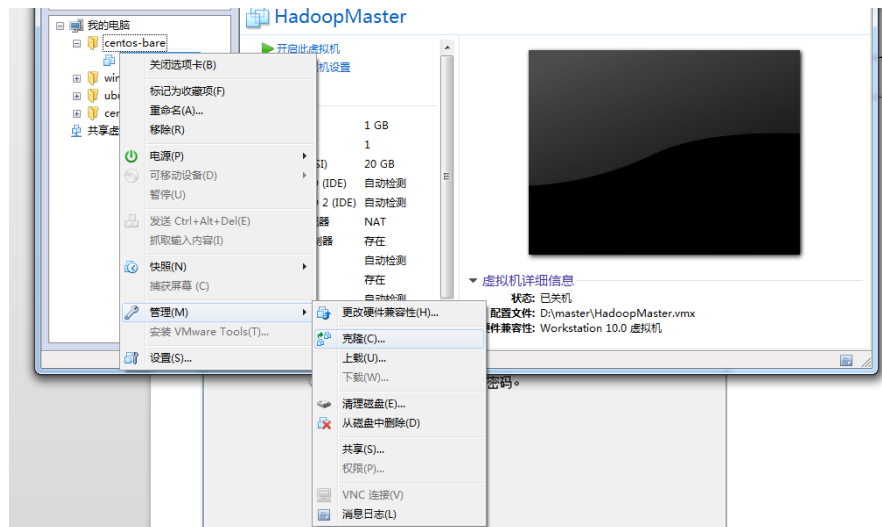
3、如果出现如下图的显示，请您更换笔记本。



SecurAble 就是一款测试测试电脑能否支持 Windows7 的 XP 兼容模式的免费软件，另外 SecurAble 还可以测试你的机器硬件是否支持 Hyper-V 和 KVM，要运行 Hyper-v 和 KVM，物理主机厂的 CPU 必须支持虚拟化，而且主机要 64 位的，同时 BIOS 要开启硬件级别的数据执行保护(Hardware D.E.P)，这些信息通过 SecurAble 就可以找到答案。

2.3 克隆 HadoopSlave

点击下图所示的“克隆”选项



点击“下一步”看到下面的界面

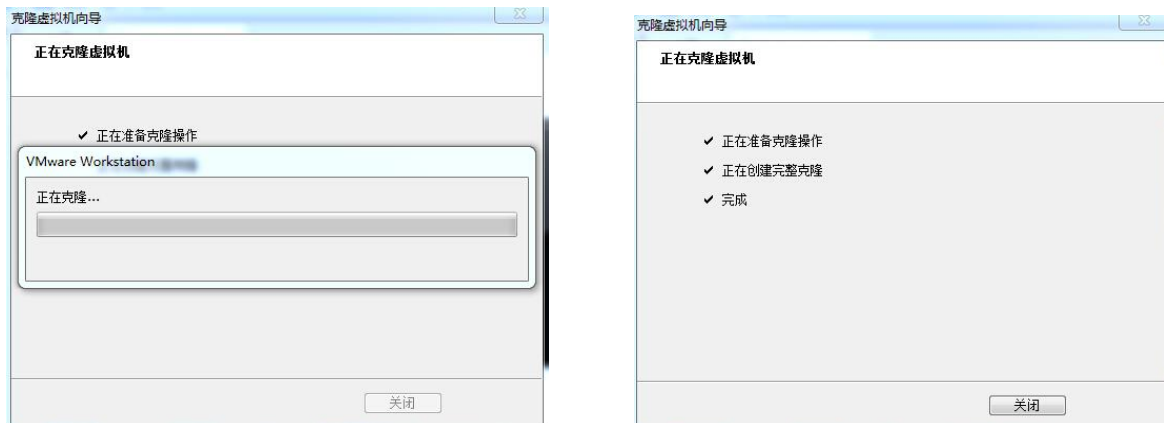




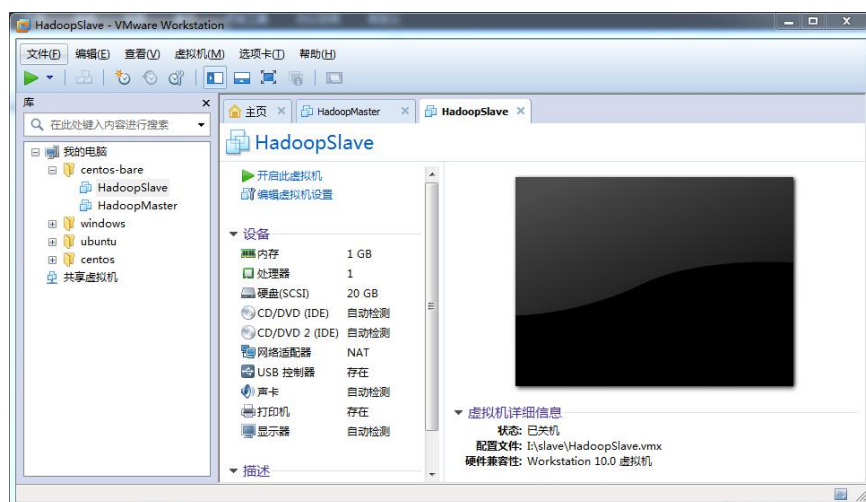
使用默认选项，点击“下一步”，选择“创建完整克隆（F）”，点击“下一步”，如下图所示。



将虚拟机重命名为 HadoopSlave，选择一个存储位置（占用空间 10GB 左右），点击完成



点击“关闭”按钮后，发现“HadoopSlave”虚拟机已经在左侧的列表栏中





2.4 windows 中安装 SSH Secure Shell Client 传输软件

在 Hadoop In Action Experiment 软件包下面的 software 目录中, 包含一个 SSH Secure Shell Client 3.2.9.RAR 的安装文件, 该文件用于 windows 系统与 Linux 系统之间进行文件传输。

1. 安装 SSH Secure Shell Client

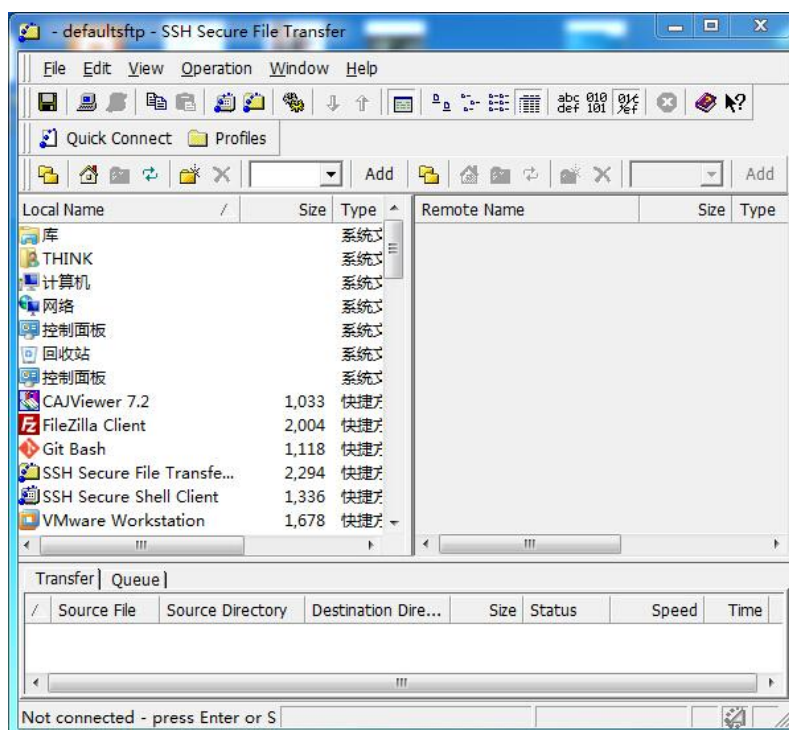
在本机 windows 操作系统的任意位置, 解压并点击安装 SSH Secure Shell Client 软件。

