Hadoop 伪分布式搭建

笔者：电杆

2018 年 10 月 10 日

目录

1. Hadoop伪分布式部署 1
1.1安装工具 1
1.2 安装环境搭建 2
1.3 hadoop 安装 7
1.4 hadoop伪分布式配置 9

1. Hadoop伪分布式部署

参考文档：

https://blog.csdn.net/c1481118216/article/details/73326049

https://blog.csdn.net/hliq5399/article/details/78193113

1.1 安装工具

1.1.1 CentOS7 操作系统

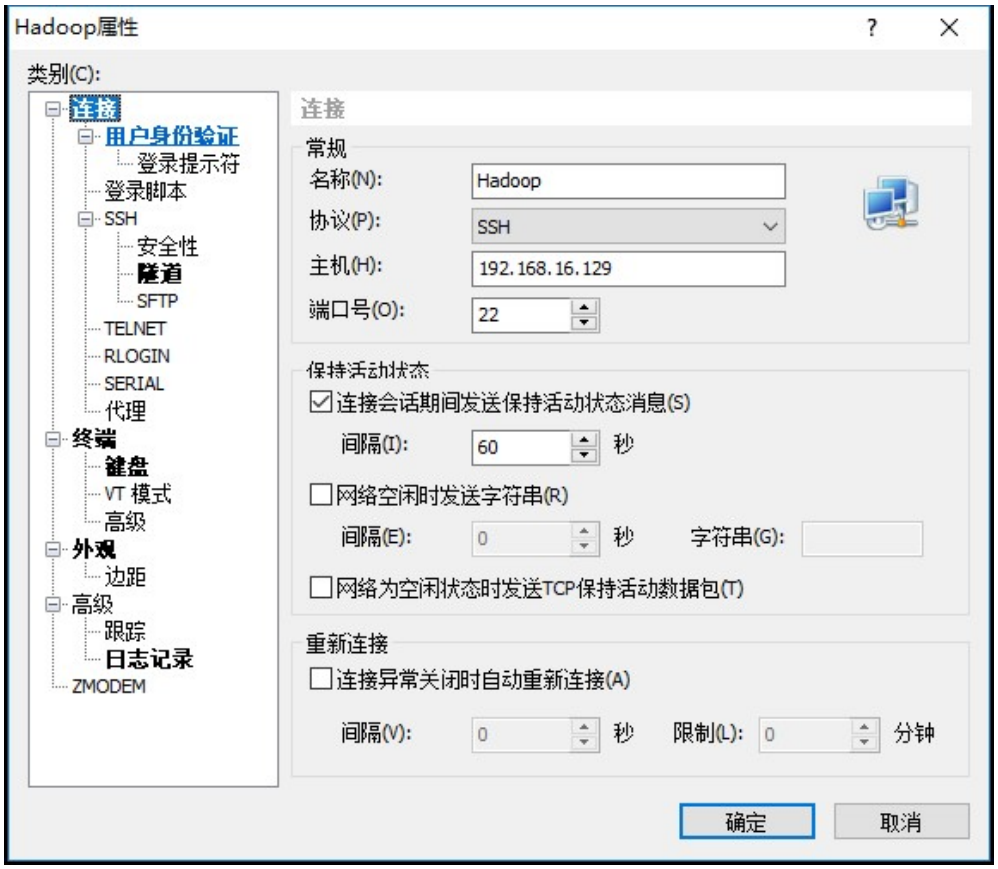
Hadoop 是运行在 Linux，虽然借助工具也可以运行在 Windows 上，但是建议还是运行在 Linux 系统上，所以需要先安装 CentOS 系统在 VMware 上，安装 CentOS 操作可参考

