Hadoop 伪分布模式 HDFS+YARN

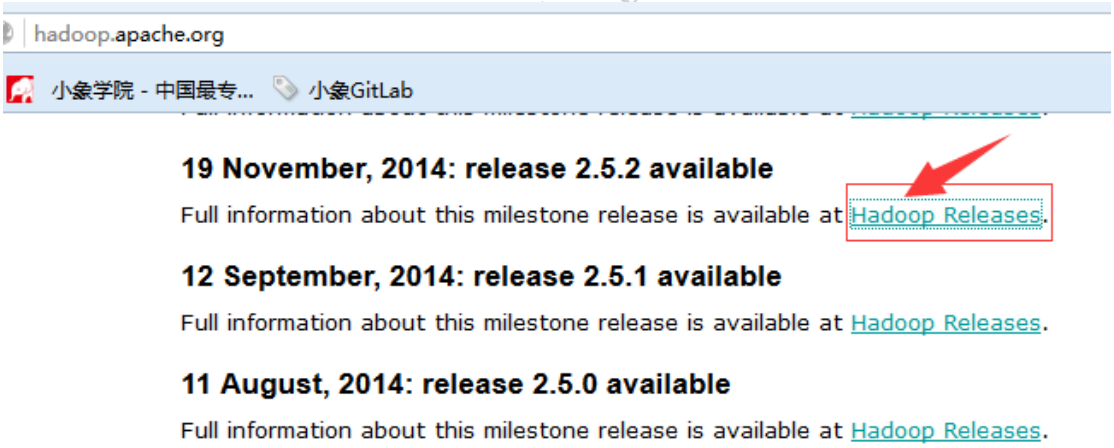
目录

Hadoop2.5.2 伪分布模式 HDFS+YARN....1
1. 下载 Hadoop2.5.2....2
2. 配置 hosts....4
2.1. 查看主机名....4
2.2. 查看 IP 地址....4
2.3. 配置主机名和 IP 的映射关系....5
3. 修改 hadoop 配置文件....5
3.1. 配置文件存放位置....5
3.2. 修改 slaves 文件....5
3.3. 修改 hadoop-env.sh 文件....6
3.4. 修改 mapred-site.xml 文件....6
3.5. 修改 core-site.xml 文件....6
3.6. 修改 hdfs-site.xml 文件....7
3.7. 修改 yarn-site.xml 文件....8
4. 搭建伪分布环境....9
4.1. 格式化 namenode....9
4.2. 启动 namenode....9
4.3. 启动 datanode....9
4.4. 启动 yarn....10
5. 执行一个 MapReduce 任务....11
6. 在 win7 的浏览器中访问 hadoop 集群....13
6.1. 关闭 Centos6.6 虚拟机的防火墙....13
6.2. 在 win7 的浏览器中访问....14
7. 停止集群....15
8. 配置免密码登录....15
8.1. 生成密钥....15
8.2. 拷贝密钥....16
8.3. 再次启动集群....17

1. 下载 Hadoop2.5.2

1.1. 在 Apache Hadoop 官网 http://hadoop.apache.org/上下载。
点击 Download Hadoop 连接

1.2. 选择 Releases 2.5.2 的 Hadoop 。



1.3. 点击 binary 连接。

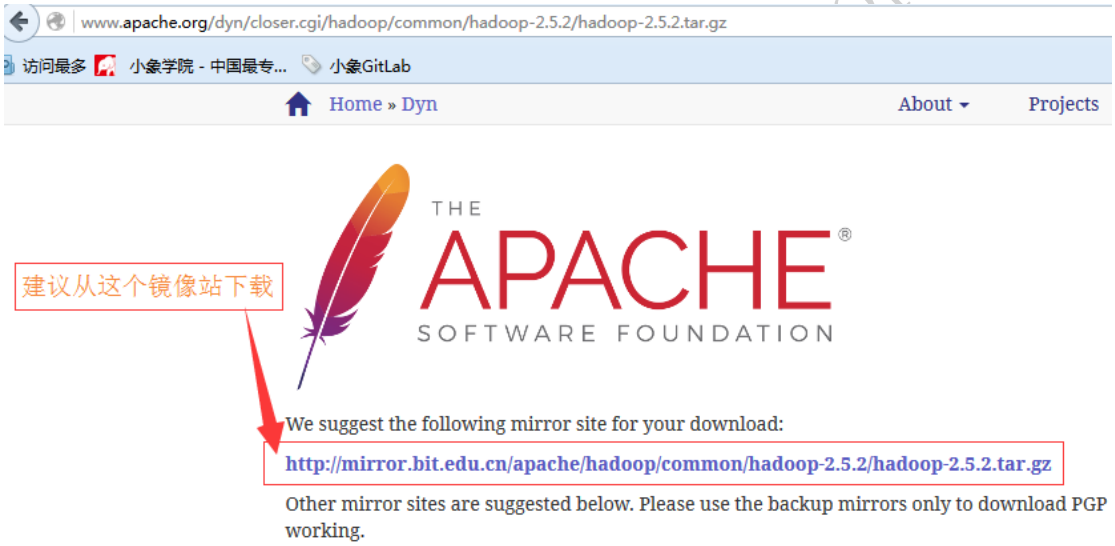
Apache Hadoop Releases

Download

Hadoop is released as source code tarballs with corresponding binary tarballs for convenience.

