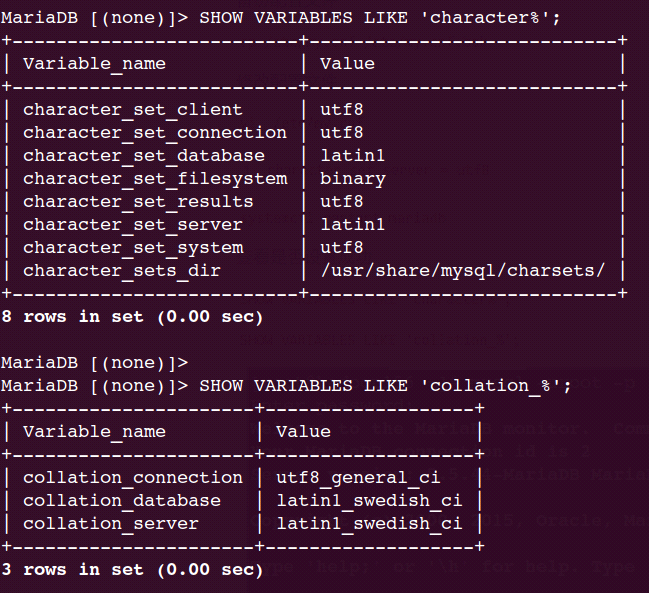
设置字符编码为utf-8。

首先，查看默认编码：

|  |
| --- |
| mysql -u root -p  Enter password:  SHOW VARIABLES LIKE 'character%';  SHOW VARIABLES LIKE 'collation\_%'; |

执行结果如下：



图：执行结果图

如果显示编码为 latin1，下面进行修改：

备份配置文件：

|  |
| --- |
| cp /etc/my.cnf /etc/my.cnf.bak |

修改配置文件：

vi /etc/my.cnf #在[mysqld]下加入下面内容

|  |
| --- |
| character\_set\_server=utf8 |

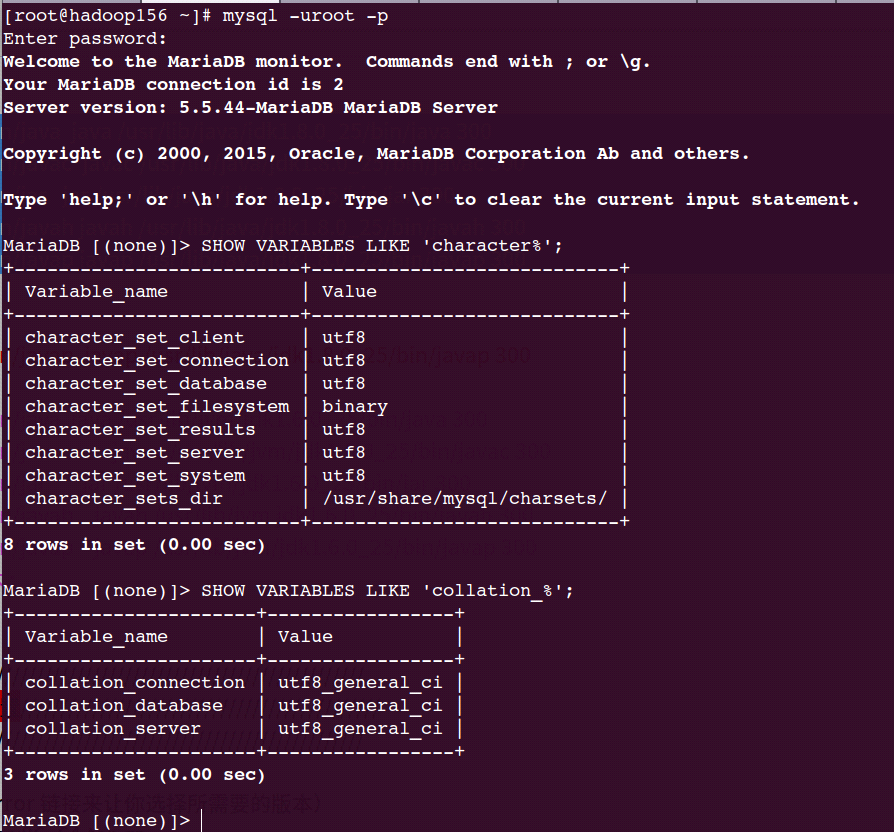
重启数据库：

|  |
| --- |
| systemctl restart mariadb |

重新登录，查看设置是否成功：

|  |
| --- |
| SHOW VARIABLES LIKE 'character%';  SHOW VARIABLES LIKE 'collation\_%'; |