一路点击“NEXT”安装完成, 在 windows 桌面上会看到如下图的快捷方式。

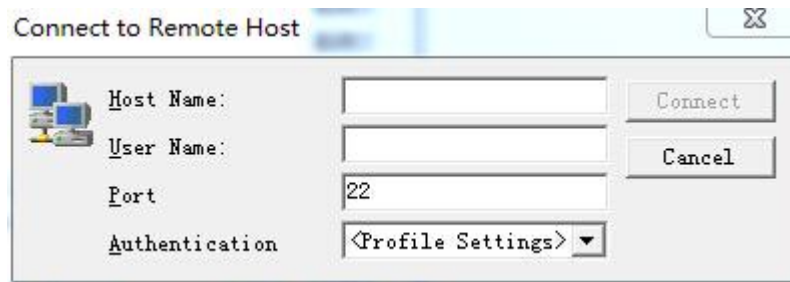


2. 打开并传输文件测试

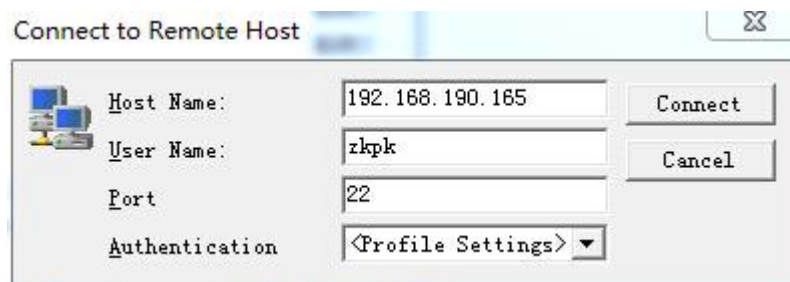
点击黄色文件夹快捷方式“SSH Secure File Transfer Client”, 会出现如下图的弹窗:



点击“Quick Connect”, 弹出连接对话框

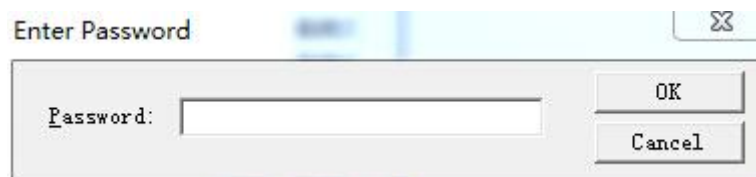


输入已经安装的 CentOS 的主机名和用户名，如下图所示：

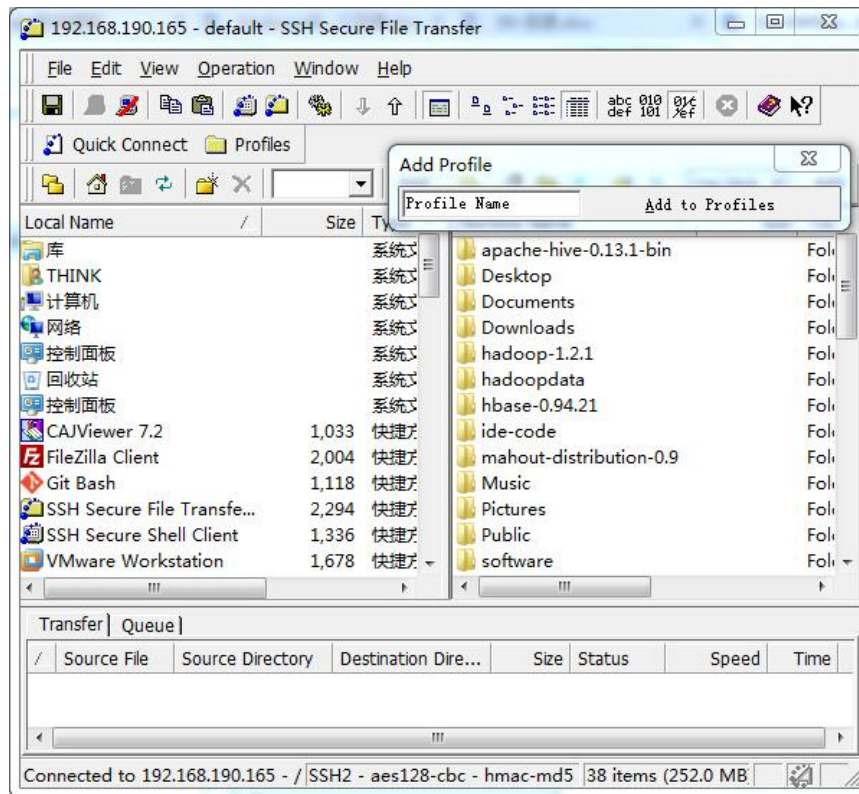


假设安装的 CentOS 虚拟机的 IP 地址是 192.168.190.165，用户名是 zkpk

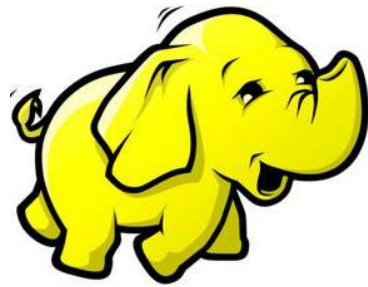
点击“Connect”，弹出输入密码的对话框



输入密码 zkpk，点击“OK”，会看到下面的对话框，表示连接成功。



上图中左侧是 windows 本机目录，右侧是 Linux 目录，拖拽文件即可实现复制。



3 章

CentOS 6 安装 Hadoop

主要内容

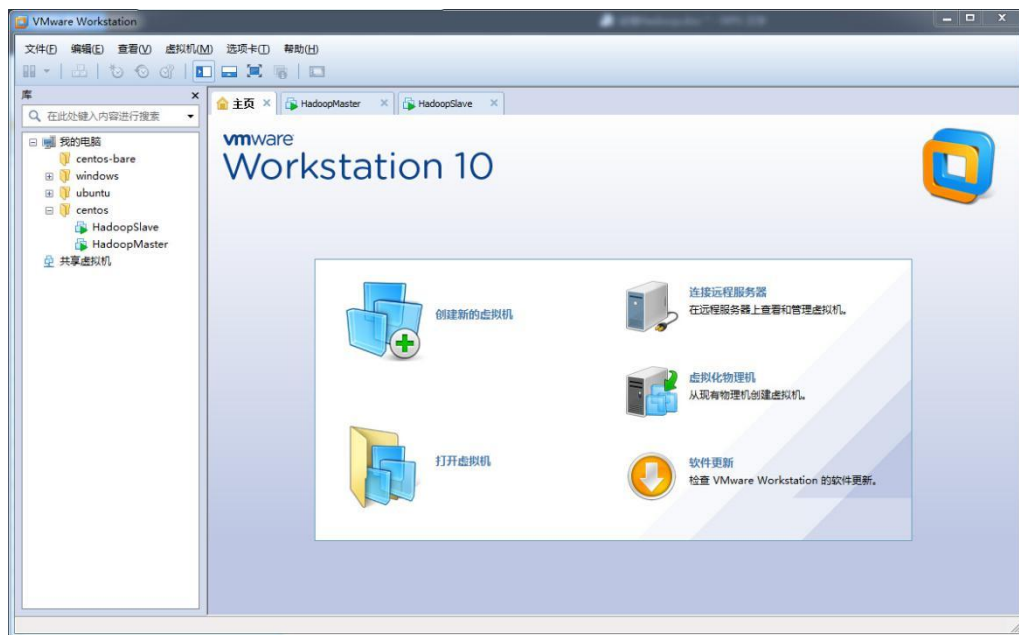
- 启动两台虚拟客户机
- Linux 系统配置
- Hadoop 配置部署
- 启动 Hadoop 集群