Version	Release Date	Tarball
2.6.3	17 Dec, 2015	source
		binary
2.6.2	28 Oct, 2015	source
		binary
2.7.1	06 July, 2015	source
		binary
2.6.0	18 Nov, 2014	source
		binary
2.5.2	19 Nov, 2014	source
		binary

1.4. 网站会自动推荐一个镜像站的下载地址。



1.5. 复制下载地址，在 chinahadoop0 虚机上，下载 hadoop-2.5.2.tar.gz 软件包。

执行命令 pwd 查看当前所处的目录位置。

执行命令 mkdir software 新建一个目录。

执行命令 cd software 进入 software 目录。

执行命令

wget http://mirror.bit.edu.cn/apache/hadoop/common/hadoop-2.5.2/hadoop-2.5.2.tar.gz

下载 hadoop-2.5.2.tar.gz 软件包（虚拟机连接到互联网才可以下载）。

```
[chinahadoop@chinahadoop0 ~]$ pwd
/home/chinahadoop
[chinahadoop@chinahadoop0 ~]$ mkdir software
[chinahadoop@chinahadoop0 ~]$ ls
software  公共的  模板  视频  图片  文档  下载  音乐  桌面
[chinahadoop@chinahadoop0 ~]$ cd software/
[chinahadoop@chinahadoop0 software]$ wget http://mirror.bit.edu.cn/apache/hadoop
/common/hadoop-2.5.2/hadoop-2.5.2.tar.gz
--2016-01-28 17:04:23--  http://mirror.bit.edu.cn/apache/hadoop/common/hadoop-2.
5.2/hadoop-2.5.2.tar.gz
正在解析主机 mirror.bit.edu.cn... 219.143.204.117, 2001:da8:204:2001:250:56ff:fe
a1:22
正在连接 mirror.bit.edu.cn|219.143.204.117|:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度：147197492 (140M) [application/octet-stream]
正在保存至：“ hadoop-2.5.2.tar.gz”

 4% [>                                       ] 6,806,964   2.07M/s eta(英国中部
 4% [>                                       ] 7,009,952   1.86M/s eta(英国中部
```

1.6. 下载完成后，执行命令 ls 看到刚刚下载的包，解压该包。

执行命令 tar zxvf hadoop-2.5.2.tar.gz

```
[chinahadoop@chinahadoop0 software]$ ls
hadoop-2.5.2.tar.gz
[chinahadoop@chinahadoop0 software]$ tar zxvf hadoop-2.5.2.tar.gz
[chinahadoop@chinahadoop0 software]$ ls
hadoop-2.5.2  hadoop-2.5.2.tar.gz
```

2. 配置 hosts

2.1. 查看主机名

查看当前机器的主机名，执行命令 hostname

查看配置文件中的主机名，执行命令 cat /etc/sysconfig/network

```
[chinahadoop@chinahadoop0 ~]$ hostname
chinahadoop0
[chinahadoop@chinahadoop0 ~]$ cat /etc/sysconfig/network
NETWORKING=yes
HOSTNAME=chinahadoop0
[chinahadoop@chinahadoop0 ~]$
```

主机名都是 chinahadoop0 就不需要修改了。

若想修改，则使用命令 hostname 新名字（即时生效，电脑重启后主机名会失效。）

还需修改 network 文件中 HOSTNAME 的值。（修改后的主机名永久生效。）

这里就不做修改了。

2.2. 查看 IP 地址

因为采用自动分配 IP 地址，它可能会发生变化，所以需要使用 ifconfig 命令再查下 IP 地址。

当然可以手动设置一个 IP 地址，必须保证当前网络中 IP 地址唯一，不能和其他电脑 IP 重复。

执行命令 ifconfig

```
[chinahadoop@chinahadoop0 ~]$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:36:0C:32
          inet addr:192.168.1.119  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe36:c32/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:785 errors:0 dropped:0 overruns:0 frame:0
          TX packets:318 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:81863 (79.9 KiB)  TX bytes:35570 (34.7 KiB)
```

2.3. 配置主机名和 IP 的映射关系

执行命令 sudo vim /etc/hosts
加入内容 192.168.1.119 chinahadoop0

```
1 chinahadoop0
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.1.106 win7
192.168.1.119 chinahadoop0
~
~
:wq
```

到此 hosts 文件就配置完了。

3. 修改 hadoop 配置文件

3.1. 配置文件存放位置

配置文件都在 hadoop 安装目录的 etc/hadoop/下面。

```
[chinahadoop@chinahadoop0 software]$ cd hadoop-2.5.2
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ls etc/hadoop/
capacity-scheduler.xml      httpfs-site.xml
configuration.xsl           log4j.properties
container-executor.cfg      mapred-env.cmd
core-site.xml               mapred-env.sh
hadoop-env.cmd              mapred-queues.xml.template
hadoop-env.sh               mapred-site.xml.template
hadoop-metrics2.properties  slaves
hadoop-metrics.properties   ssl-client.xml.example
hadoop-policy.xml           ssl-server.xml.example
hdfs-site.xml               yarn-env.cmd
httpfs-env.sh               yarn-env.sh
httpfs-log4j.properties     yarn-site.xml
httpfs-signature.secret
[chinahadoop@chinahadoop0 hadoop-2.5.2]$
```

3.2. 修改 slaves 文件

把 slaves 文件中 localhost 修改为 chinahadoop0

```
chinahadoop0
~
:wq
```

3.3. 修改 hadoop-env.sh 文件

把 hadoop-env.sh 文件中 JAVA_HOME 的值修改为 jdk 的安装目录。
使用命令echo $JAVA_HOME可以查看jdk的安装目录。

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ echo $JAVA_HOME
/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.95.x86_64
```

执行命令vim etc/hadoop/hadoop-env.sh
将 JAVA_HOME 的值修改为/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.95.x86_64

```
# 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=/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.95.x86_64