https://blog.csdn.net/yiyihuazi/article/details/78557216

1.1.2 Xshell 工具

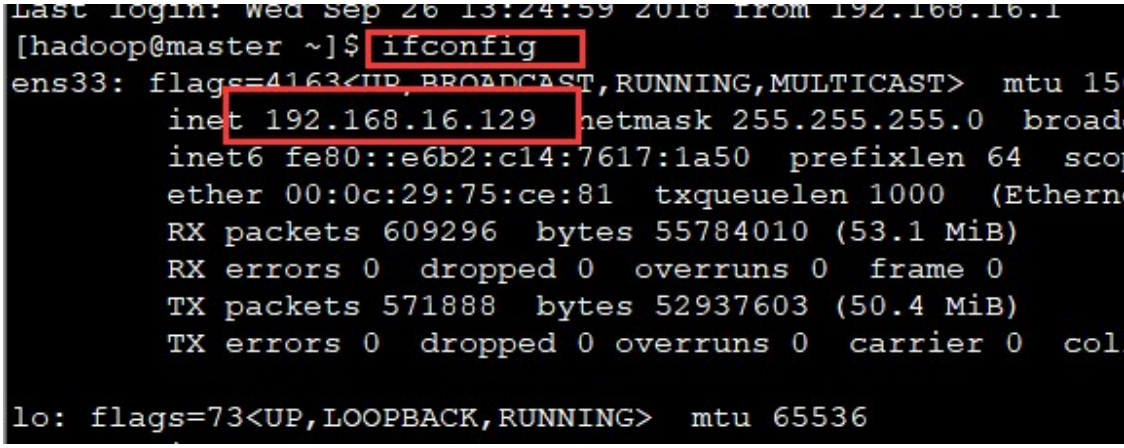
用于连接虚拟机，执行操作命令。

1. 安装 xshell 后，连接虚拟机，配置连接：



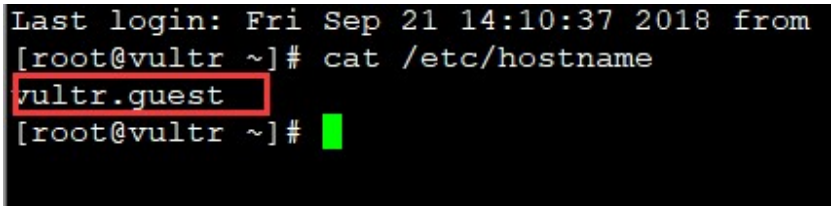
找到虚拟机 IP， 更改 hostname

$ ifconfig



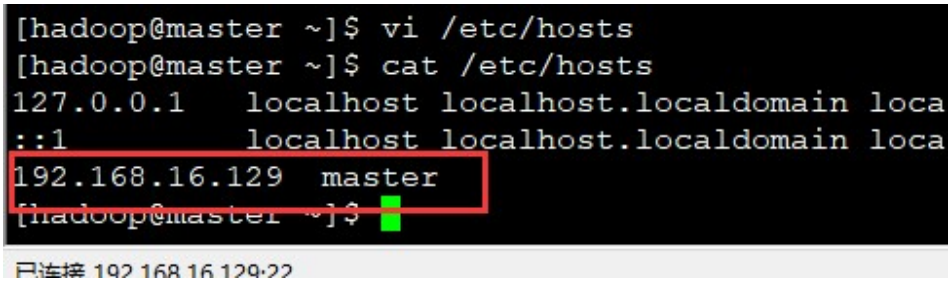
[hadoop@master ~]$ vi /etc/hostname

将下图中改为自己 hostname，



使用命令

[hadoop@master ~]$ vi /etc/hosts，在最下面添加自己更改的用户名



1.2 安装环境搭建

1.2.1 创建 hadoop 用户并赋予权限

1、使用 root 用户登录并执行：

[root@localhost ~]# useradd -m hadoop -s /bin/bash

[root@localhost ~]# passwd Hadoop

连续两次输入密码，以设定密码，

2、 更改 hadoop 用户权限

使用 root 用户登录执行：

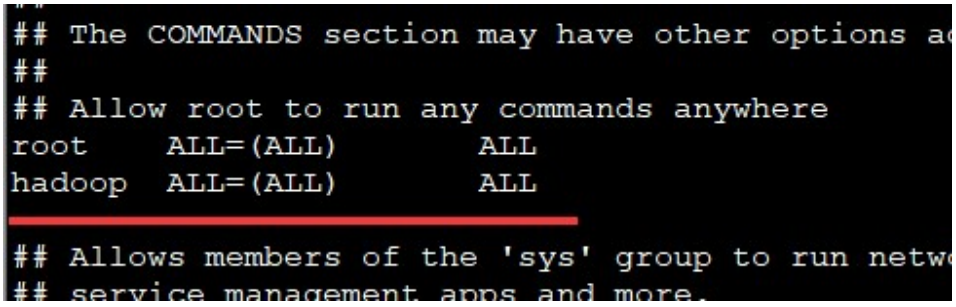
[root@master ~]# visudo

找到：

root ALL=(ALL) ALL

并在下面添加刚刚添加的用户：

hadoop ALL=(ALL) ALL



然后退出保存。

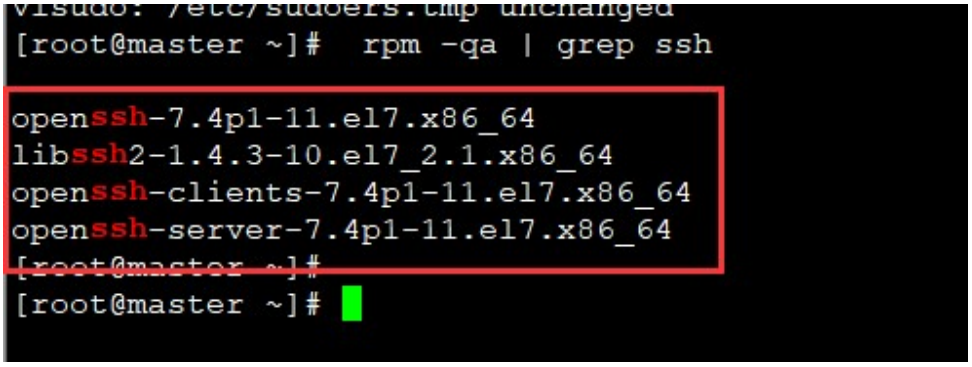
然后退出 root 用户，用 hadoop 登录

1.2.2 安装 SSH，配置 SSH 免密登录

一般情况下 CentOS7 默认安装了 SSH ，可使用以下代码检查是否安装：

[root@master ~]# rpm -qa | grep ssh

