

CentOS 6.8 + Hadoop2.7.3

集群环境搭建

技术文档

(版本 : V1.0.0)

版本	日期	说明	作者
V1.0.0	2016/11/8	创建	Simon Hoo

目 录

一 .	环境准备	- 3 -
1.1	服务器及配置	- 3 -
1.2	软件准备	- 4 -
二 .	系统环境配置	- 4 -
2.1	系统设置	- 4 -
2.1.1	设置 hostname	- 4 -
2.1.2	创建用户	- 6 -
2.1.3	安装 JDK	- 6 -
2.1.4	设置 SSH 免密码登录	- 7 -
三 .	安装 Master (Name Node)节点	- 8 -
3.1	解压安装包	- 8 -
3.1.1	解压 hadoop 安装包	- 8 -
3.2	设置环境变量	- 8 -
3.3	修改\$HADOOP_HOME/etc/hadoop/*-env.sh	- 8 -
3.4	修改\$HADOOP_HOME /etc/hadoop/core-site.xml	- 9 -
3.5	修改\$HADOOP_HOME /etc/hadoop/hdfs-site.xml	- 9 -
3.6	修改\$HADOOP_HOME /etc/hadoop/yarn-site.xml	- 10 -
3.7	修改\$HADOOP_HOME/etc/hadoop/mapred-site.xml	- 11 -
3.8	修改\$HADOOP_HOME/etc/hadoop/slaves	- 11 -
四 .	安装 Slave (DataNode)节点	- 11 -
4.1	复制 Master 上配置好的 hadoop 到 Slave 节点上	- 11 -
五 .	格式化 NameNode	- 12 -
六 .	启动 Hadoop	- 12 -
6.1	Master 上启动	- 12 -
6.2	JPS 检验是否成功启动	- 13 -
6.3	管理界面	- 13 -
七 .	测试项目（提交一个 MapReduce 任务）	- 15 -
八 .	应用程序接入	- 15 -
5.1	创建 JAVA 程序	- 15 -