# The jsvc implementation to use. Jsvc is required to run secure datanodes.
#export JSVC_HOME=${JSVC_HOME}
:wq
```

3.4. 修改 mapred-site.xml 文件

执行命令 mv etc/hadoop/mapred-site.xml.template etc/hadoop/mapred-site.xml
就可以将文件 mapred-site.xml.template 重命名为 mapred-site.xml
输入命令 vim etc/hadoop/mapred-site.xml

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ mv etc/hadoop/mapred-site.xml.template
 etc/hadoop/mapred-site.xml
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ vim etc/hadoop/mapred-site.xml
```

加入下面内容

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

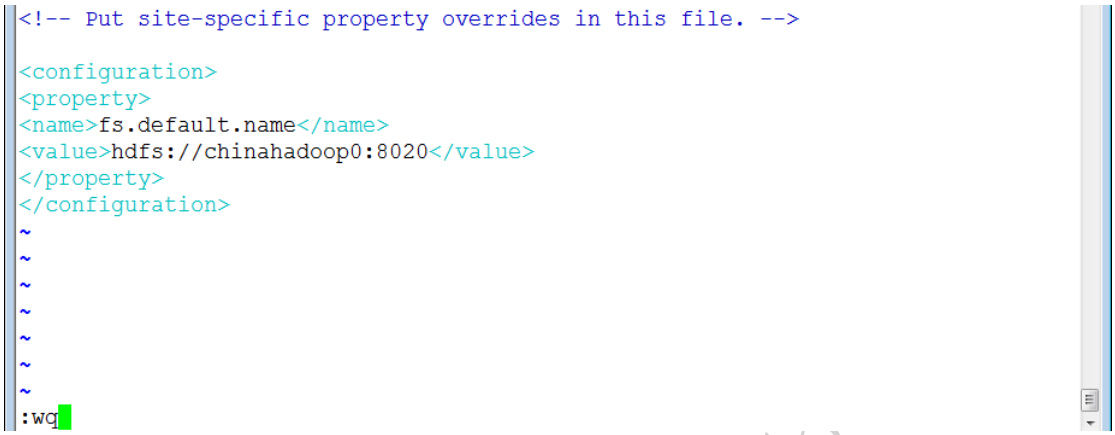
```
<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
</property>
</configuration>
~
~
~
~
~
~
~
:wq
```

3.5. 修改 core-site.xml 文件

查看当前机器的hostname

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ hostname
chinahadoop0
```


输入命令vim etc/hadoop/core-site.xml
加入下面内容
<property>
<name>fs.default.name</name>
<value>hdfs://chinahadoop0:8020</value>
</property>



3.6. 修改 hdfs-site.xml 文件

输入命令vim etc/hadoop/hdfs-site.xml
加入下面内容
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>/home/chinahadoop/dfs/name</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>/home/chinahadoop/dfs/data</value>
</property>

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

<configuration>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>/home/chinahadoop/dfs/name</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>/home/chinahadoop/dfs/data</value>
</property>
</configuration>
~
:wq
```

注意： 1、单机版副本数dfs.replication 的值默认是3 这里写为1

2、dfs.namenode.name.dir 和dfs.datanode.data.dir 的默认值，在hadoop 安装目录下的tmp 目录下。

3、这里修改为非tmp 目录，此目录无需存在。它是在启动hadoop 时目录是自动创建的。

3.7. 修改 yarn-site.xml 文件

输入命令vim etc/hadoop/yarn-site.xml

加入下面内容

<property>

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

<value>mapreduce_shuffle</value>

</property>

```
<configuration>

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

</configuration>
~
~
~
~
~
~
~
~
:wq
```

4. 搭建伪分布环境

4.1. 格式化 namenode

第一次搭建hadoop环境，需要格式化。
执行命令bin/hadoop namenode -format

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ pwd
/home/chinahadoop/software/hadoop-2.5.2
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ls /home/chinahadoop/
software  公共的  模板  视频  图片  文档  下载  音乐  桌面
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ bin/hadoop namenode -format
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

16/01/29 12:08:12 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = chinahadoop0/192.168.1.119
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 2.5.2
STARTUP_MSG:   classpath = /home/chinahadoop/software/hadoop-2.5.2/etc/hadoop:/h
```

……

```
16/01/29 12:08:17 INFO common.Storage: Storage directory /home/chinahadoop/dfs/n
ame has been successfully formatted.
16/01/29 12:08:17 INFO namenode.NNStorageRetentionManager: Going to retain 1 ima
ges with txid >= 0
16/01/29 12:08:17 INFO util.ExitUtil: Exiting with status 0
16/01/29 12:08:17 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at chinahadoop0/192.168.1.119
************************************************************/
```

从控制台输出的信息上可以看到
16/01/29 12:08:17 INFO common.Storage: Storage directory /home/chinahadoop/dfs/name has been successfully formatted.
说明格式化成功，并且配置文件中配置的目录/home/chinahadoop/dfs/name 已经被创建。

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ls /home/chinahadoop/
dfs  software  公共的  模板  视频  图片  文档  下载  音乐  桌面
[chinahadoop@chinahadoop0 hadoop-2.5.2]$
```