如图显示责任安装成功



如果未安装可执行 yum 进行安装：

sudo yum install openssh-clients

sudo yum install openssh-server

安装过程中会让输入 [y/N]，输入 y 即 可）：

安装完成 执行：

[root@master ~]# ssh localhost

此时会有如下提示(SSH 首次登陆提示)，输入 y es 。然后按提示输入密码，这样 就登陆到本机了。

```
[root@master ~]# ssh localhost
The authenticity of host 'localhost (::1)' can't be established.
ECDSA key fingerprint is SHA256:vWcB0nWUwuBLAnPVwHoFAo0j3YxKNSHID8LFKOC2sOQ
ECDSA key fingerprint is MD5:71:0d:e0:8b:4e:ff:6d:06:df:5c:cd:2c:9b:9c:fc:1
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
root@localhost's password:
Last login: Wed Sep 26 15:04:30 2018 from 192.168.16.1
[root@master ~]#
```

配置 SSH 免密登录

执行：

[root@master ~]# exit

退出刚刚的连接

```
[root@master ~]# exit
logout
Connection to localhost closed.
[root@master ~]#
```

进入.ssh 文件夹：

[root@master ~]# cd ~/.ssh/

```
[root@master ~]#  cd ~/.ssh/
[root@master .ssh]#
```