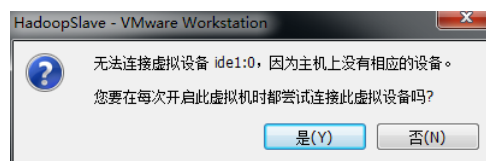
第3章 CentOS 6 安装 Hadoop

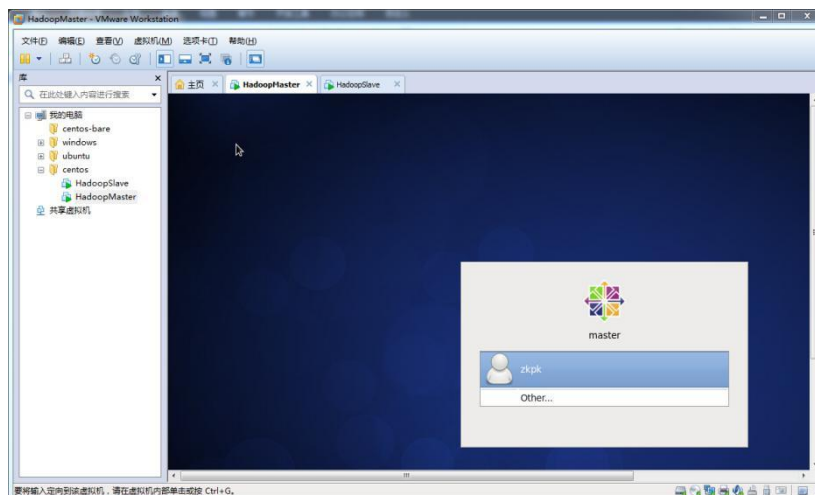
3.1 启动两台虚拟客户机

打开 VMware Workstation10

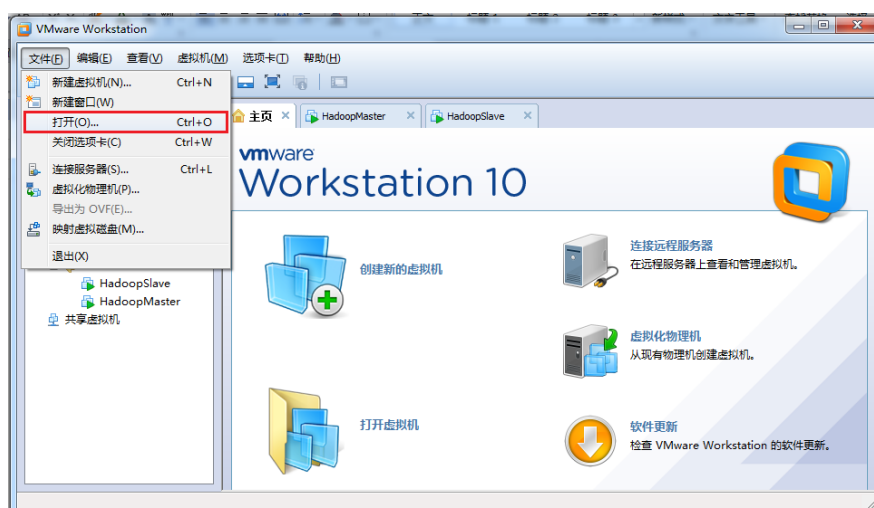


打开之前已经安装好的虚拟机：HadoopMaster 和 HadoopSlave，出现异常，选择“否”进入





如果之前没有打开过两个虚拟机，请使用“文件”→“打开”选项，选择之前的虚拟机安装包（在一体软件包里面的）



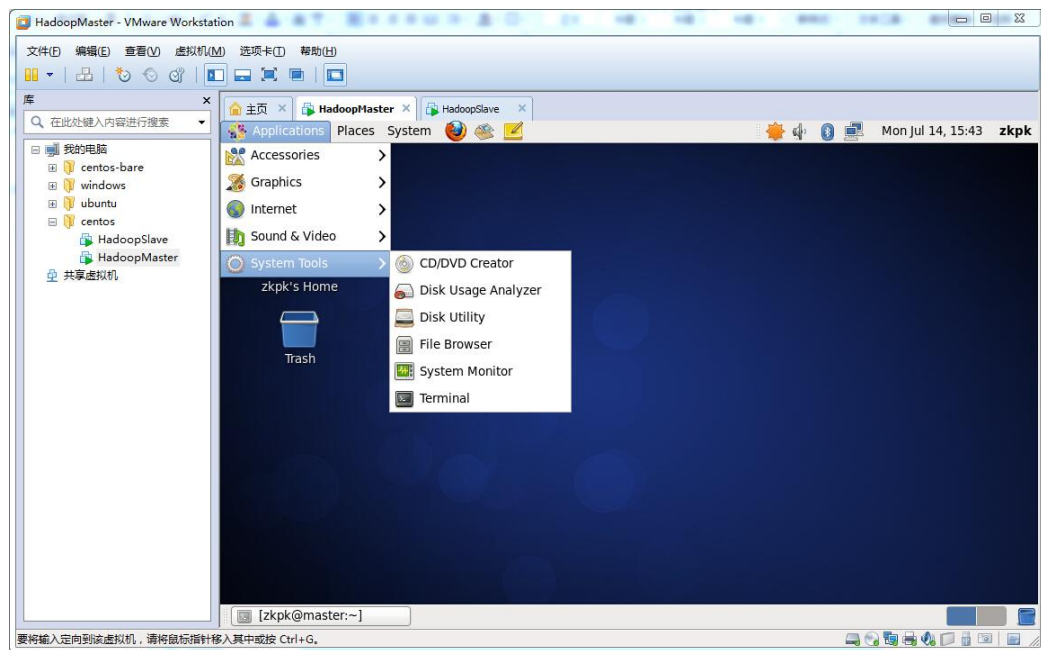
3.2 Linux 系统配置

以下操作步骤需要在 HadoopMaster 和 HadoopSlave 节点上分别完整操作，都使用 root 用户，从当前用户切换 root 用户的命令如下：

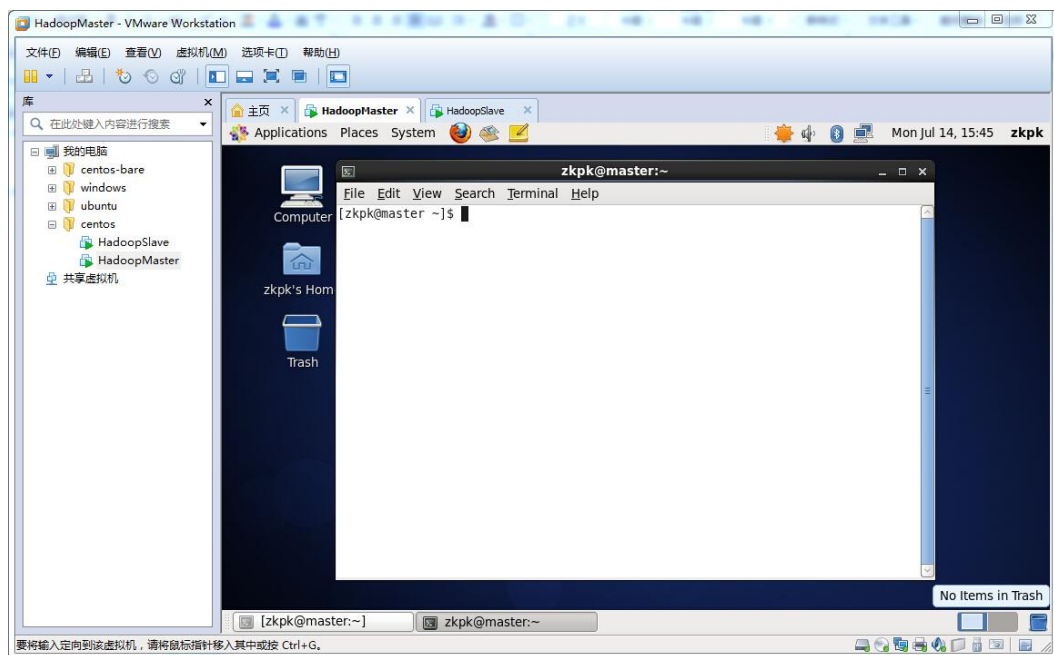
```
[zkpk@master ~]$ su root
```

输入密码：zkpk

本节所有的命令操作都在终端环境，打开终端的过程如下图的 Terminal 菜单：



终端打开后如下图中命令行窗口所示。



3.2.1 软件包和数据包说明

将完整软件包“/home/zpkp/resources”下的 software 是相关的安装软件包，sogou-data 是数据包。



3.2.2 配置时钟同步

1、配置自动时钟同步

该项同时需要在 **HadoopSlave** 节点配置。

使用 Linux 命令配置

```
[root@master zkpk]$ crontab -e
```

该命令是 vi 编辑命令，按 i 进入插入模式，按 Esc，然后键入:wq 保存退出

键入下面的一行代码，输入 i，进入插入模式（星号之间和前后都有空格）

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

2、手动同步时间

直接在 Terminal 运行下面的命令：

```
[root@master zkpk]$ /usr/sbin/ntpdate cn.pool.ntp.org
```