4.2. 启动 namenode

执行命令 sbin/hadoop-daemon.sh start namenode
同时 jps 查看下

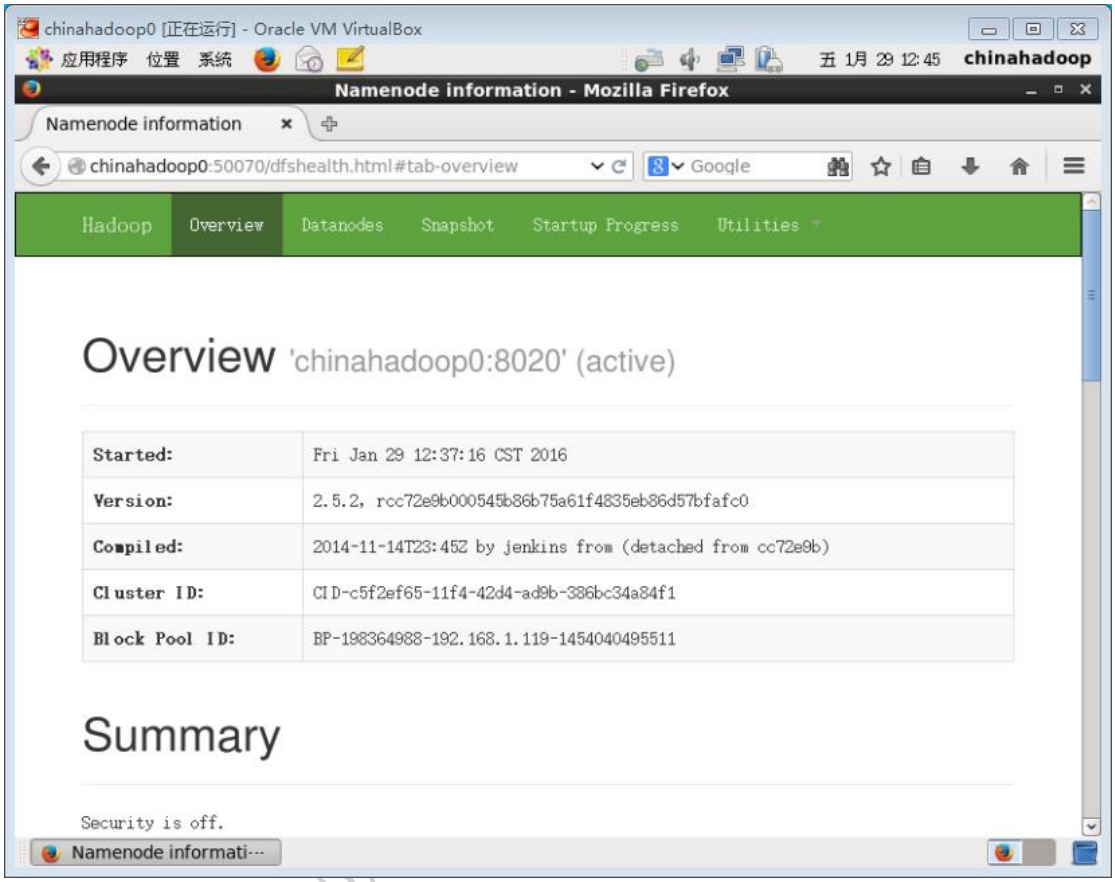
```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ jps
3237 Jps
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ sbin/hadoop-daemon.sh start namenode
starting namenode, logging to /home/chinahadoop/software/hadoop-2.5.2/logs/hadoo
p-chinahadoop-namenode-chinahadoop0.out
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ jps
3268 NameNode
3331 Jps
```

4.3. 启动 datanode

执行命令 sbin/hadoop-daemon.sh start datanode

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ sbin/hadoop-daemon.sh start datanode
starting datanode, logging to /home/chinahadoop/software/hadoop-2.5.2/logs/hadoo
p-chinahadoop-datanode-chinahadoop0.out
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ jps
3268 NameNode
3357 DataNode
3429 Jps
```

因为虚拟机已经配置了主机名和IP的映射关系，所以在Centos6.6虚拟机的浏览器窗口中输入http://chinahadoop0:50070

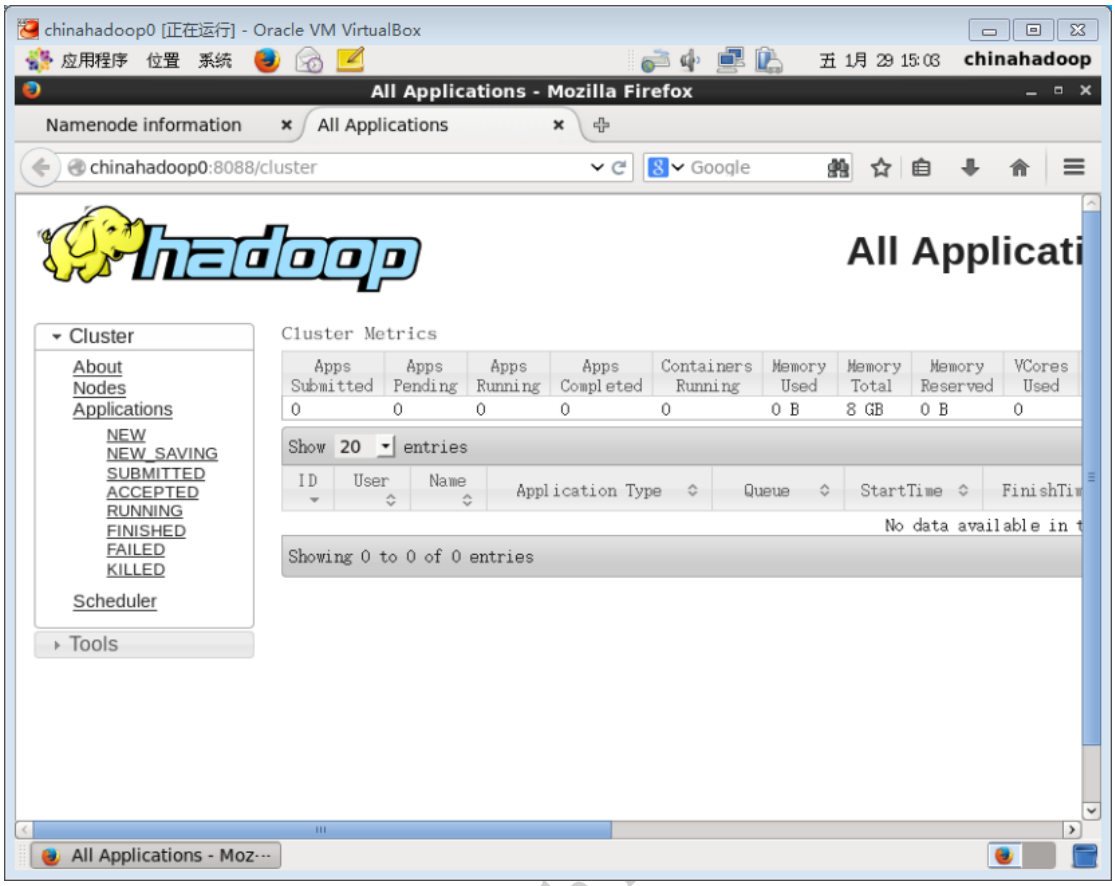


4.4. 启动 yarn

执行命令 sbin/start-yarn.sh

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ./sbin/start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /home/chinahadoop/software/hadoop-2.5.2/log
s/yarn-chinahadoop-resourcemanager-chinahadoop0.out
chinahadoop@chinahadoop0's password: 需要输入密码
chinahadoop0: starting nodemanager, logging to /home/chinahadoop/software/hadoop
-2.5.2/logs/yarn-chinahadoop-nodemanager-chinahadoop0.out
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ jps
3268 NameNode
7743 Jps
3357 DataNode
7617 NodeManager
7320 ResourceManager
[chinahadoop@chinahadoop0 hadoop-2.5.2]$
```