一. 环境准备

1.1 服务器及配置

服务器	IP 地址	安装软件	备注
Hadoop Master (Name Node)	10.50.130.55	Hadoop 2.7.3 JDK 1.7	CentOS 6.8 64 位 CPU: 2x2CPUs RAM: 8GB Disk: 50GB
Hadoop Slave 1 (Data Node)	10.50.130.56	Hadoop 2.7.3 JDK 1.7	CentOS 6.8 64 位 CPU: 2x2CPUs RAM: 8GB Disk: 50GB
Hadoop Slave 2 (Data Node)	10.50.130.57	Hadoop 2.7.3 JDK 1.7	CentOS 6.8 64 位 CPU: 2x2CPUs RAM: 8GB Disk: 50GB
Hadoop Slave 3 (Data Node)	10.50.130.58	Hadoop 2.7.3 JDK 1.7	CentOS 6.8 64 位 CPU: 2x2CPUs RAM: 8GB Disk: 50GB

```

simonhoo — root@hadoop-master:~/software — ssh root@10.50.130.55 — 80...
[root@hadoop-master software]# uname -a
Linux hadoop-master.beyondsoft-dev.com 2.6.32-642.6.2.el6.x86_64 #1 SMP Wed Oct 26 06:52:09 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
[root@hadoop-master software]# cat /etc/issue
CentOS release 6.8 (Final)
Kernel \r on an \m

[root@hadoop-master software]#

simonhoo — root@hadoop-slave1:~/software — ssh root@10.50.130.56 — 80...
[root@hadoop-slave1 software]# uname -a
Linux hadoop-slave1.beyondsoft-dev.com 2.6.32-642.6.2.el6.x86_64 #1 SMP Wed Oct 26 06:52:09 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
[root@hadoop-slave1 software]# cat /etc/issue
CentOS release 6.8 (Final)
Kernel \r on an \m

[root@hadoop-slave1 software]#

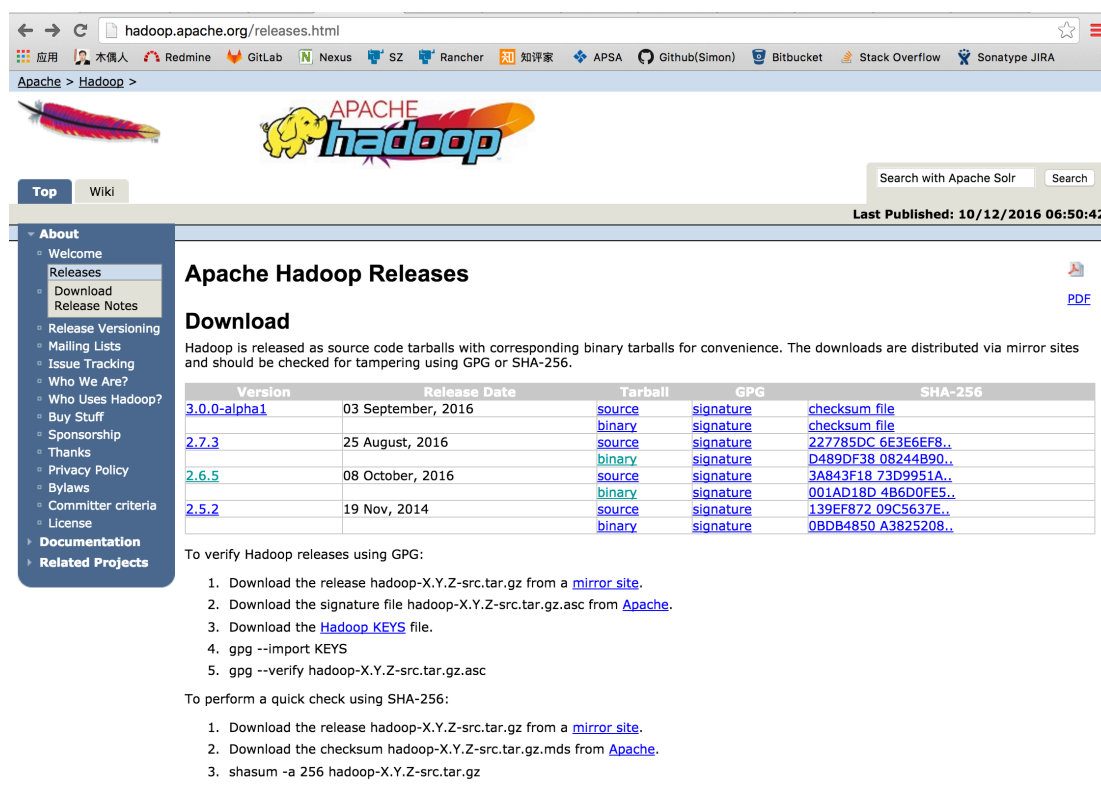
simonhoo — root@hadoop-slave2:~/software — ssh root@10.50.130.57 — 80...
[root@hadoop-slave2 software]# uname -a
Linux hadoop-slave2.beyondsoft-dev.com 2.6.32-642.6.2.el6.x86_64 #1 SMP Wed Oct 26 06:52:09 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
[root@hadoop-slave2 software]# cat /etc/issue
CentOS release 6.8 (Final)
Kernel \r on an \m

[root@hadoop-slave2 software]#

simonhoo — root@hadoop-slave3:~/software — ssh root@10.50.130.58 — 80...
[root@hadoop-slave3 software]# uname -a
Linux hadoop-slave3.beyondsoft-dev.com 2.6.32-642.6.2.el6.x86_64 #1 SMP Wed Oct 26 06:52:09 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
[root@hadoop-slave3 software]# cat /etc/issue
CentOS release 6.8 (Final)
Kernel \r on an \m

[root@hadoop-slave3 software]#
  
```

1.2 软件准备



Apache Hadoop Releases

Download

Hadoop is released as source code tarballs with corresponding binary tarballs for convenience. The downloads are distributed via mirror sites and should be checked for tampering using GPG or SHA-256.