执行结果如下：



图：执行结果图

* **3.14 安装MySQL驱动**

所有主机都需安装同一版本的mysql-connector驱动。

|  |
| --- |
| yum -y install mysql-connector-java |

安装后，mysql-connector-java.jar的路径为 ：/usr/share/java/mysql-connector-java.jar 。

* **3.15 安装其他依赖**

所有主机安装以下依赖：

安装psmisc，如果未安装psmisc，后面启动CM agent和server会提示找不到pstree；

安装perl；

安装nfs-utils portmap；

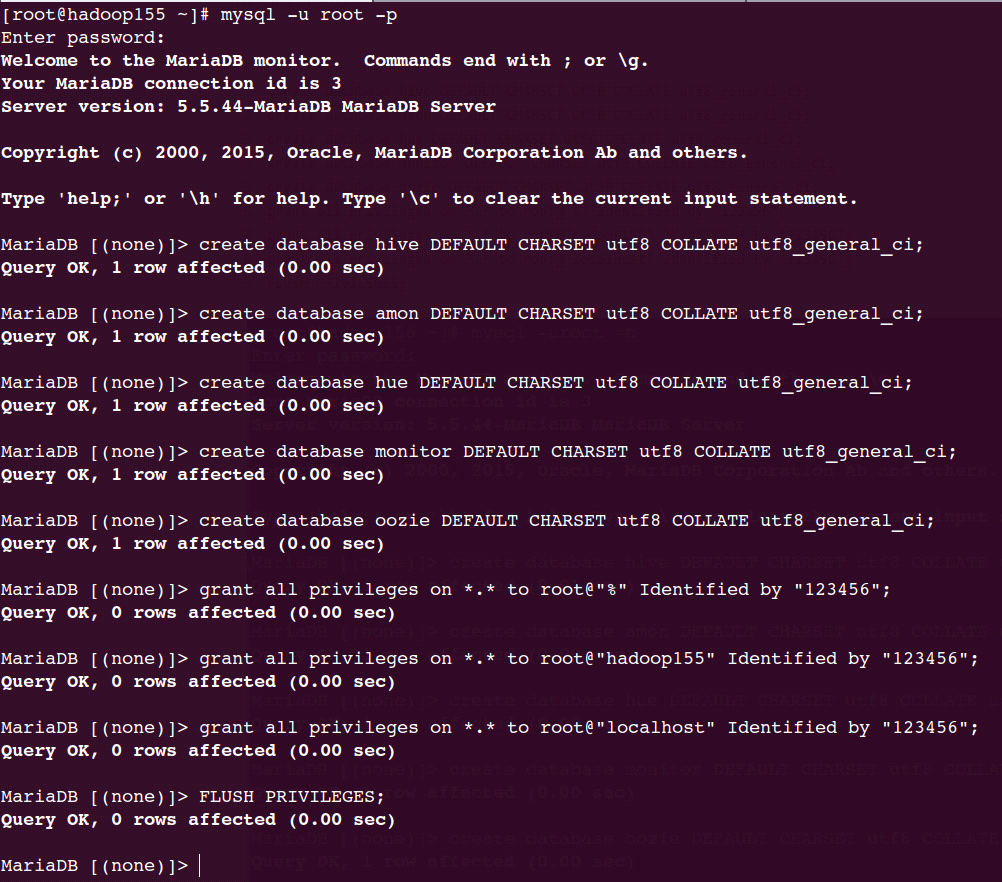
|  |
| --- |
| yum -y install psmisc  yum –y install perl  yum -y install nfs-utils portmap  systemctl start rpcbind  systemctl enable rpcbind |