3.2.3 配置主机名

1、HadoopMaster 节点

使用 gedit 编辑主机名，如果不可以使用 gedit，请直接使用 vi 编辑器（后面用到 gedit 的地方也同此处处理一致）。

```
[root@master zkpk]$ gedit /etc/sysconfig/network
```

配置信息如下，如果已经存在则不修改，将 HadoopMaster 节点的主机名改为 master，即下面代码的第 2 行所示。

```
NETWORKING=yes #启动网络  
HOSTNAME=master #主机名
```

确实修改生效命令：

```
[root@master zkpk]$ hostname master
```



检测主机名是否修改成功命令如下，在操作之前需要关闭当前终端，重新打开一个终端：

```
[root@master zkpk]$ hostname
```

执行完命令，会看到下图的打印输输出：

```
zkpk@master:~  
File Edit View Search Terminal Help  
[zkpk@master ~]$ hostname  
master  
[zkpk@master ~]$
```

2、HadoopSlave 节点

使用 gedit 编辑主机名：

```
[root@slave zkpk]$ gedit /etc/sysconfig/network
```

配置信息如下，如果已经存在则不修改，将 Hadoops slave 节点的主机名改为 slave，即下面代码的第 2 行所示。

```
NETWORKING=yes #启动网络  
HOSTNAME=slave #主机名
```

确实修改生效命令：

```
[root@slave zkpk]$ hostname slave
```

检测主机名是否修改成功命令如下，在操作之前需要关闭当前终端，重新打开一个终端：

```
[root@slave zkpk]$ hostname
```

执行完命令，会看到下图的打印输输出

```
zkpk@slave:~  
File Edit View Search Terminal Help  
[zkpk@slave ~]$ hostname  
slave  
[zkpk@slave ~]$
```

3.2.5 使用 setup 命令配置网络环境

该项也需要在 HadoopSlave 节点配置。

在终端中执行下面的命令：

```
[zkpk@master ~]$ ifconfig
```




如果看到下面的打印输出

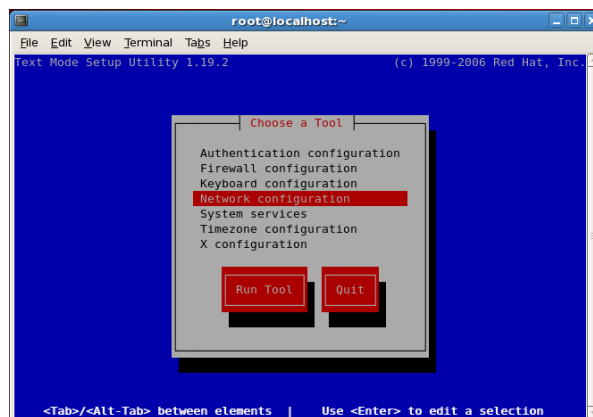
```
zkpk@master:~  
File Edit View Search Terminal Help  
[zkpk@master ~]$ ifconfig  
eth1  Link encap:Ethernet  HWaddr 00:0C:29:D0:74:01  
      inet addr:192.168.190.147  Bcast:192.168.190.255  Mask:255.255.255.0  
      inet6 addr: fe80::20c:29ff:fed0:7401/64 Scope:Link  
      UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
      RX packets:1115 errors:0 dropped:0 overruns:0 frame:0  
      TX packets:125 errors:0 dropped:0 overruns:0 carrier:0  
      collisions:0 txqueuelen:1000  
      RX bytes:143972 (140.5 KiB)  TX bytes:11234 (10.9 KiB)  
  
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:16436  Metric:1  
      RX packets:8 errors:0 dropped:0 overruns:0 frame:0  
      TX packets:8 errors:0 dropped:0 overruns:0 carrier:0  
      collisions:0 txqueuelen:0  
      RX bytes:480 (480.0 b)  TX bytes:480 (480.0 b)  
  
[zkpk@master ~]$
```

如果看到出现红线标注部分出现，即存在内网 IP、广播地址、子网掩码，说明该节点不需要配置网络，否则进行下面的步骤。

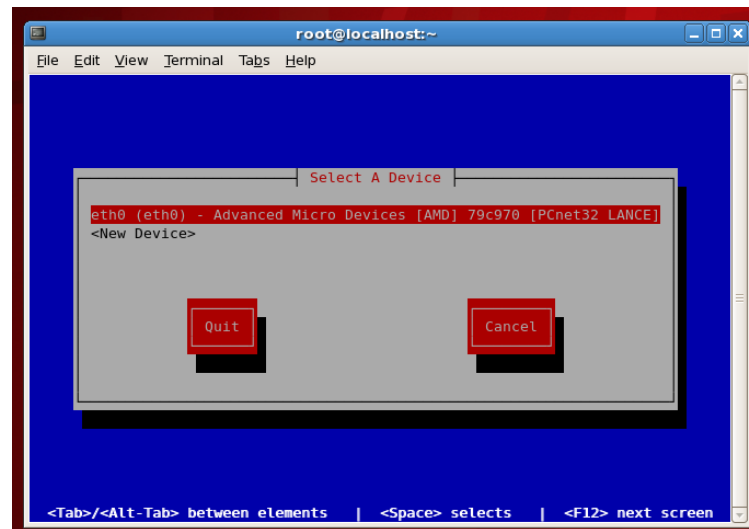
在终端中执行下面命令：

```
[zkpk@master ~]$ setup
```

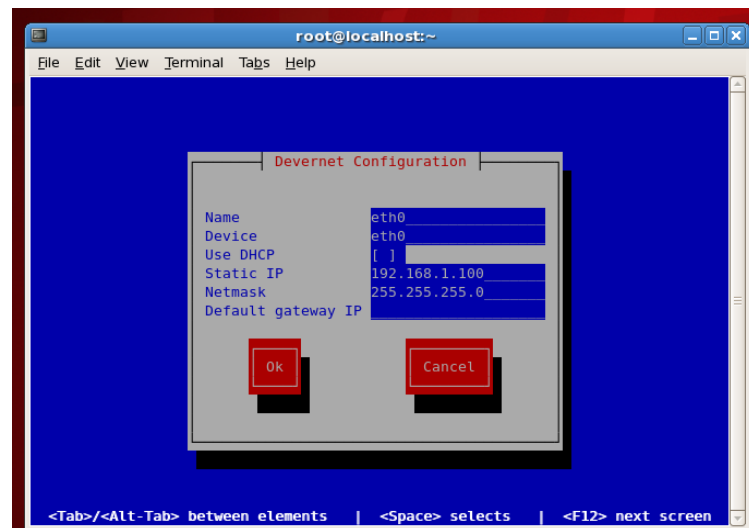
会出现下图中的内容：



使用光标键移动选择“Network configuration”，回车进入该项



使用光标键移动选择 eth0，回车进入该项



按照图中的方式输入各项内容

重启网络服务

```
[root@master zkpk]$ /sbin/service network restart
```

检查是否修改成功:

```
[zkpk@master ~]$ ifconfig
```

看到如下图的内容（IP 不一定和下图相同，根据你之前的配置），说明配置成功，特别关注红线部分



```
zkpk@master:~  
File Edit View Search Terminal Help  
[zkpk@master ~]$ ifconfig  
eth1      Link encap:Ethernet  HWaddr 00:0C:29:D0:74:01  
          inet addr:192.168.190.147  Bcast:192.168.190.255  Mask:255.255.255.0  
          inet6 addr: fe80::20c:29ff:fed0:7401/64 Scope:Link  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:1115 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:125 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:143972 (140.5 KiB)  TX bytes:11234 (10.9 KiB)  
  
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:16436  Metric:1  
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:0  
          RX bytes:480 (480.0 b)  TX bytes:480 (480.0 b)  
  
[zkpk@master ~]$
```