若如该目录。可在执行一次 ssh 然后在进入此目录

执行：

[root@master .ssh]# ssh-keygen -t rsa

```
Connection to localhost closed.
[root@master ~]#  cd ~/.ssh/
[root@master .ssh]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:5kaIbNkG9UayoO43lonIihYw9tOsTWjGtchDYq7rsiI root@mas
The key's randomart image is:
+---[RSA 2048]----+
|     . o .       |
|    . o =        |
|   . . . o       |
|o= o * o         |
|=.B @ = S        |
|.= & O +         |
|o.* %   o        |
|E. + o .         |
|@+               |
+----[SHA256]-----+
[root@master .ssh]#
```

[root@master .ssh]# cat id_rsa.pub >> authorized_keys

加入授权

[root@master .ssh]# chmod 600 ./authorized_keys

修改文件权限

```
|@+               |
+----[SHA256]-----+
[root@master .ssh]#  cat id_rsa.pub >> authorized_keys
[root@master .ssh]# chmod 600 ./authorized_keys
```

测试是否配置成功：

[root@master .ssh]# ssh localhost

Last login: Wed Sep 26 15:13:21 2018 from ::1

```
[root@master .ssh]#  cat id_rsa.pub >> authorized_keys
[root@master .ssh]# chmod 600 ./authorized_keys
[root@master .ssh]#  ssh localhost
Last login: Wed Sep 26 15:13:21 2018 from ::1
[root@master ~]#
```

1.2.3 安装 java JDK

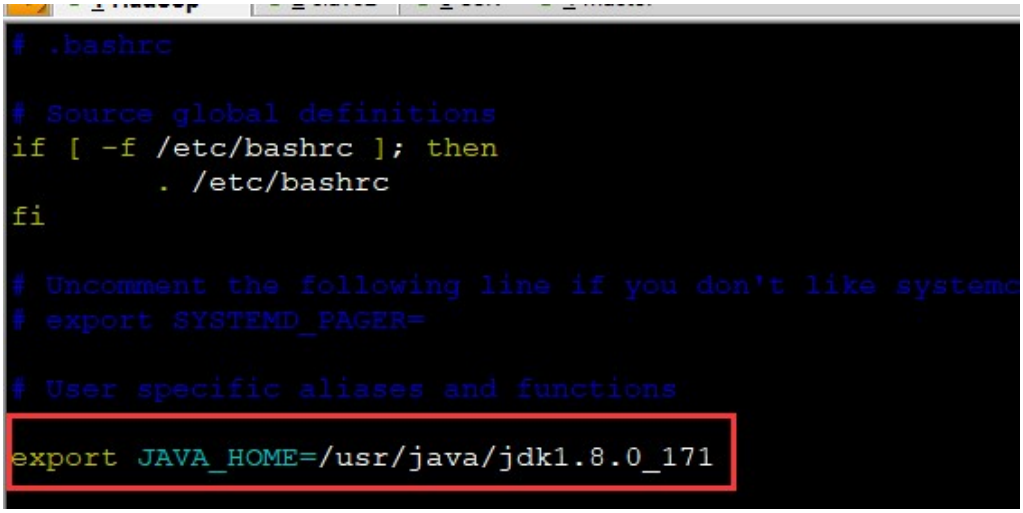
部署需要我们安装 JAVA JDK，在此处我们安装 OpenJDK 1.8。安装过程中会让输入 hadoop 密码和[y/N]输入 y 即可，

[hadoop@master ~]$ sudo yum install java-1.8.0openjdk java-1.8.0-openjdk-devel

默认安装位置为 /usr/lib/jvm/java-1.7.0-openjdk（该路径 可以通过执行 rpm -ql java-1.8.0-openjdk-devel | grep '/bin/javac' 命 令确定，执行后会输出一个路径，除去路径末尾的 “/bin/javac”，剩下的就是正确的路径 了）