* **创建数据库和用户**

创建hive和oozie的数据库。

登录数据库执行下面的SQL创建数据库。

|  |
| --- |
| create database hive DEFAULT CHARSET utf8 COLLATE utf8\_general\_ci;  create database amon DEFAULT CHARSET utf8 COLLATE utf8\_general\_ci;  create database hue DEFAULT CHARSET utf8 COLLATE utf8\_general\_ci;  create database monitor DEFAULT CHARSET utf8 COLLATE utf8\_general\_ci;  create database oozie DEFAULT CHARSET utf8 COLLATE utf8\_general\_ci;  CREATE USER 'hive'@'localhost' IDENTIFIED BY 'hive';  GRANT ALL PRIVILEGES ON hive.\* TO 'hive'@'localhost';  CREATE USER 'hive'@'%' IDENTIFIED BY 'hive';  GRANT ALL PRIVILEGES ON hive.\* TO 'hive'@'%';  CREATE USER 'hive'@'hadoop155'IDENTIFIED BY 'hive';  GRANT ALL PRIVILEGES ON hive.\* TO 'hive'@'hadoop155';  CREATE USER 'oozie'@'localhost' IDENTIFIED BY 'oozie';  GRANT ALL PRIVILEGES ON oozie.\* TO 'oozie'@'localhost';  CREATE USER 'oozie'@'%' IDENTIFIED BY 'oozie';  GRANT ALL PRIVILEGES ON oozie.\* TO 'oozie'@'%';  CREATE USER 'oozie'@'hadoop155'IDENTIFIED BY 'oozie';  GRANT ALL PRIVILEGES ON oozie.\* TO 'oozie'@'hadoop155';  CREATE USER 'monitor'@'localhost' IDENTIFIED BY 'monitor';  GRANT ALL PRIVILEGES ON monitor.\* TO 'monitor'@'localhost';  CREATE USER 'monitor'@'%' IDENTIFIED BY 'monitor';  GRANT ALL PRIVILEGES ON monitor.\* TO 'monitor'@'%';  CREATE USER 'monitor'@'hadoop155'IDENTIFIED BY 'monitor';  GRANT ALL PRIVILEGES ON monitor.\* TO 'monitor'@'hadoop155';  FLUSH PRIVILEGES; |



图：执行结果图

以上是集群的基础配置，接下来开始安装CM和CDH。

安装CDH的过程比较简单，大致~~思路~~步骤如下：

1、先在在所有主机安装 CM agent,并启动；

2、然后，选择一台主机（hadoop155）初始化数据库，并安装CM server；

3、启动CM Server，通过浏览器访问CM server web页面地址（[http://hadoop155:7180](http://hadoop155:7180/)）,选择相应的服务组件进行安装。

* **安装CM和CDH**

针对不同的环境，采用不同的安装方式，本文采用Parcels安装方式。

1、如果用户的集群能够访问到Cloudera的网站，直接可以进行安装。

2、如果用户的集群不能够访问公网，则只需要在集群内部建立一个软件仓库，然后再进行自动化的集群部署。安装过程如下:

* 安装环境准备，下载相关安装文件，安装JDK、yum、数据库等基本软件。
* 选择一台主机，安装CM，用户只需要启动安装脚本即可，通常情况下能够快速完成。
* CM是一个Web应用，提供了基于浏览器的界面，用户可以通过浏览器可视化的进行CDH的安装部署。
* 通过CM界面，添加其他需要安装的节点，选择要安装的Hadoop组件，以及每个节点承担的角色，选择安装，CM会自动的将需要安装的软件分发到对应的节点，并完成安装。
* 同时CM会提供一个默认的CDH配置，并且提供了配置的管理界面，供用户在后续进行定制化的配置。
* 在安装过程中，如果任意一台机器安装出现问题，CM都会进行汇报，并提供日志，供用户分析，解决问题。
* 当所有节点的软件都安装完成之后，CM会启动所有的服务。
* **5.1 安装CM**

CM是本文主要的介绍对象和安装内容。它细分为免费的Express版本和功能完全并提供众多增值服务的收费Enterprise版本。

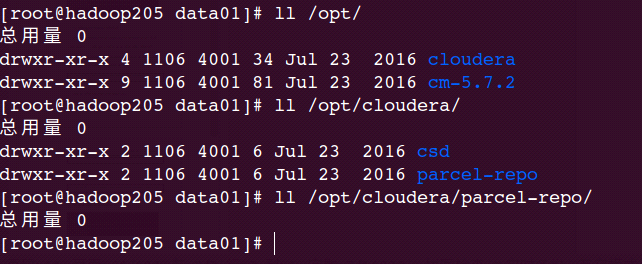
CM负责统一配置管理CDH集群，本文是围绕CM的安装部署来展开，然后再通过CM安装CDH及其其它相关组件。

1、上传CM安装包并解压。

所有主机上传CM相关包cloudera-manager-centos7-cm5.7.2\_x86\_64.tar.gz到硬盘/data01上，并解压到已创建的目录/opt/下。

|  |
| --- |
| tar -zxvf /home/cloudera-manager-centos7-cm5.15.1\_x86\_64.tar.gz -C /opt/ |