Version	Release Date	Tarball	GPG	SHA-256
3.0.0-alpha1	03 September, 2016	source	signature	checksum file
2.7.3	25 August, 2016	binary	signature	checksum file
2.6.5	08 October, 2016	source	signature	227785DC 6E3E6EF8..
2.5.2	19 Nov, 2014	binary	signature	D489DF38 08244B90..
		source	signature	3A843F18 73D9951A..
		binary	signature	001AD18D 4B6D0FE5..
		source	signature	139EF872 09C5637E..
		binary	signature	0BDB4850 A3825208..

To verify Hadoop releases using GPG:

1. Download the release `hadoop-X.Y.Z-src.tar.gz` from a [mirror site](#).
2. Download the signature file `hadoop-X.Y.Z-src.tar.gz.asc` from [Apache](#).
3. Download the [Hadoop KEYS](#) file.
4. `gpg --import KEYS`
5. `gpg --verify hadoop-X.Y.Z-src.tar.gz.asc`

To perform a quick check using SHA-256:

1. Download the release `hadoop-X.Y.Z-src.tar.gz` from a [mirror site](#).
2. Download the checksum `hadoop-X.Y.Z-src.tar.gz.mds` from [Apache](#).
3. `shasum -a 256 hadoop-X.Y.Z-src.tar.gz`

注：JDK 自行在 ORACLE 官网下载。

二． 系统环境配置

2.1 系统设置

2.1.1 设置 hostname

Master 节点：

```
[root@localhost ~]# vi /etc/hosts
10.50.130.55 hadoop-master
10.50.130.56 hadoop-slave1
10.50.130.57 hadoop-slave2
10.50.130.58 hadoop-slave3
```

```
[root@localhost ~]# vi /etc/sysconfig/network
```

```
HOSTNAME = hadoop-master
```

```
[root@localhost ~]# vi /etc/selinux/config  
SELINUX=disabled
```

```
[root@localhost ~]# reboot
```

Slave1 节点 :

```
[root@localhost ~]# vi /etc/hosts  
10.50.130.55 hadoop-master  
10.50.130.56 hadoop-slave1  
10.50.130.57 hadoop-slave2  
10.50.130.58 hadoop-slave3
```

```
[root@localhost ~]# vi /etc/sysconfig/network  
HOSTNAME = hadoop-slave1
```

```
[root@localhost ~]# vi /etc/selinux/config  
SELINUX=disabled
```

```
[root@localhost ~]# reboot
```

Slave2 节点 :

```
[root@localhost ~]# vi /etc/hosts  
10.50.130.55 hadoop-master  
10.50.130.56 hadoop-slave1  
10.50.130.57 hadoop-slave2  
10.50.130.58 hadoop-slave3
```

```
[root@localhost ~]# vi /etc/sysconfig/network  
HOSTNAME = hadoop-slave2
```

```
[root@localhost ~]# vi /etc/selinux/config  
SELINUX=disabled
```

```
[root@localhost ~]# reboot
```

Slave3 节点 :

```
[root@localhost ~]# vi /etc/hosts
```

```
10.50.130.55 hadoop-master
10.50.130.56 hadoop-slave1
10.50.130.57 hadoop-slave2
10.50.130.58 hadoop-slave3
```

```
[root@localhost ~]# vi /etc/sysconfig/network
HOSTNAME = hadoop-slave3
```

```
[root@localhost ~]# vi /etc/selinux/config
SELINUX=disabled
```

```
[root@localhost ~]# reboot
```

2.1.2 创建用户

```
[root@hadoop-master ~]# useradd hadoop
[root@hadoop-master ~]# passwd Hadoop
```

2.1.3 安装 JDK

```
[root@hadoop-master ~]# mkdir -p /usr/local/java
[root@hadoop-master ~]# cd /root/software
[root@hadoop-master software]# tar -xzf jdk-7u79-linux-x64.tar.gz -C /usr/local/java
```

设置 JAVA_HOME:

```
[root@hadoop-master ~]# vi /etc/profile
export JAVA_HOME=/usr/local/java/jdk1.7.0_79
export CLASSPATH=.:$JAVA_HOME/jre/lib/rt.jar:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
export PATH=$PATH:$JAVA_HOME/bin
```

```
[root@hadoop-master ~]# source /etc/profile
[root@hadoop-master ~]# java -version
[root@hadoop-master ~]# chown -R hadoop.hadoop /usr/local/java/
```