执行命令，添加自己的 java 路径

[hadoop@master ~]$ vim ~/.bashrc



[hadoop@master ~]$ source ~/.bashrc

设置好后我们来检验一下是否设置正确：

[hadoop@master ~]$ echo $JAVA_HOME

[hadoop@master ~]$ java -version

[hadoop@master ~]$ $JAVA_HOME/bin/java –version

如果设置正确的话，$JAVA_HOME/bin/java -version 会输出 java 的版本信息，且 和 java -version 的输出结果一样，如下图所示

1.2.4 关闭防火墙与 Selinux

1、关闭防火墙

关闭防火墙或将相关端口加入

查看防火墙状态：

```
firewall-cmd    --state
```

关闭防火墙:

//临时关闭

systemctl stop firewalld

//禁止开机启动

systemctl disable firewalld

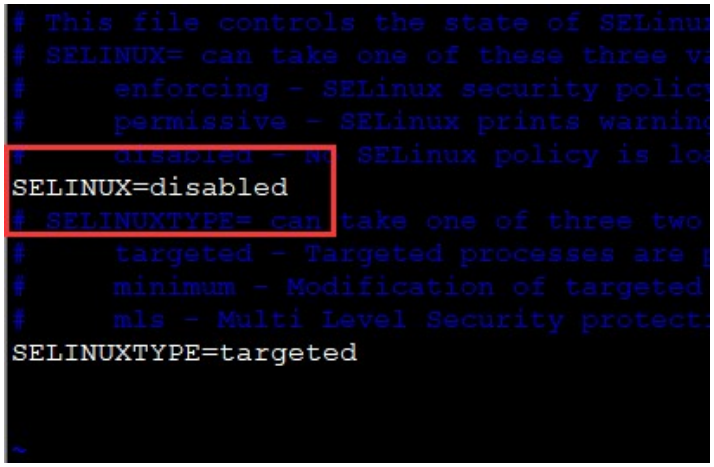
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.

Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.

2、修改：

将文件中的值修改如下图： SELINUX=disabled

[hadoop@master hadoop]$ vi /etc/selinux/config



1.3 hadoop 安装

1.3.1 hadoop 下载

在下载 hadoop 之前需要安装 wget 包才可以进行下载,

[hadoop@master~]$ sudo yum -y install wget

也可在镜像网址下载，然后通过 ftp 工具拷贝到虚拟机中

https://mirrors.cnnic.cn/apache/hadoop/common/

下载 hadoop

[hadoop@master ~]$ wget https://mirrors.cnnic.cn/apache/hadoop/common/hadoop3.1.0/hadoop-3.1.0.tar.gz

拷贝 Hadoop 至 /usr/local/ 中：

[hadoop@master ~]$ sudo tar -zxf hadoop-3.1.0.tar.gz -C /usr/local

解压到/usr/local/

[hadoop@master ~]$ cd /usr/local/

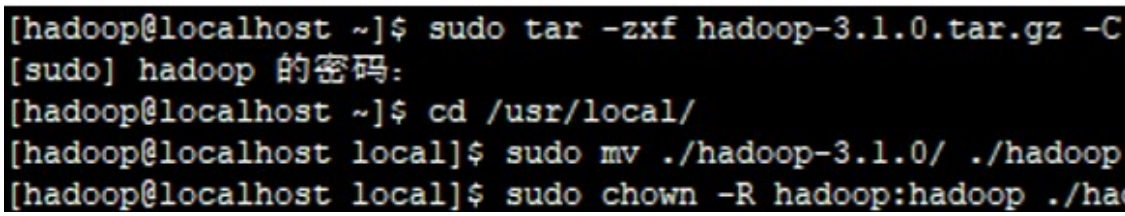
进入该文件夹

[hadoop@master local]$ sudo mv ./hadoop-3.1.0/ ./Hadoop

修改目录名称

[hadoop@master local]$ sudo chown -R hadoop: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