解压正常，目录结构如下：



图：执行结果图

2、所有节点创建cloudera-scm用户授予sudo权限。

|  |
| --- |
| useradd --system --home=/opt/cloudera-manager/cm-5.15.1/run/cloudera-scm-server --no-create-home --shell=/bin/false --comment "Cloudera SCM User" cloudera-scm  usermod -aG wheel cloudera-scm |

3、设置agent向server发送心跳。

配置所有主机的cloudera-manger-agent指向CM server主机（hadoop155），实现agent向CM server发送心跳。操作如下：

首先，查看/opt/cloudera-manager/cm-5.7.2/etc/cloudera-scm-agent/config.ini文件内容是否为localhost。

|  |
| --- |
| cat /opt/cm-5.15.1/etc/cloudera-scm-agent/config.ini|grep server\_host |

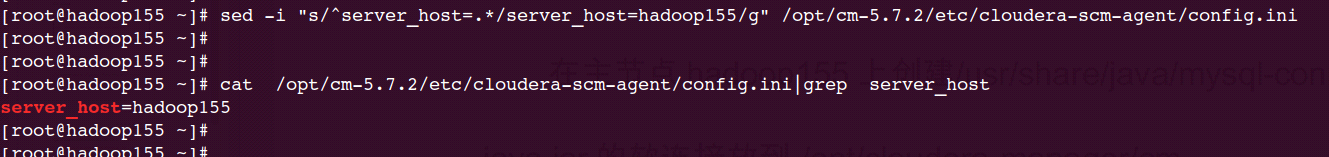
如果是，修改localhost改为hadoop155。

|  |
| --- |
| sed -i "s/^server\_host=.\*/server\_host=hadoop1/g" /opt/cm-5.15.1/etc/cloudera-scm-agent/config.ini |

最后，确认是否修改成功。

|  |
| --- |
| cat /opt/cm-5.15.1/etc/cloudera-scm-agent/config.ini|grep server\_host |

执行结果如图所示说明修改成功。

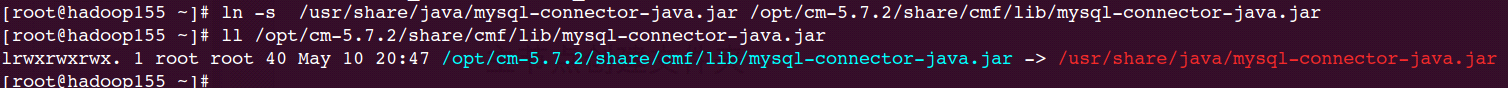


图：执行结果图

4、设置数据库驱动（创建软连接）。

在主节点hadoop155上创建/usr/share/java/mysql-connector-java.jar的软连接放到 /opt/cloudera-manager/cm-5.7.2/share/cmf/lib/。

|  |
| --- |
| ln -s /usr/share/java/mysql-connector-java.jar /opt/cm-5.15.1/share/cmf/lib/mysql-connector-java.jar |

执行结果如下：

图：执行结果图

1. 创建文件夹

主节点创(hadoop155)建文件夹：

|  |
| --- |
| mkdir /var/cloudera-scm-server  chown -R cloudera-scm:cloudera-scm /var/cloudera-scm-server  chown -R cloudera-scm:cloudera-scm /opt/cloudera\* |

所有节点创建：

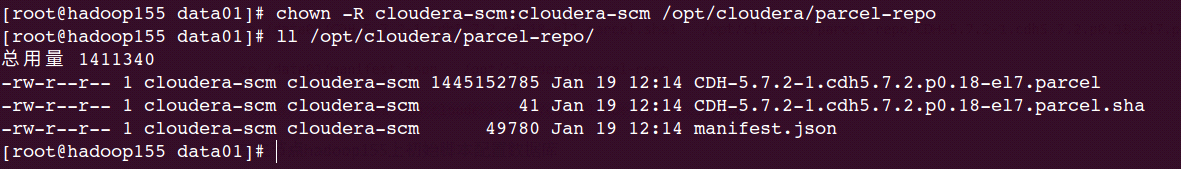
|  |
| --- |
| mkdir -p /opt/cloudera/parcels  chown cloudera-scm:cloudera-scm /opt/\* |