2.1.4 设置 SSH 免密码登录

Master 节点：

```
[root@hadoop-master ~]# su hadoop
[hadoop@hadoop-master ~]$ ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
[hadoop@hadoop-master ~]$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
[hadoop@hadoop-master ~]$ chmod 600 ~/.ssh/authorized_keys
```

复制到 Slave 节点：

```
[hadoop@hadoop-master ~]$ scp ~/.ssh/id_rsa.pub hadoop@10.50.130.56:/tmp/authorized_keys
[hadoop@hadoop-master ~]$ scp ~/.ssh/id_rsa.pub hadoop@10.50.130.57:/tmp/authorized\_keys
[hadoop@hadoop-master ~]$ scp ~/.ssh/id_rsa.pub hadoop@10.50.130.58:/tmp/authorized\_keys
```

在 Slave 节点追加 /tmp/authorized_keys 到 ~/.ssh/authorized_keys：

```
[hadoop@hadoop-slave1 ~]$ cat /tmp/authorized_keys >> ~/.ssh/authorized_keys
[hadoop@hadoop-slave1 ~]$ chmod 700 ~/.ssh/
[hadoop@hadoop-slave1 ~]$ chmod 600 ~/.ssh/authorized_keys
```

```
[hadoop@hadoop-slave2 ~]$ cat /tmp/authorized_keys >> ~/.ssh/authorized_keys
[hadoop@hadoop-slave2 ~]$ chmod 700 ~/.ssh/
[hadoop@hadoop-slave2 ~]$ chmod 600 ~/.ssh/authorized_keys
```

```
[hadoop@hadoop-slave3 ~]$ cat /tmp/authorized_keys >> ~/.ssh/authorized_keys
[hadoop@hadoop-slave3 ~]$ chmod 700 ~/.ssh/
[hadoop@hadoop-slave3 ~]$ chmod 600 ~/.ssh/authorized_keys
```

在 Master 节点测试 SSH 免登录到 Slave 节点：

```
[hadoop@hadoop-master ~]$ ssh hadoop-master
[hadoop@hadoop-master ~]$ ssh hadoop-slave1
[hadoop@hadoop-master ~]$ ssh hadoop-slave2
[hadoop@hadoop-master ~]$ ssh hadoop-slave3
```

当然值得注意的是：首次登陆是需要确认的，hadoop-slave1(1~3)结点首次连接时需要，“YES”确认连接，这意味着 hadoop-master 结点连接 hadoop-slave1(1~3)结点时需要人工询问，无法自动连接，输入 yes 后成功接入，紧接着注销退出至 hadoop-master 结点。要实现 ssh 免密码连接至其它结点，还差一步，只需要再执行一遍 ssh hadoop-slave(1~3)，如果没有要求你输入“yes”，就算成功了。

三 . 安装 Master (Name Node)节点

3.1 解压安装包

3.1.1 解压 hadoop 安装包

```
[root@hadoop-master ~]# cd /software
[root@hadoop-master ~]# tar -xzvf hadoop-2.7.3.tar.gz
[root@hadoop-master ~]# cp -r hadoop-2.7.3 /usr/local/hadoop
[root@hadoop-master ~]# chown -R hadoop.hadoop /usr/local/hadoop/
```

3.2 设置环境变量

```
[root@hadoop-master ~]# vi /etc/profile
# Setting HADOOP env
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

[root@hadoop-master ~]# source /etc/profile
```

3.3 修改\$HADOOP_HOME/etc/hadoop/*-env.sh

```
[root@hadoop-master ~]# cd $HADOOP_HOME/etc/hadoop
[root@hadoop-master hadoop]# vi hadoop-env.sh
# The java implementation to use.
#export JAVA_HOME=${JAVA_HOME}
export JAVA_HOME=/usr/local/java/jdk1.7.0_79

[root@hadoop-master hadoop]# vi mapred-env.sh
export JAVA_HOME=/usr/local/java/jdk1.7.0_79
```



```
[root@hadoop-master hadoop]# vi yarn-env.sh
export JAVA_HOME=/usr/local/java/jdk1.7.0_79
```