[hadoop@master local]$ cd /usr/local/Hadoop

[hadoop@master hadoop]$ ./bin/hadoop version

```
[hadoop@master bin]$ hadoop version
Hadoop 3.1.0
Source code repository https://github.com/apache/hadoop -r 16b70619a24cdcf
Compiled by centos on 2018-03-30T00:00Z
Compiled with protoc 2.5.0
From source with checksum 14182d20c972b3e2105580a1ad6990
This command was run using /usr/local/hadoop/share/hadoop/common/hadoop-co
[hadoop@master bin]$
```

如图显示则安装成功。

1.4 hadoop 伪分布式配置

1.4.1 hadoop 环境变量配置

在设置 Hadoop 伪分布式配置前，我们还需要设置 HADOOP 环境变量，执行如下命令在~/.bashrc 配置：

[hadoop@master bin]$ vim ~/.bashrc

在末尾添加：

export HADOOP_HOME=/usr/local/hadoop

export HADOOP_INSTALL=$HADOOP_HOME

export HADOOP_MAPRED_HOME=$HADOOP_HOME

export HADOOP_COMMON_HOME=$HADOOP_HOME

export HADOOP_HDFS_HOME=$HADOOP_HOME

export YARN_HOME=$HADOOP_HOME

export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native

export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin

```
# User specific aliases and functions

export JAVA_HOME=/usr/java/jdk1.8.0_171

export HADOOP_HOME=/usr/local/hadoop
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
```

使修改生效：

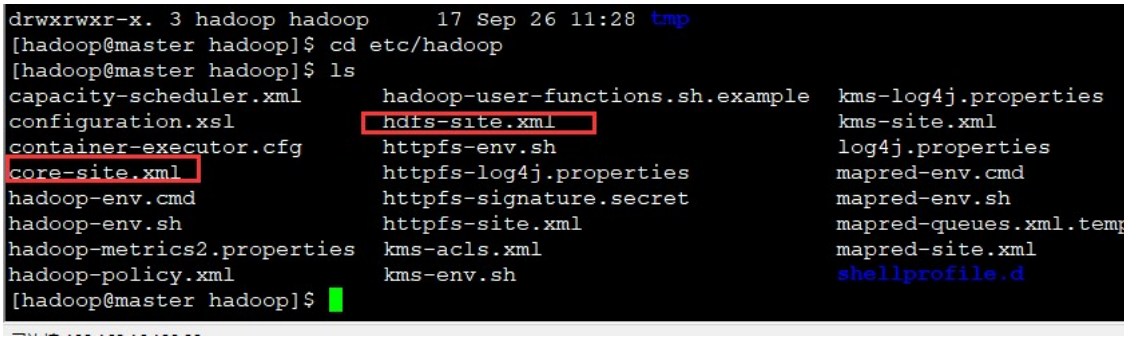
[hadoop@master hadoop]$ source ~/.bashrc

1.4.2 修改配置文件

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 。Hadoop 的配置文件是 xml 格式，每 个配置以声明 property 的 name 和 value 的方式来实现

进入配置文件目录

[hadoop@master hadoop]$ cd etc/hadoop



需要修改的问框起来的

1、修改 core-site.xml

[hadoop@master hadoop]$ vim core-site.xml

添加配置

```
<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://master:9000</value>
    </property>
```

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

<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://master:9000</value>
        </property>
</configuration>
```

2、 修改 hdfs-site.xml

[hadoop@master hadoop]$ vim hdfs-site.xml

添加：

<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> <property>

<name>dfs.http.address</name>

<value>192.168.16.129:50070</value>

</property>

```
<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>  <property>
        <name>dfs.http.address</name>
        <value>192.168.16.129:50070</value>
</property>
</configuration>
"hdfs-site.xml" 34L, 1251C
```

配置完成后，返回两级，回到 local 下面的 hadoop 中执行 NameNode 的格式化:

[hadoop@localhost hadoop]$ cd /usr/local/Hadoop

[hadoop@localhost hadoop]$ ./bin/hdfs namenode -format

大致查看以下输出：发现 successfully formatted.（输出尾部） 并无 error 提示，做下一步