在Centos6.6虚拟机的浏览器窗口中输入http://chinahadoop0:8088



5. 执行一个 MapReduce 任务

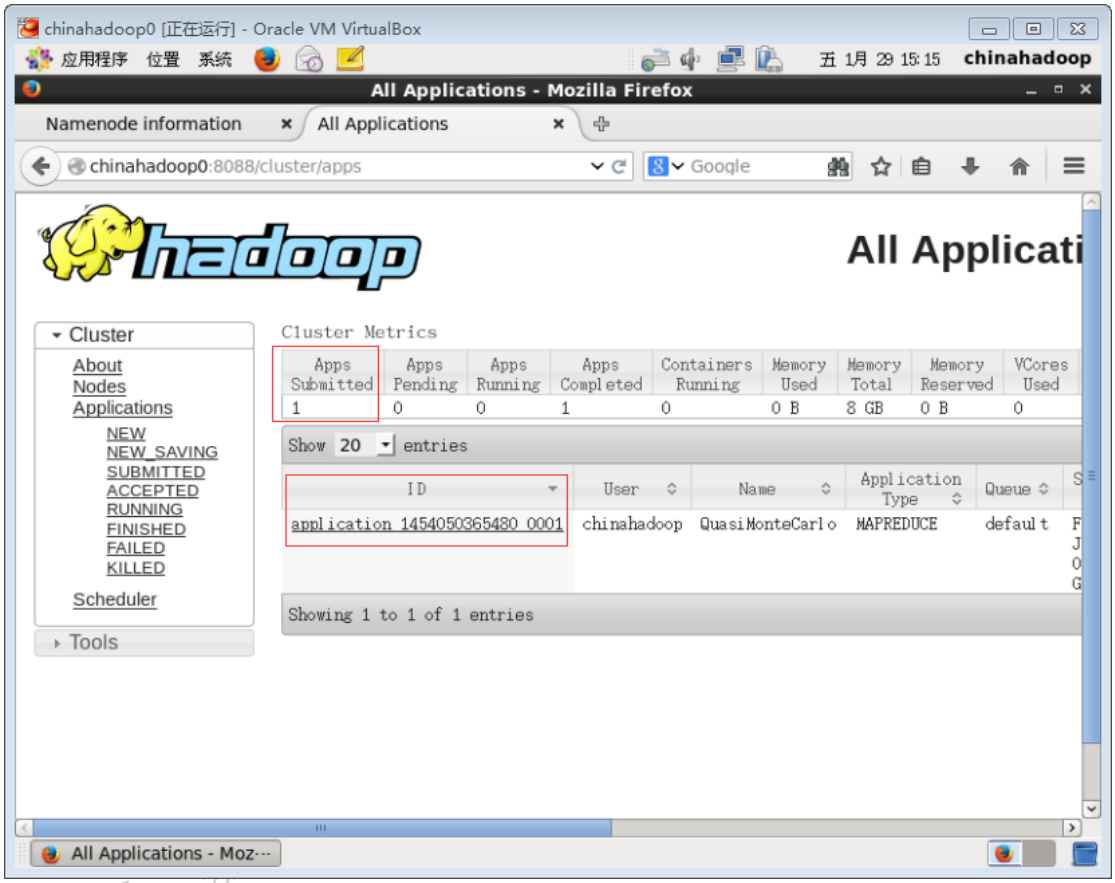
执行命令 bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.5.2.jar pi 2 10