3.4 修改\$HADOOP_HOME /etc/hadoop/core-site.xml

```
[root@hadoop-master ~]# cd $HADOOP_HOME/etc/hadoop
[root@hadoop-master ~]# vi core-site.xml
<!-- 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://hadoop-master:9000</value>
    </property>
</configuration>
```

3.5 修改\$HADOOP_HOME /etc/hadoop/hdfs-site.xml

```
[root@hadoop-master ~]# cd $HADOOP_HOME/etc/hadoop
[root@hadoop-master ~]# vi hdfs-site.xml
<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
        <name>dfs.namenode.secondary.http-address</name>
        <value>hadoop-master:50090</value>
    </property>
    <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>
```

3.6 修改\$HADOOP_HOME /etc/hadoop/yarn-site.xml

```
[root@hadoop-master ~]# cd $HADOOP_HOME/etc/hadoop
```

```
[root@hadoop-master ~]# vi yarn-site.xml
```

```
<configuration>
    <!-- Site specific YARN configuration properties -->
    <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
    </property>
    <property>
        <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
        <value>org.apache.hadoop.mapred.ShuffleHandler</value>
    </property>
    <property>
        <name>yarn.resourcemanager.address</name>
        <value>hadoop-master:8032</value>
    </property>
    <property>
        <name>yarn.resourcemanager.scheduler.address</name>
        <value>hadoop-master:8030</value>
    </property>
    <property>
        <name>yarn.resourcemanager.resource-tracker.address</name>
        <value>hadoop-master:8031</value>
    </property>
    <property>
        <name>yarn.resourcemanager.admin.address</name>
        <value>hadoop-master:8033</value>
    </property>
    <property>
        <name>yarn.resourcemanager.webapp.address</name>
        <value>hadoop-master:8088</value>
    </property>
</configuration>
```

3.7 修改\$HADOOP_HOME/etc/hadoop/mapred-site.xml

```
[root@hadoop-master ~]# cd $HADOOP_HOME/etc/hadoop
[root@hadoop-master ~]# vi mapred-site.xml
<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
    <property>
        <name>mapreduce.jobhistory.address</name>
        <value>hadoop-master:10020</value>
    </property>
    <property>
        <name>mapreduce.jobhistory.webapp.address</name>
        <value>hadoop-master:19888</value>
    </property>
</configuration>
```

3.8 修改\$HADOOP_HOME/etc/hadoop/slaves

```
[root@hadoop-master ~]# cd $HADOOP_HOME/etc/hadoop
[root@hadoop-master ~]# vi slaves
hadoop-slave1
hadoop-slave2
hadoop-slave3
```

四 . 安装 Slave (DataNode)节点

4.1 复制 Master 上配置好的 hadoop 到 Slave 节点上

```
[hadoop@hadoop-master ~]$ cd /usr/local
[hadoop@hadoop-master local]$ scp -r hadoop root@hadoop-slave1:/usr/local
[hadoop@hadoop-master local]$ scp -r hadoop root@hadoop-slave2:/usr/local
[hadoop@hadoop-master local]$ scp -r hadoop root@hadoop-slave3:/usr/local\
```

修改 Slave 节点上的文件归属：

```
[root@hadoop-slave1 ~]# chown -R hadoop.hadoop /usr/local/hadoop/
```

```
[root@hadoop-slave2 ~]# chown -R hadoop.hadoop /usr/local/hadoop/
```

```
[root@hadoop-slave3 ~]# chown -R hadoop.hadoop /usr/local/hadoop/
```

五 . 格式化 NameNode

hadoop 登录到 hadoop-master:

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

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

```
simonhoo — hadoop@hadoop-master:/usr/local/hadoop — ssh root@10.50.130.55 — 185x44
16/11/09 17:28:58 INFO blockmanagement.BlockManager: maxReplicationStreams = 2
16/11/09 17:28:58 INFO blockmanagement.BlockManager: replicationRecheckInterval = 3000
16/11/09 17:28:58 INFO blockmanagement.BlockManager: encryptDataTransfer = false
16/11/09 17:28:58 INFO blockmanagement.BlockManager: maxNumBlocksToLog = 1000
16/11/09 17:28:58 INFO namenode.FSNamesystem: fsOwner = hadoop (auth:SIMPLE)
16/11/09 17:28:58 INFO namenode.FSNamesystem: supergroup = supergroup
16/11/09 17:28:58 INFO namenode.FSNamesystem: isPermissionEnabled = true
16/11/09 17:28:58 INFO namenode.FSNamesystem: HA Enabled: false
16/11/09 17:28:58 INFO namenode.FSNamesystem: Append Enabled: true
16/11/09 17:28:59 INFO util.GSet: Computing capacity for map INodeMap
16/11/09 17:28:59 INFO util.GSet: VM type = 64-bit
16/11/09 17:28:59 INFO util.GSet: 1.0% max memory 889 MB = 8.9 MB
16/11/09 17:28:59 INFO util.GSet: capacity = 2^20 = 1048576 entries
16/11/09 17:28:59 INFO namenode.FSDirectory: ACLs enabled? false
16/11/09 17:28:59 INFO namenode.FSDirectory: XAttrs enabled? true
16/11/09 17:28:59 INFO namenode.FSDirectory: Maximum size of an xattr: 16384
16/11/09 17:28:59 INFO namenode.NameNode: Caching file names occurring more than 10 times
16/11/09 17:28:59 INFO util.GSet: Computing capacity for map cachedBlocks
16/11/09 17:28:59 INFO util.GSet: VM type = 64-bit
16/11/09 17:28:59 INFO util.GSet: 0.25% max memory 889 MB = 2.2 MB
16/11/09 17:28:59 INFO util.GSet: capacity = 2^18 = 262144 entries
16/11/09 17:28:59 INFO namenode.FSNamesystem: dfs.namenode.safemode.threshold-pct = 0.9990000128746033
16/11/09 17:28:59 INFO namenode.FSNamesystem: dfs.namenode.safemode.min.datanodes = 0
16/11/09 17:28:59 INFO namenode.FSNamesystem: dfs.namenode.safemode.extension = 30000
16/11/09 17:28:59 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.window.num.buckets = 10
16/11/09 17:28:59 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.num.users = 10
16/11/09 17:28:59 INFO metrics.TopMetrics: NNTop conf: dfs.namenode.top.windows.minutes = 1,5,25
16/11/09 17:28:59 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
16/11/09 17:28:59 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total heap and retry cache entry expiry time is 600000 millis
16/11/09 17:28:59 INFO util.GSet: Computing capacity for map NameNodeRetryCache
16/11/09 17:28:59 INFO util.GSet: VM type = 64-bit
16/11/09 17:28:59 INFO util.GSet: 0.029999999329447746% max memory 889 MB = 273.1 KB
16/11/09 17:28:59 INFO util.GSet: capacity = 2^15 = 32768 entries
16/11/09 17:28:59 INFO namenode.FSImage: Allocated new BlockPoolId: BP-99413354-10.50.130.55-1478683739234
16/11/09 17:28:59 INFO common.Storage: Storage directory /usr/local/hadoop/tmp/dfs/name has been successfully formatted.
16/11/09 17:28:59 INFO namenode.FSImageFormatProtobuf: Saving image file /usr/local/hadoop/tmp/dfs/name/current/fsimage.ckpt_000000000000000000 using no compression
16/11/09 17:28:59 INFO namenode.FSImageFormatProtobuf: Image file /usr/local/hadoop/tmp/dfs/name/current/fsimage.ckpt_000000000000000000 of size 353 bytes saved in 0 seconds.
16/11/09 17:28:59 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
16/11/09 17:28:59 INFO util.ExitUtil: Exiting with status 0
16/11/09 17:28:59 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at hadoop-master/10.50.130.55
*****/
[hadoop@hadoop-master hadoop]$
```

注意：在你重新格式化分布式文件系统之前，需要将文件系统中的数据先清除，否则，datanode 将创建不成功，这一点很重要。

六 . 启动 Hadoop

6.1 Master 上启动

hadoop 登录到 hadoop-master:

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

```
[hadoop@hadoop-master hadoop]$ ./sbin/start-all.sh
```

6.2 JPS 检验是否成功启动

hadoop-master:

```
[hadoop@hadoop-master ~]$ jps  
1866 SecondaryNameNode  
1671 NameNode  
2023 ResourceManager  
2290 Jps
```

hadoop-slave1:

```
[hadoop@hadoop-slave1 ~]$ jps  
1611 Jps  
1372 DataNode  
1474 NodeManager
```

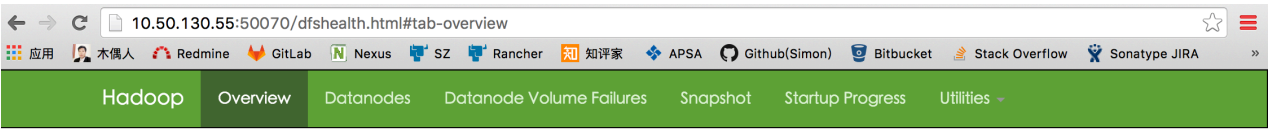
hadoop-slave2:

```
[hadoop@hadoop-slave2 ~]$ jps  
1356 DataNode  
1458 NodeManager  
1595 Jps
```

hadoop-slave3:

```
[hadoop@hadoop-slave3 ~]$ jps  
1592 Jps  
1353 DataNode  
1455 NodeManager
```

6.3 管理界面



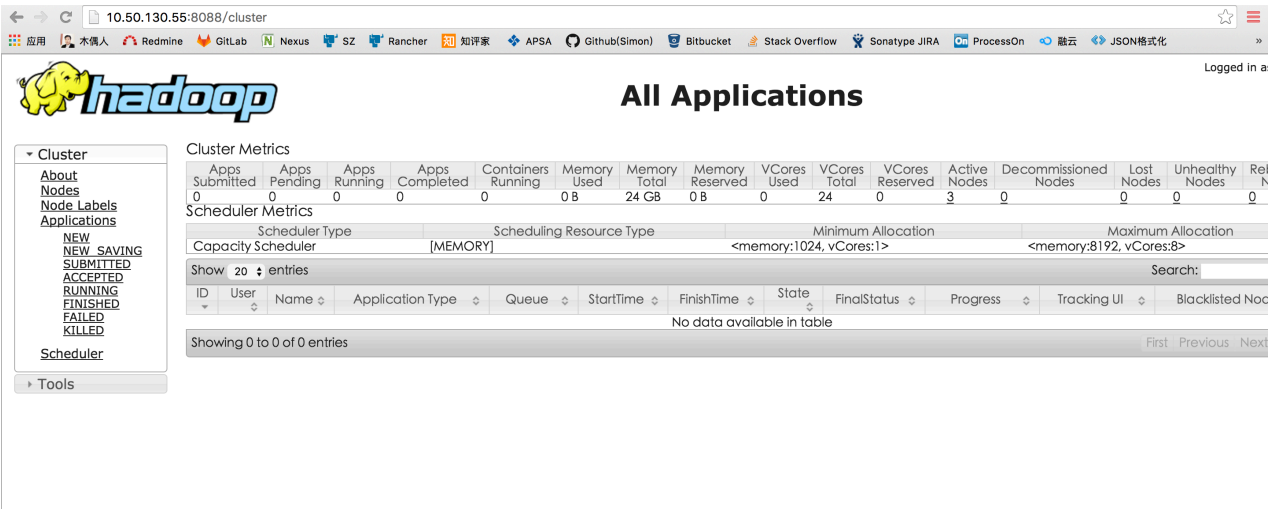
Overview 'hadoop-master:9000' (active)