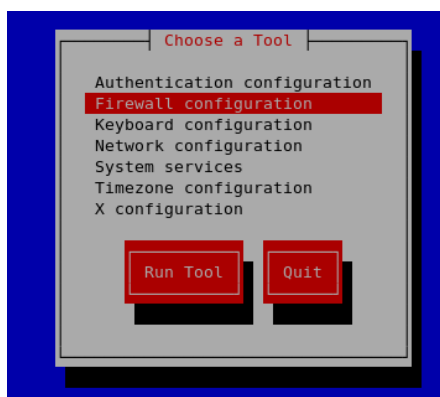
3.2.6 关闭防火墙

该项也需要在 **HadoopSlave** 节点配置。

在终端中执行下面命令：

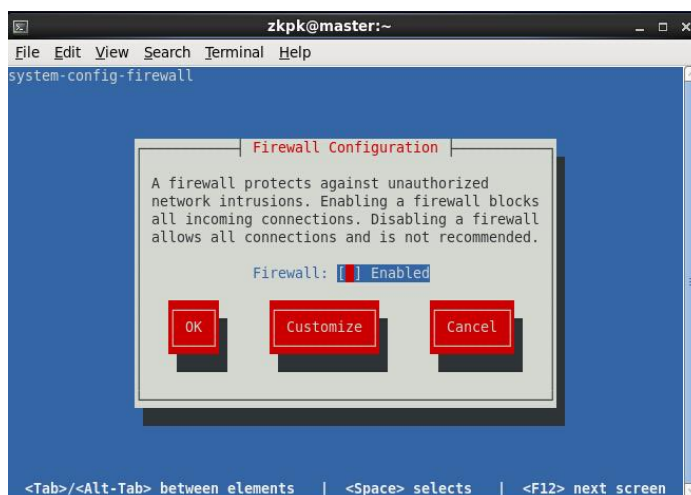
```
[zkpk@master ~]$ setup
```

会出现下图中的内容：

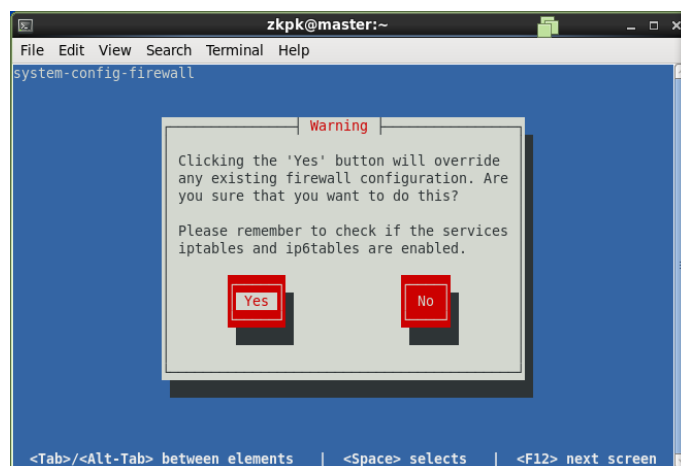


光标移动选择“Firewall configuration”选项，回车进入选项

如果该项前面有“*”标，则按一下空格键关闭防火墙，如下图所示，然后光标移动选择“OK”保存修改内容



选择 OK



3.2.7 配置 hosts 列表

该项也需要在 HadoopSlave 节点配置。

需要在 root 用户下（使用 su 命令），编辑主机名列表的命令：

```
[root@master zpkp]$ gedit /etc/hosts
```

将下面两行添加到/etc/hosts 文件中：

```
192.168.1.100 master
192.168.1.101 slave
```

注意：这里 master 节点对应 IP 地址是 192.168.1.100, slave 对应的 IP 是 192.168.1.101, 而自己在做配置时，需要将这两个 IP 地址改为你的 master 和 slave 对应的 IP 地址。

查看 master 的 IP 地址使用下面的命令：

```
[zpkp@master ~]$ ifconfig
```



master 节点的 IP 是下图中红线标注的内容。

```
zkpk@master:~
File Edit View Search Terminal Help
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.190.147 master
192.168.190.144 slave
[zkpk@master ~]$ ifconfig
eth1      Link encap:Ethernet  HWaddr 00:0c:29:00:74:01
          inet addr:192.168.190.147  Bcast:192.168.190.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fed0:7401/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2018 errors:0 dropped:0 overruns:0 frame:0
          TX packets:189 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:237402 (231.8 KiB)  TX bytes:16314 (15.9 KiB)

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:16436  Metric:1
          RX packets:22 errors:0 dropped:0 overruns:0 frame:0
          TX packets:22 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1656 (1.6 KiB)  TX bytes:1656 (1.6 KiB)

[zkpk@master ~]$
```

slave 的 IP 地址也是这样查看。

验证是否配置成功的命令是：

```
[zkpk@master ~]$ ping master
[zkpk@master ~]$ ping slave
```

如果出现下图的信息表示配置成功：

```
zkpk@master:/home/zkpk
File Edit View Search Terminal Help
[root@master zkpk]# ping master
PING master (192.168.190.147) 56(84) bytes of data.
64 bytes from master (192.168.190.147): icmp_seq=1 ttl=64 time=0.037 ms
64 bytes from master (192.168.190.147): icmp_seq=2 ttl=64 time=0.050 ms
64 bytes from master (192.168.190.147): icmp_seq=3 ttl=64 time=0.032 ms
^C
--- master ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2698ms
rtt min/avg/max/mdev = 0.032/0.039/0.050/0.010 ms
```

如果出现下图的内容，表示配置失败：

```
[root@master zkpk]# ping slave
PING slave (192.168.190.144) 56(84) bytes of data.
From master (192.168.190.147) icmp_seq=1 Destination Host Unreachable
From master (192.168.190.147) icmp_seq=2 Destination Host Unreachable
From master (192.168.190.147) icmp_seq=3 Destination Host Unreachable
From master (192.168.190.147) icmp_seq=4 Destination Host Unreachable
From master (192.168.190.147) icmp_seq=5 Destination Host Unreachable
From master (192.168.190.147) icmp_seq=6 Destination Host Unreachable
^C
--- slave ping statistics ---
7 packets transmitted, 0 received, 100% packet loss, time 6188ms
pipe 4
```

3.2.8 安装 JDK

该项也需要在 HadoopSlave 节点配置。

将 JDK 文件解压，放到/usr/java 目录下



```
[zkpk@master ~]$ cd /home/zkpk/resources/software/jdk
[zkpk@master jdk]$ mkdir /usr/java
[zkpk@master jdk]$ mv ~/resources/software/jdk/jdk-7u71-linux-x64.gz /usr/java/
[zkpk@master jdk]$ cd /usr/java
[zkpk@master java]$ tar -xvf /usr/java/jdk-7u71-linux-x64.gz
```

使用 **gedit** 配置环境变量

```
[zkpk@master java]$ gedit /home/zkpk/.bash_profile
```

复制粘贴以下内容添加到到上面 **gedit** 打开的文件中:

```
export JAVA_HOME=/usr/java/jdk1.7.0_71/
export PATH=$JAVA_HOME/bin:$PATH
```

使改动生效命令:

```
[zkpk@master java]$ source /home/zkpk/.bash_profile
```

测试配置:

```
[zkpk@master ~]$ java -version
```

如果出现下图的信息, 表示 JDK 安装成功:

```
zkpk@master:~
File Edit View Search Terminal Help
[zkpk@master ~]$ java -version
java version "1.7.0_71"
Java(TM) SE Runtime Environment (build 1.7.0_71-b14)
Java HotSpot(TM) 64-Bit Server VM (build 24.71-b01, mixed mode)
[zkpk@master ~]$
```

3.2.9 免密钥登录配置

该部分所有的操作都要在 **zkpk** 用户下, 切换回 **zkpk** 的命令是:

```
su -zkpk
```

密码是: zkpk

1、HadoopMaster 节点

在终端生成密钥, 命令如下 (一路点击回车生成密钥)

```
[zkpk@master ~]$ ssh-keygen -t rsa
```

生成的密钥在 **.ssh** 目录下如下图所示:



```
zkpk@master: ~/.ssh
File Edit View Search Terminal Help
[zkpk@master ~]$ cd .ssh
[zkpk@master .ssh]$ ls -l
total 8
-rw----- 1 zkpk zkpk 1675 Jul 14 18:19 id_rsa
-rw-r--r-- 1 zkpk zkpk 393 Jul 14 18:19 id_rsa.pub
[zkpk@master .ssh]$
```

复制公钥文件

```
[zkpk@master .ssh]$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

执行 `ls -l` 命令后会看到下图的文件列表：

```
[zkpk@master .ssh]$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
[zkpk@master .ssh]$ ls -l
total 12
-rw-rw-r-- 1 zkpk zkpk 393 Jul 14 18:23 authorized_keys
-rw----- 1 zkpk zkpk 1675 Jul 14 18:19 id_rsa
-rw-r--r-- 1 zkpk zkpk 393 Jul 14 18:19 id_rsa.pub
```

修改 `authorized_keys` 文件的权限，命令如下：