执行任务的信息如下：

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.5.2.jar pi 2 10
Number of Maps  = 2
Samples per Map = 10
16/01/29 15:05:53 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Wrote input for Map #0
Wrote input for Map #1
Starting Job
16/01/29 15:05:55 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
16/01/29 15:05:56 INFO input.FileInputFormat: Total input paths to process : 2
16/01/29 15:05:56 INFO mapreduce.JobSubmitter: number of splits:2
16/01/29 15:05:57 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1454050365480_0001
16/01/29 15:05:57 INFO impl.YarnClientImpl: Submitted application application_1454050365480_0001
16/01/29 15:05:57 INFO mapreduce.Job: The url to track the job: http://chinahadoop0:8088/proxy/application_1454050365480_0001/
16/01/29 15:05:57 INFO mapreduce.Job: Running job: job_1454050365480_0001
```

```
16/01/29 15:06:19 INFO mapreduce.Job: Job job_1454050365480_0001 running in uber mode : false
16/01/29 15:06:20 INFO mapreduce.Job:  map 0% reduce 0%
16/01/29 15:08:58 INFO mapreduce.Job:  map 100% reduce 0%
16/01/29 15:10:00 INFO mapreduce.Job:  map 100% reduce 100%
16/01/29 15:10:03 INFO mapreduce.Job: Job job_1454050365480_0001 completed successfully
16/01/29 15:10:05 INFO mapreduce.Job: Counters: 49
    File System Counters
        FILE: Number of bytes read=50
        FILE: Number of bytes written=292413
        FILE: Number of read operations=0
        FILE: Number of large read operations=0
        FILE: Number of write operations=0
        HDFS: Number of bytes read=548
        HDFS: Number of bytes written=215
        HDFS: Number of read operations=11
        HDFS: Number of large read operations=0
        HDFS: Number of write operations=3
    Job Counters
        Launched map tasks=2
        Launched reduce tasks=1
        Data-local map tasks=2
        Total time spent by all maps in occupied slots (ms)=346087
        Total time spent by all reduces in occupied slots (ms)=26951
        Total time spent by all map tasks (ms)=346087
        Total time spent by all reduce tasks (ms)=26951
        Total vcore-seconds taken by all map tasks=346087
        Total vcore-seconds taken by all reduce tasks=26951
        Total megabyte-seconds taken by all map tasks=354393088
        Total megabyte-seconds taken by all reduce tasks=27597824
    Map-Reduce Framework
        Map input records=2
        Map output records=4
        Map output bytes=36
        Map output materialized bytes=56
        Input split bytes=312
        Combine input records=0
        Combine output records=0
        Reduce input groups=2
        Reduce shuffle bytes=56
        Reduce input records=4
        Reduce output records=0
        Spilled Records=8
        Shuffled Maps =2
        Failed Shuffles=0
        Merged Map outputs=2
        GC time elapsed (ms)=6021
        CPU time spent (ms)=4270
        Physical memory (bytes) snapshot=383336448
        Virtual memory (bytes) snapshot=2925305856
        Total committed heap usage (bytes)=257433600
    Shuffle Errors
        BAD_ID=0
        CONNECTION=0
```


IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters
Bytes Read=236
File Output Format Counters
Bytes Written=97
Job Finished in 250.186 seconds
Estimated value of Pi is 3.80000000000000000000
在Centos6.6虚拟机的浏览器窗口中刷新http://chinahadoop0:8088
可以看到一个任务已经被提交。



6. 在 win7 的浏览器中访问 hadoop 集群

6.1. 关闭 Centos6.6 虚拟机的防火墙

hosts 文件已经配置好 chinahadoop0 的 IP 映射关系。
即时生效：关闭防火墙 sudo service iptables stop
查看防火墙状态 sudo service iptables status

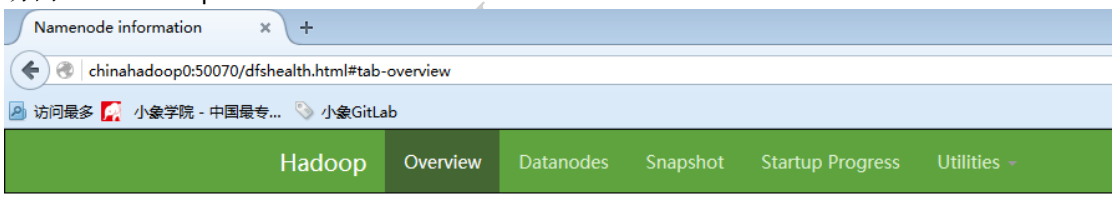
```
[chinahadoop@chinahadoop0 ~]$ sudo service iptables status
表格：filter
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0           state RELATED,
ESTABLISHED
2    ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0
3    ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
4    ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0           state NEW tcp
dpt:22
5    REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with ic
mp-host-prohibited

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination
1    REJECT     all  --  0.0.0.0/0            0.0.0.0/0           reject-with ic
mp-host-prohibited

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination

[chinahadoop@chinahadoop0 ~]$ sudo service iptables stop
iptables：将链设置为政策 ACCEPT：filter                    [确定]
iptables：清除防火墙规则：                                 [确定]
iptables：正在卸载模块：                                   [确定]
[chinahadoop@chinahadoop0 ~]$ sudo service iptables status
iptables：未运行防火墙。
```

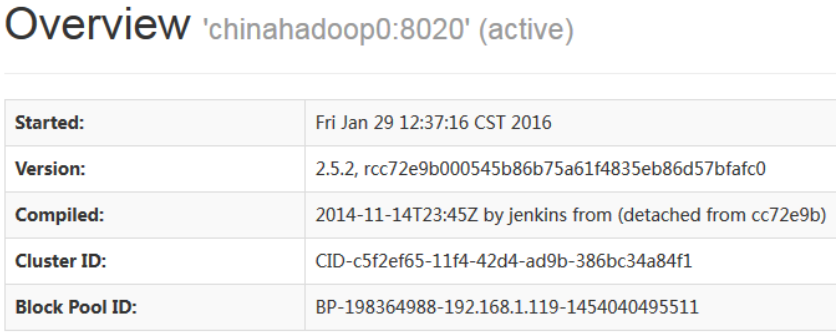
永久生效：关闭防火墙，执行命令 sudo chkconfig iptables off