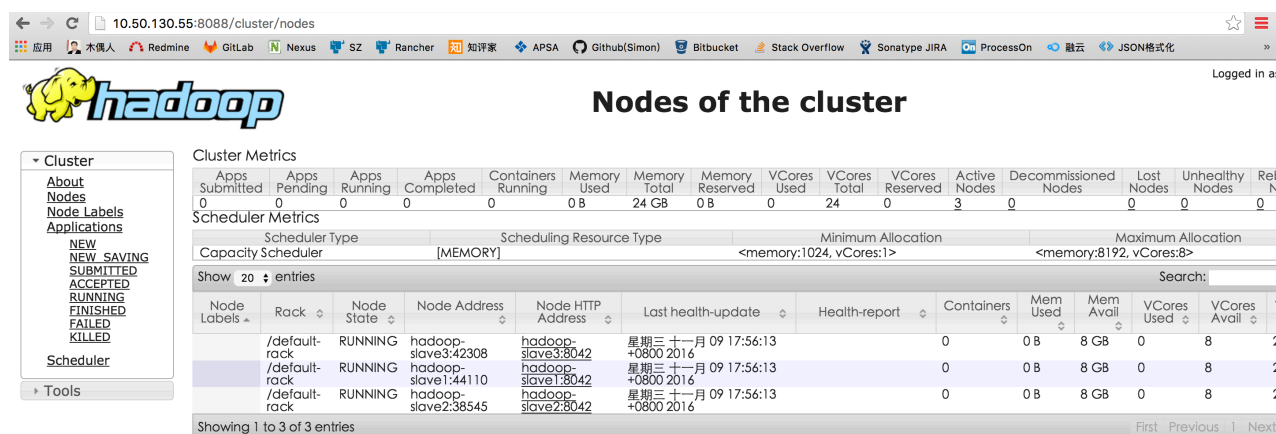
Started:	Wed Nov 09 17:32:39 CST 2016
Version:	2.7.3, rbaa91f7c6bc9cb92be5982de4719c1c8af91ccff
Compiled:	2016-08-18T01:41Z by root from branch-2.7.3
Cluster ID:	CID-bf3810fb-56af-4125-8634-c8acc8536a90
Block Pool ID:	BP-99413354-10.50.130.55-1478683739234

Summary

Security is off.
Safemode is off.
3 files and directories, 1 blocks = 4 total filesystem object(s).
Heap Memory used 69.16 MB of 223 MB Heap Memory. Max Heap Memory is 889 MB.
Non Heap Memory used 40.04 MB of 40.94 MB Committed Non Heap Memory. Max Non Heap Memory is 130 MB.

Configured Capacity:	122.7 GB
DFS Used:	180 KB (0%)
Non DFS Used:	12.66 GB





Nodes of the cluster

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rel
0	0	0	0	0	0 B	24 GB	0 B	0	24	0	3	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>	<memory:8192, vCores:8>

Show 20 entries

Node Labels	Rack	Node State	Node Address	Node HTTP Address	Last health-update	Health-report	Containers	Mem Used	Mem Avail	VCores Used	VCores Avail
/default-rack		RUNNING	hadoop-slave3:42308	hadoop-slave3:8042	星期三十一月09 17:56:13 +0800 2016		0	0 B	8 GB	0	8
/default-rack		RUNNING	hadoop-slave1:44110	hadoop-slave1:8042	星期三十一月09 17:56:13 +0800 2016		0	0 B	8 GB	0	8
/default-rack		RUNNING	hadoop-slave2:38545	hadoop-slave2:8042	星期三十一月09 17:56:13 +0800 2016		0	0 B	8 GB	0	8

Showing 1 to 3 of 3 entries

七. 测试项目（提交一个 MapReduce 任务）

创建测试项目：

```
[hadoop@hadoop-master ~]$ cd /usr/local/Hadoop
```

```
[hadoop@hadoop-master hadoop]$ ./bin/hdfs dfs -mkdir /test-project
```

复制一份文件到分布式文件系统中：

```
[hadoop@hadoop-master hadoop]$ ./bin/hdfs dfs -copyFromLocal ./LICENSE.txt /test-project
```

查看在分布式文件系统中是否复制成功：

```
[hadoop@hadoop-master hadoop]$ ./bin/hdfs dfs -ls /test-project
```

```
16/11/09 17:49:51 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
```

```
Found 1 items
```

```
-rw-r--r--    1 hadoop supergroup      84854 2016-11-09 17:48 /test-project/LICENSE.txt
```

八. 应用程序接入

5.1 创建 JAVA 程序