```
[zkpk@master .ssh]$ chmod 600 ~/.ssh/authorized_keys
```

修改完权限后，文件列表情况如下：

```
[zkpk@master .ssh]$ chmod 600 authorized_keys
[zkpk@master .ssh]$ ls -l
total 12
-rw----- 1 zkpk zkpk 393 Jul 14 18:23 authorized_keys
-rw----- 1 zkpk zkpk 1675 Jul 14 18:19 id_rsa
-rw-r--r-- 1 zkpk zkpk 393 Jul 14 18:19 id_rsa.pub
[zkpk@master .ssh]$
```

将 `authorized_keys` 文件复制到 `slave` 节点，命令如下：

```
[zkpk@master .ssh]$ scp ~/.ssh/authorized_keys zkpk@slave:~/
```

如果提示输入 `yes/no` 的时候，输入 `yes`，回车

密码是：zkpk

2、HadoopSlave 节点

在终端生成密钥，命令如下（一路点击回车生成密钥）

```
[zkpk@slave ~]$ ssh-keygen -t rsa
```

将 `authorized_keys` 文件移动到 `.ssh` 目录

```
[zkpk@slave ~]$ mv authorized_keys ~/.ssh/
```



3、验证免密钥登陆

在 HadoopMaster 机器上执行下面的命令：

```
[zkpk@master ~]$ ssh slave
```

如果出现下图的内容表示免密钥配置成功：



3.3 Hadoop 配置部署

每个节点上的 Hadoop 配置基本相同，在 HadoopMaster 节点操作，然后完成复制到另一个节点。

下面所有的操作都使用 zkpk 用户，切换 zkpk 用户的命令是：

```
[root@master zkpk]$ su - zkpk
```

密码是：zkpk

将软件包中的 Hadoop 生态系统包复制到相应 zkpk 用户的主目录下(直接拖拽方式即可拷贝)

3.3.1 Hadoop 安装包解压

进入 Hadoop 软件包，命令如下：

```
[zkpk@master ~]$ cd /home/zkpk/resources/software/hadoop/apache
```

复制并解压 Hadoop 安装包命令如下：

```
[zkpk@master apache]$ cp ~/resources/software/hadoop/apache/hadoop-2.5.2.tar.gz ~/
[zkpk@master apache]$ cd
[zkpk@master ~]$ tar -xvf ~/hadoop-2.5.2.tar.gz
[zkpk@master ~]$ cd ~/hadoop-2.5.2
```

ls -l 看到如下图的内容，表示解压成功：



```
zkpk@master:~/hadoop-2.5.1
File Edit View Search Terminal Help
[zkpk@master hadoop-2.5.1]$ ls -l
total 56
drwxr-xr-x. 2 zkpk zkpk 4096 Nov 16 19:01 bin
drwxr-xr-x. 3 zkpk zkpk 4096 Sep 5 16:30 etc
drwxr-xr-x. 2 zkpk zkpk 4096 Sep 5 16:30 include
drwxr-xr-x. 3 zkpk zkpk 4096 Sep 5 16:30 lib
drwxr-xr-x. 2 zkpk zkpk 4096 Nov 16 19:06 libexec
-rw-r--r--. 1 zkpk zkpk 15458 Sep 5 16:30 LICENSE.txt
drwxrwxr-x. 2 zkpk zkpk 4096 Nov 16 19:20 logs
-rw-r--r--. 1 zkpk zkpk 101 Sep 5 16:30 NOTICE.txt
-rw-r--r--. 1 zkpk zkpk 1366 Sep 5 16:30 README.txt
drwxr-xr-x. 2 zkpk zkpk 4096 Nov 14 07:52 sbin
drwxr-xr-x. 4 zkpk zkpk 4096 Sep 5 16:30 share
```

3.3.2 配置环境变量 hadoop-env.sh

环境变量文件中，只需要配置 JDK 的路径。

```
[zkpk@master hadoop-2.5.2]$ gedit /home/zkpk/hadoop-2.5.2/etc/hadoop/hadoop-env.sh
```

在文件的靠前的部分找到下面的一行代码：

```
export JAVA_HOME=${JAVA_HOME}
```

将这行代码修改为下面的代码：

```
export JAVA_HOME=/usr/java/jdk1.7.0_71/
```

然后保存文件。

3.3.3 配置环境变量 yarn-env.sh

环境变量文件中，只需要配置 JDK 的路径。

```
[zkpk@master hadoop-2.5.2]$ gedit ~/hadoop-2.5.2/etc/hadoop/yarn-env.sh
```

在文件的靠前的部分找到下面的一行代码：

```
# export JAVA_HOME=/home/y/libexec/jdk1.6.0/
```

将这行代码修改为下面的代码（将#号去掉）：

```
export JAVA_HOME=/usr/java/jdk1.7.0_71/
```

然后保存文件。



3.3.4 配置核心组件 core-site.xml

使用 gedit 编辑：

```
[zkpk@master hadoop-2.5.2]$ gedit ~/hadoop-2.5.2/etc/hadoop/core-site.xml
```

用下面的代码替换 core-site.xml 中的内容：

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- 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>/home/zkpk/hadoopdata</value>
  </property>
</configuration>
```

3.3.5 配置文件系统 hdfs-site.xml

使用 gedit 编辑：

```
[zkpk@master hadoop-2.5.2]$ gedit ~/hadoop-2.5.2/etc/hadoop/hdfs-site.xml
```

用下面的代码替换 hdfs-site.xml 中的内容：

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
</configuration>
```



3.3.6 配置文件系统 yarn-site.xml

使用 gedit 编辑:

```
[zkpk@master hadoop-2.5.2]$ gedit ~/hadoop-2.5.2/etc/hadoop/yarn-site.xml
```

用下面的代码替换 yarn-site.xml 中的内容:

```
<?xml version="1.0"?>

<configuration>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
  <property>
    <name>yarn.resourcemanager.address</name>
    <value>master:18040</value>
  </property>
  <property>
    <name>yarn.resourcemanager.scheduler.address</name>
    <value>master:18030</value>
  </property>
  <property>
    <name>yarn.resourcemanager.resource-tracker.address</name>
    <value>master:18025</value>
  </property>
  <property>
    <name>yarn.resourcemanager.admin.address</name>
    <value>master:18141</value>
  </property>
  <property>
    <name>yarn.resourcemanager.webapp.address</name>
    <value>master:18088</value>
  </property>
</configuration>
```

3.3.7 配置计算框架 mapred-site.xml

复制 mapred-site-template.xml 文件:

```
[zkpk@master hadoop-2.5.2]$ cp ~/hadoop-2.5.2/etc/hadoop/mapred-site.xml.template
~/hadoop-2.5.2/etc/hadoop/mapred-site.xml
```



使用 **gedit** 编辑：

```
[zkpk@master ~]$ gedit ~/hadoop-2.5.2/etc/hadoop/mapred-site.xml
```

用下面的代码替换 **mapred-site.xml** 中的内容

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>

</configuration>
```

3.3.8 在 master 节点配置 slaves 文件

使用 **gedit** 编辑：

```
[zkpk@master hadoop-2.5.2]$ gedit ~/hadoop-2.5.2/etc/hadoop/slaves
```

用下面的代码替换 **slaves** 中的内容：

```
slave
```

3.3.9 复制到从节点

使用下面的命令将已经配置完成的 **Hadoop** 复制到从节点 **HadoopSlave** 上：

```
[zkpk@master hadoop-2.5.2]$ cd
[zkpk@master ~]$ scp -r hadoop-2.5.2 zkpk@slave:~/
```

注意：因为之前已经配置了免密钥登录，这里可以直接远程复制。

3.4 启动 Hadoop 集群

下面所有的操作都使用 **zkpk** 用户，切换 **zkpk** 用户的命令是：

```
su - zkpk
```

密码是：zkpk



3.4.1 配置 Hadoop 启动的系统环境变量

该节的配置需要同时在两个节点（HadoopMaster 和 HadoopSlave）上进行操作，操作命令如下：

```
[zkpk@master hadoop-2.5.2]$ cd
[zkpk@master ~]$ gedit ~/.bash_profile
```