```
[chinahadoop@chinahadoop0 ~]$ sudo chkconfig iptables off
[chinahadoop@chinahadoop0 ~]$
```

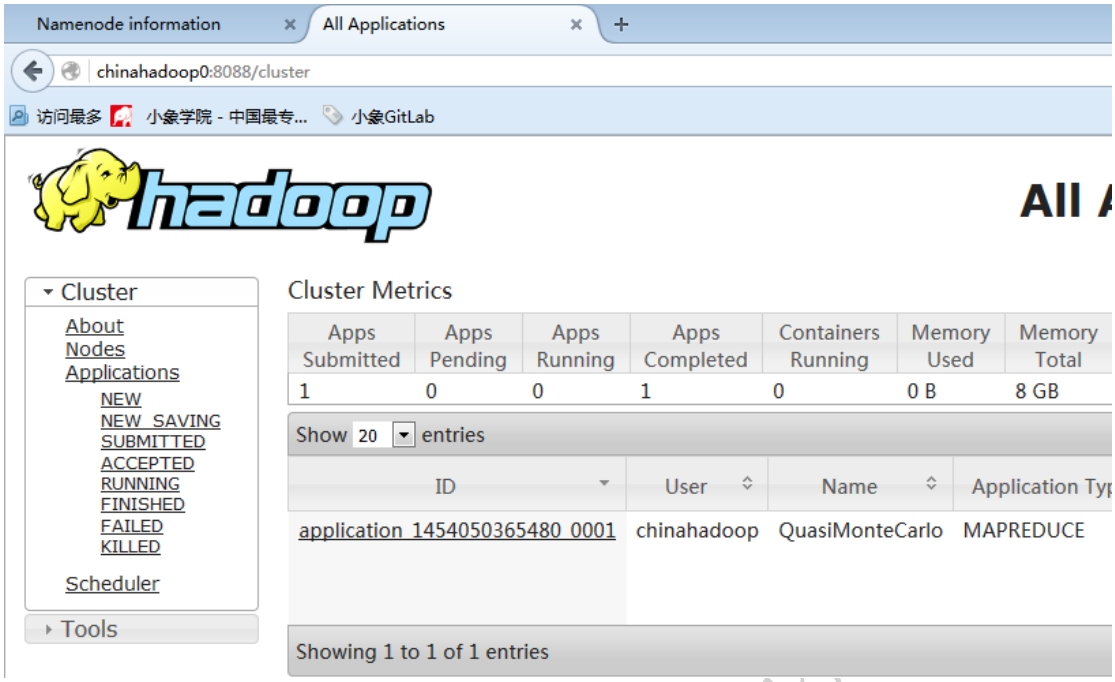
6.2. 在 win7 的浏览器中访问

访问 chinahadoop0:50070





访问 chinahadoop0:8088



7. 停止集群

执行命令 sbin/stop-yarn.sh
执行命令 sbin/hadoop-daemon.sh stop datanode
执行命令 sbin/hadoop-daemon.sh stop namenode

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ sbin/stop-yarn.sh
stopping yarn daemons
stopping resourcemanager
chinahadoop@chinahadoop0's password: 输入密码
chinahadoop0: stopping nodemanager
no proxyserver to stop
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ sbin/hadoop-daemon.sh stop datanode
stopping datanode
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ sbin/hadoop-daemon.sh stop namenode
stopping namenode
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ jps
3046 Jps
[chinahadoop@chinahadoop0 hadoop-2.5.2]$
```

8. 配置免密码登录

8.1. 生成密钥

通过演示，能看出在启动和停止 yarn 的时候需要输入密码，因此需要配置免密码登录。
默认生成 rsa 类型的密钥，可直接执行命令 ssh-keygen
也可以执行命令 ssh-keygen -t rsa

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ssh 这里按两下 Tab 键，就会出现下面的列表。
ssh          ssh-agent    sshd         ssh-keyscan
ssh-add      ssh-copy-id  ssh-keygen
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ssh-keygen -t rsa 按回车键
Generating public/private rsa key pair.
Enter file in which to save the key (/home/chinahadoop/.ssh/id_rsa): 按回车键
Enter passphrase (empty for no passphrase): 按回车键
Enter same passphrase again: 按回车键
Your identification has been saved in /home/chinahadoop/.ssh/id_rsa.
Your public key has been saved in /home/chinahadoop/.ssh/id_rsa.pub.
The key fingerprint is:
d8:ea:70:41:15:d3:0c:af:b1:ee:66:c3:45:b7:d4:18 chinahadoop@chinahadoop0
The key's randomart image is:
+--[ RSA 2048]----+
|         ==      |
|        . oo E   |
|       . . .  +  |
|      . o +. + . |
|       o S. o .  |
|        +  . .   |
|     . o...      |
|      + .=       |
|       .o..      |
+-----------------+
[chinahadoop@chinahadoop0 hadoop-2.5.2]$
```

8.2. 拷贝密钥

执行命令 ssh-copy-id
根据提示信息，找到 ssh-copy-id 文件。

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ls ~/.ssh/
id_rsa  id_rsa.pub
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ssh-copy-id
Usage: /usr/bin/ssh-copy-id [-i [identity_file]] [user@]machine
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ sudo vim /usr/bin/ssh-copy-id
```