```
2018-05-19 16:06:13,024 INFO common.Storage: Storage directory /usr/local/hadoop/tmp/dfs/name ha
s been successfully formatted.
2018-05-19 16:06:13,036 INFO namenode.FSImageFormatProtobuf: Saving image file /usr/local/hadoop
/tmp/dfs/name/current/fsimage.ckpt_0000000000000000000 using no compression
2018-05-19 16:06:13,229 INFO namenode.FSImageFormatProtobuf: Image file /usr/local/hadoop/tmp/df
s/name/current/fsimage.ckpt_0000000000000000000 of size 391 bytes saved in 0 seconds .
2018-05-19 16:06:13,240 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with t
xid >= 0
2018-05-19 16:06:13,252 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at localhost/127.0.0.1
************************************************************/
[hadoop@localhost hadoop]$ ^C
```

接着开启 NaneNode 和 DataNode 守护进程：

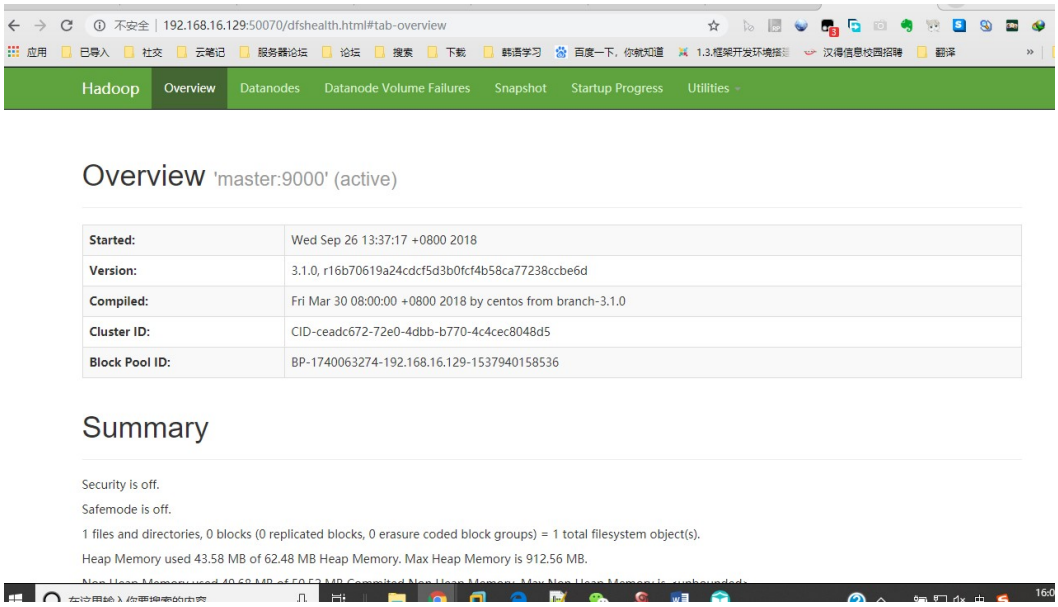
[hadoop@localhost hadoop]$ ./sbin/start-dfs.sh

使用 jps 查看进程：

```
[hadoop@master hadoop]$ jps
20052 Jps
17181 NameNode
17293 DataNode
17502 SecondaryNameNode
[hadoop@master hadoop]$
```

1.4.3 运行查看

在操作完以上步骤后 hadoop 伪分布式搭建即搭建完成，可在外部浏览器使用 IP：50070 查看，（首先要保证虚拟机与物理机内外 ping 通）打开如下页面即为成功搭建



参考

本文档参考许多网上博客，先后不分等级，帮助同等要重，

参考文档：

https://blog.csdn.net/c1481118216/article/details/73326049 https://blog.csdn.net/hliq5399/article/details/78193113

致谢！