将下面的代码追加到.bash_profile 末尾：

```
#HADOOP
export HADOOP_HOME=/home/zkpk/hadoop-2.5.2
export PATH=$HADOOP_HOME/bin:$HADOOP_HOME/sbin:$PATH
```

然后执行命令：

```
[zkpk@master ~]$ source ~/.bash_profile
```

3.4.2 创建数据目录

该节的配置需要同时在两个节点（HadoopMaster 和 HadoopSlave）上进行操作。

在 zkpk 的用户主目录下，创建数据目录，命令如下：

```
[zkpk@master ~]$ mkdir /home/zkpk/hadoopdata
```

3.4.3 启动 Hadoop 集群

1、格式化文件系统

格式化命令如下，该操作需要在 HadoopMaster 节点上执行：

```
[zkpk@master ~]$ hdfs namenode -format
```

看到下图的打印信息表示格式化成功，如果出现 Exception/Error，则表示出问题：



```
zkpk@master:~/hadoop-2.5.1/etc/hadoop
File Edit View Search Terminal Help
[zkpk@master hadoop]$ hdfs namenode -format
14/11/17 03:55:18 INFO namenode.NameNode: STARTUP_MSG:
/*****
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = master/192.168.1.100
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 2.5.1
STARTUP_MSG:   classpath = /home/zkpk/hadoop-2.5.1/etc/hadoop:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/mockito-all-1.8.5.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/protobuf-java-2.5.0.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/log4j-1.2.17.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/jackson-mapper-asl-1.9.13.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/jasper-compiler-5.5.23.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/jersey-json-1.9.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/httpcore-4.2.5.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/xmlenc-0.52.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/commons-configuration-1.6.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/httpclient-4.2.5.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/jsr305-1.3.9.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/paranamer-2.3.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/snappy-java-1.0.4.1.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/jackson-core-asl-1.9.13.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/jasper-runtime-5.5.23.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/asm-3.2.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/jersey-core-1.9.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/jetty-util-6.1.26.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/lib/jettison-1.1.jar:/home/zkpk/hadoop-2.5.1/share/hadoop/common/
```

2、启动 Hadoop

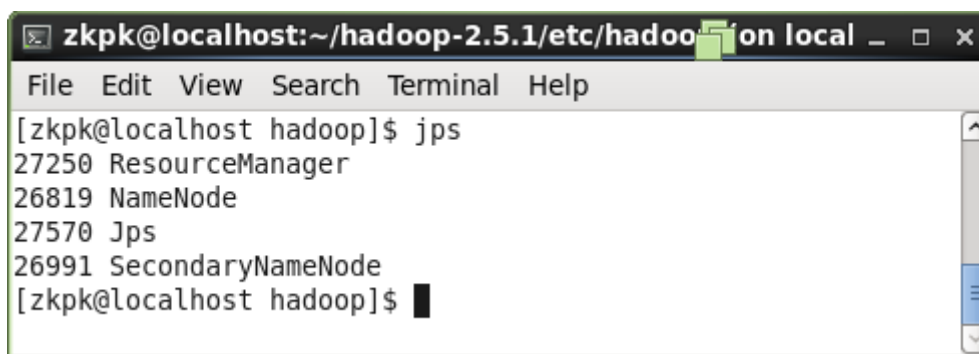
使用 `start-all.sh` 启动 Hadoop 集群，首先进入 Hadoop 安装主目录，然后执行启动命令：

```
[zkpk@master ~]$ cd ~/hadoop-2.5.2
[zkpk@master hadoop-2.5.2]$ sbin/start-all.sh
```

执行命令后，提示出入 yes/no 时，输入 yes。

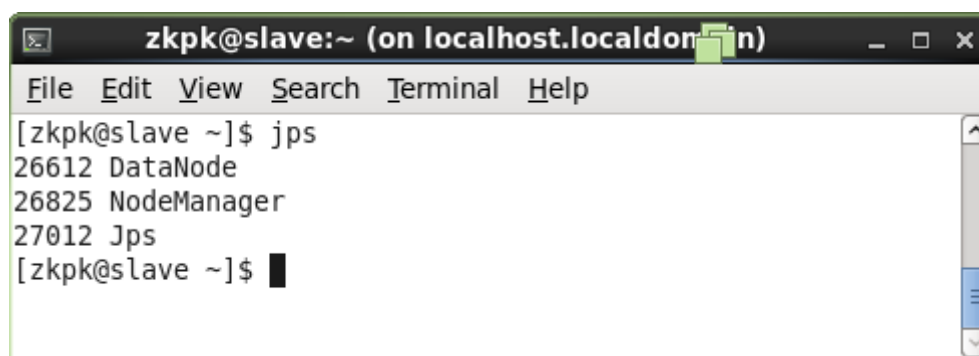
3、查看进程是否启动

在 HadoopMaster 的终端执行 `jps` 命令，在打印结果中会看到 4 个进程，分别是 ResourceManager、Jps、NameNode 和 SecondaryNameNode，如下图所示。如果出现了这 4 个进程表示主节点进程启动成功。



```
zkpk@localhost:~/hadoop-2.5.1/etc/hadoop$ jps
27250 ResourceManager
26819 NameNode
27570 Jps
26991 SecondaryNameNode
zkpk@localhost hadoop$
```

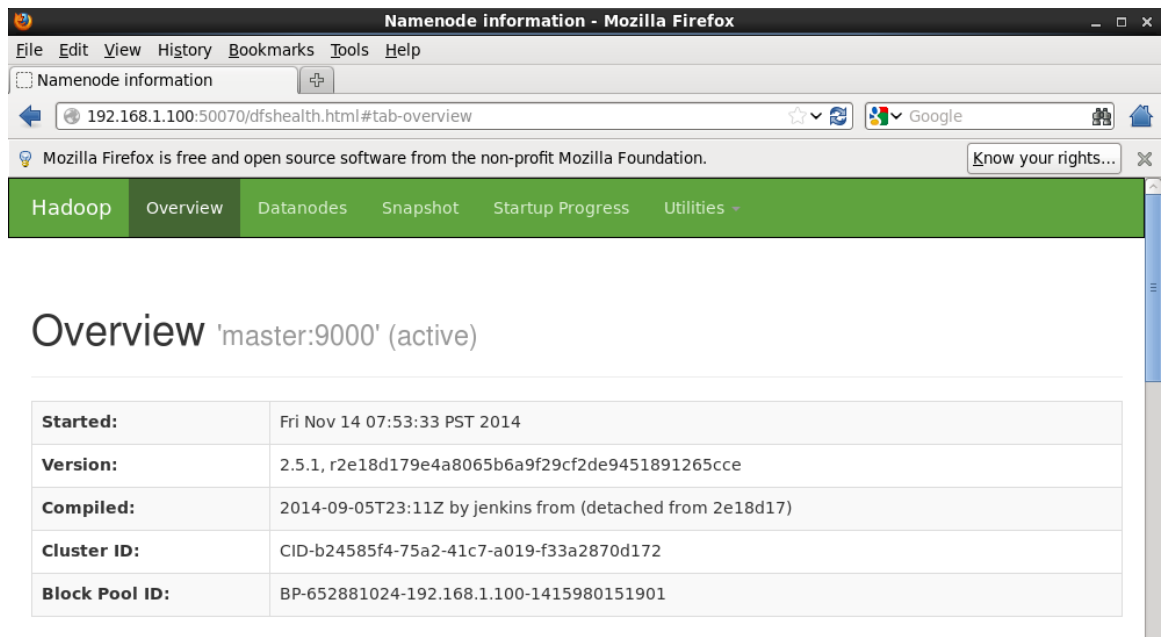
在 HadoopSlave 的终端执行 `jps` 命令，在打印结果中会看到 3 个进程，分别是 NodeManager、DataNode 和 Jps，如下图所示。如果出现了这 3 个进程表示从节点进程启动成功。