使用 vim 查看下脚本内容，是因为有颜色区分，看的更清晰些。
首先，看到了默认值 ID_FILE

```
#!/bin/sh

# Shell script to install your public key on a remote machine
# Takes the remote machine name as an argument.
# Obviously, the remote machine must accept password authentication,
# or one of the other keys in your ssh-agent, for this to work.

ID_FILE="${HOME}/.ssh/id_rsa.pub"
```

其次，发现刚刚执行命令时，控制台输出的信息。

```
if [ "$#" -lt 1 ] || [ "$1" = "-h" ] || [ "$1" = "--help" ]; then
  echo "Usage: $0 [-i [identity_file]] [user@]machine" >&2 这里是控制台输出的信息
  exit 1
fi
        因为该命令需要正确输入参数，脚本才能往下执行。

{ eval "$GET_ID" ; } | ssh $1 "umask 077; test -d ~/.ssh || mkdir ~/.ssh ; cat >
> ~/.ssh/authorized_keys && (test -x /sbin/restorecon && /sbin/restorecon ~/.ssh
 ~/.ssh/authorized_keys >/dev/null 2>&1 || true)" || exit 1
```

最后，查看完后，强制退出即可，不要修改文件内容。强制退出命令:q!
总结五点：一、identity_file 的默认值是 ${HOME}/.ssh/id_rsa.pub
二、中括号 [ ] 是可选参数，
三、因为当前是 chinahadoop 用户登录，所以${HOME}=/home/chinahadoop
四、[user@]这里不输入默认是当前登录的用户 chinahadoop
五、machine 这里使用主机名。
得出结论，执行命令 ssh-copy-id chinahadoop0 即可

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ssh-copy-id chinahadoop0
The authenticity of host 'chinahadoop0 (192.168.1.119)' can't be established.
RSA key fingerprint is b7:34:e0:14:85:24:e3:e0:54:8d:85:2f:7c:9f:ca:7a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'chinahadoop0,192.168.1.119' (RSA) to the list of kno
wn hosts.
chinahadoop@chinahadoop0's password: 输入密码
Now try logging into the machine, with "ssh 'chinahadoop0'", and check in:

  .ssh/authorized_keys

to make sure we haven't added extra keys that you weren't expecting.

[chinahadoop@chinahadoop0 hadoop-2.5.2]$
```

查看那.ssh 目录下都什么文件，执行命令 ls ~/.ssh/

当前是 chinahadoop 用户登录的，则该命令等价于 ls /home/chinahadoop/.ssh/

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ls ~/.ssh/
authorized_keys  id_rsa  id_rsa.pub  known_hosts
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ls /home/chinahadoop/.ssh/
authorized_keys  id_rsa  id_rsa.pub  known_hosts
[chinahadoop@chinahadoop0 hadoop-2.5.2]$
```

查看 authorized_keys 文件内容，执行命令 cat ~/.ssh/authorized_keys

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAlLEujqqfTQLONauLWfsG7LHpVCVy7UBenZTXxmxwxW+f
BLk03Jn7xfM6OmsHUou9UkM+TBzlpAO5s7/RejZyOmWG4H3f8bTwxyY2GKFyCLifptRh2iA8l5D+3f8L
b4jh5NNw1Weu17zB4MLWc6DakkCpp5x9y45zzNhi6j/H6dpw4h+1B2DdTdkZ/O/u+Bj5Smg34vR0BBPs
EtCX1oMt8/jS0A461ikNUuV8/prp6RmBicv7cMKoCjdl7umS4gMD3JwYQkFPfPeEp4GOxQkAx5EbIEe8
nlTq/n1D7hncSJ6Ml2XAAW/tr4FS5y+eEuSyYbTrEVSW1D3MRyt8fyMnWw== chinahadoop@chinaha
doop0
[chinahadoop@chinahadoop0 hadoop-2.5.2]$
```

执行命令 ssh chinahadoop0 验证下

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ssh chinahadoop0
Last login: Sat Jan 30 17:02:29 2016 from chinahadoop0
[chinahadoop@chinahadoop0 ~]$ exit
logout
Connection to chinahadoop0 closed.
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ ssh chinahadoop0
Last login: Sat Jan 30 17:02:39 2016 from chinahadoop0
[chinahadoop@chinahadoop0 ~]$
```

8.3. 再次启动集群

再次启动集群，发现不需要再输入密码。

```
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ sbin/hadoop-daemon.sh start namenode
starting namenode, logging to /home/chinahadoop/software/hadoop-2.5.2/logs/hadoo
p-chinahadoop-namenode-chinahadoop0.out
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ sbin/hadoop-daemon.sh start datanode
starting datanode, logging to /home/chinahadoop/software/hadoop-2.5.2/logs/hadoo
p-chinahadoop-datanode-chinahadoop0.out
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ sbin/start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /home/chinahadoop/software/hadoop-2.5.2/log
s/yarn-chinahadoop-resourcemanager-chinahadoop0.out
The authenticity of host 'chinahadoop0 (192.168.1.119)' can't be established.
RSA key fingerprint is b7:34:e0:14:85:24:e3:e0:54:8d:85:2f:7c:9f:ca:7a.
Are you sure you want to continue connecting (yes/no)? yes
chinahadoop0: Warning: Permanently added 'chinahadoop0,192.168.1.119' (RSA) to t
he list of known hosts.
chinahadoop0: starting nodemanager, logging to /home/chinahadoop/software/hadoop
-2.5.2/logs/yarn-chinahadoop-nodemanager-chinahadoop0.out
[chinahadoop@chinahadoop0 hadoop-2.5.2]$ jps
3732 ResourceManager
4147 Jps
3534 NameNode
3911 NodeManager
3617 DataNode
[chinahadoop@chinahadoop0 hadoop-2.5.2]$
```