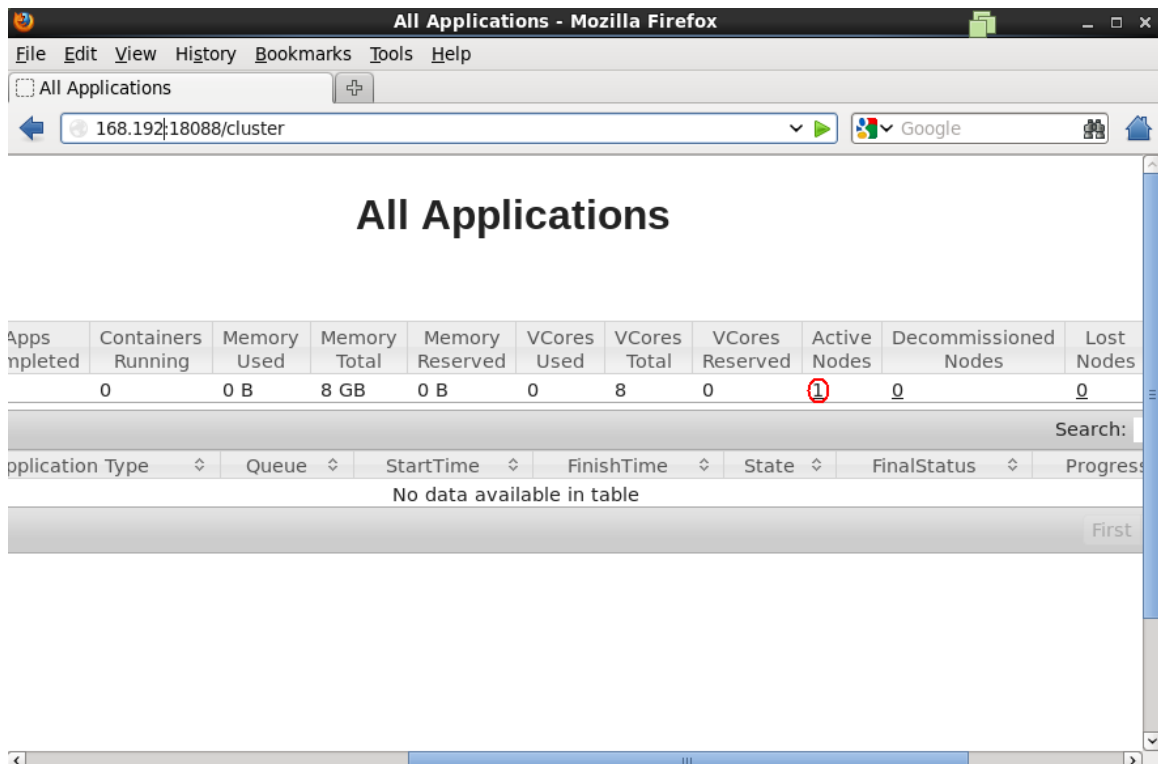
```
zkpk@slave:~ (on localhost.localdomain)$ jps
26612 DataNode
26825 NodeManager
27012 Jps
zkpk@slave ~]$
```

4、Web UI 查看集群是否成功启动

在 HadoopMaster 上启动 Firefox 浏览器，在浏览器地址栏中输入输入 <http://master:50070/>，检查 namenode 和 datanode 是否正常。UI 页面如下图所示。



在 HadoopMaster 上启动 Firefox 浏览器，在浏览器地址栏中输入输入 <http://master:18088/>，检查 Yarn 是否正常，页面如下图所示。



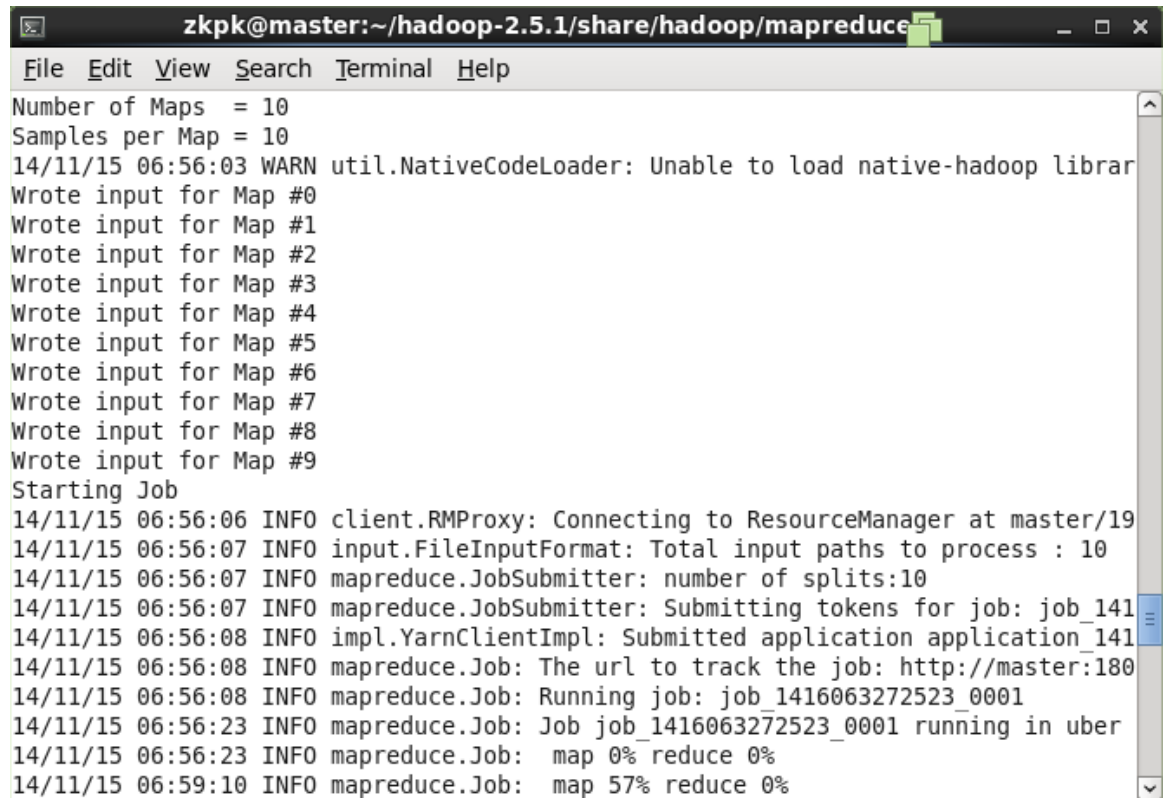
5、运行 PI 实例检查集群是否成功

进入 Hadoop 安装主目录，执行下面的命令：



```
cd
[zkpk@master ~]$ cd ~/hadoop-2.5.2/share/hadoop/mapreduce/
[zkpk@master mapreduce]$ hadoop jar
~/hadoop-2.5.2/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.5.1.jar pi 10 10
```

会看到如下的执行结果：



```
zkpk@master:~/hadoop-2.5.1/share/hadoop/mapreduce
File Edit View Search Terminal Help
Number of Maps = 10
Samples per Map = 10
14/11/15 06:56:03 WARN util.NativeCodeLoader: Unable to load native-hadoop library from classpath due to an error: java.lang.UnsatisfiedLinkError: /lib: libhadoop.so.1: cannot open shared object file: No such file or directory
Wrote input for Map #0
Wrote input for Map #1
Wrote input for Map #2
Wrote input for Map #3
Wrote input for Map #4
Wrote input for Map #5
Wrote input for Map #6
Wrote input for Map #7
Wrote input for Map #8
Wrote input for Map #9
Starting Job
14/11/15 06:56:06 INFO client.RMProxy: Connecting to ResourceManager at master/192.168.1.101:8020
14/11/15 06:56:07 INFO input.FileInputFormat: Total input paths to process : 10
14/11/15 06:56:07 INFO mapreduce.JobSubmitter: number of splits:10
14/11/15 06:56:07 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1416063272523_0001
14/11/15 06:56:08 INFO impl.YarnClientImpl: Submitted application application_1416063272523_0001
14/11/15 06:56:08 INFO mapreduce.Job: The url to track the job: http://master:18080/jobtracking/job_1416063272523_0001/
14/11/15 06:56:08 INFO mapreduce.Job: Running job: job_1416063272523_0001
14/11/15 06:56:23 INFO mapreduce.Job: Job job_1416063272523_0001 running in uber mode
14/11/15 06:56:23 INFO mapreduce.Job: map 0% reduce 0%
14/11/15 06:59:10 INFO mapreduce.Job: map 57% reduce 0%
```

最后输出：

Estimated value of Pi is 3.20000000000000000000

如果以上的 3 个验证步骤都没有问题，说明集群正常启动。