5、上传CDH parcels。

在主机hadoop155上创建parcel-repo目录，把准备好的CDH-5.7.2-1.cdh5.7.2.p0.18-el7.parcel 、CDH-5.7.2-1.cdh5.7.2.p0.18-el7.parcel.sha1、 manifest.json三个文件上传到硬盘/data01下，然后复制到已创建的/opt/cloudera/parcel-repo目录下，修改CDH-5.7.2-1.cdh5.7.2.p0.18-el7.parcel.sha1文件的后缀.sha1改为.sha，并且修改/opt/cloudera/parcel-repo目录及其目录下面的文件的属主、属组为cloudera-scm。

|  |
| --- |
| mv /home/CDH-5.15.1-1.cdh5.15.1.p0.4-el7.parcel /opt/cloudera/parcel-repo/  mv /home/CDH-5.15.1-1.cdh5.15.1.p0.4-el7.parcel.sha /opt/cloudera/parcel-repo/CDH-5.15.1-1.cdh5.15.1.p0.4-el7.parcel.sha  mv /home/manifest.json /opt/cloudera/parcel-repo/  chown -R cloudera-scm:cloudera-scm /opt/cloudera\* |

执行结果如下：



图：执行结果图

6、初始化数据库。

在主机hadoop155上初始化数据库。

|  |
| --- |
| /opt/cm-5.15.1/share/cmf/schema/scm\_prepare\_database.sh mysql -hhadoop1 -uroot -p123456 --scm-host hadoop1 scmdbn scmdbu scmdbp |

/opt/cloudera-manager/cm-5.7.2/share/cmf/schema/scm\_prepare\_database.sh是用来创建和配置CM server数据库的脚本。各参数说明如下：

mysql：使用的数据库类型为mysql(mariadb)。

-hhadoop155：数据库安装所在主机。

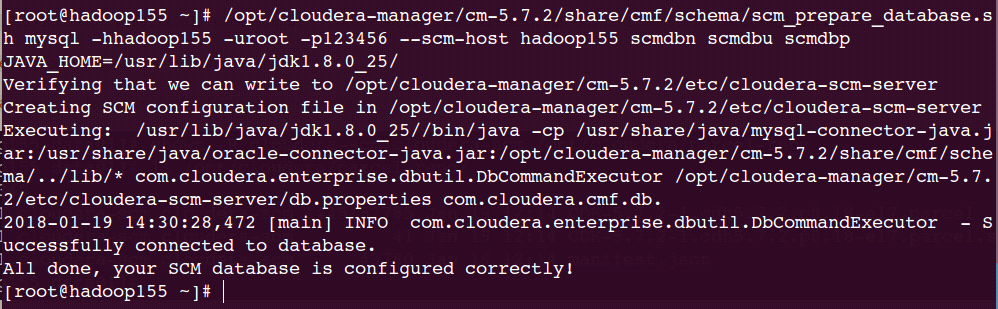
-uroot：以root身份登录mysql。

-p123456：登录密码。

--scm-host hadoop155：CM Server所在的主机，一般是和mysql安装的主机是在同一个主机上。

最后三个参数分别是：数据库名称，数据库用户名，数据库密码。

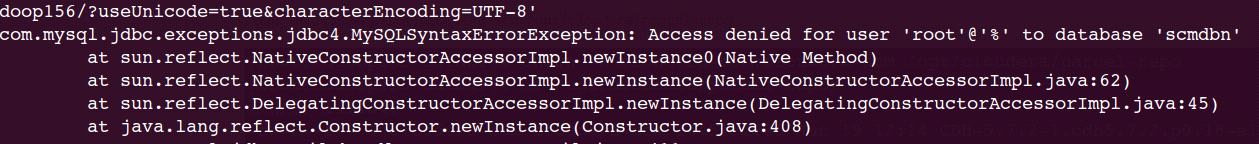
如果出现下图中的 All done，your SCM database is configured correctly！ 则说明数据库配置正确。



图：执行结果图

这一步骤容易出错，可能出现的情况如下：

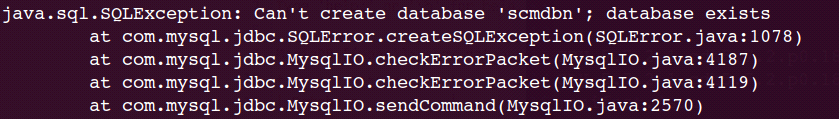
(1)如果出现Access denied for user 'root'@'localhost' (using password: YES)：



图：执行结果图

请检查/etc/hosts里面是否有 127.0.0.1 localhost，然后确认mariadb的数据库创建语句中权限分配是否正确。

(2)如果出现Can't create database 'scmdbn'; database exists：

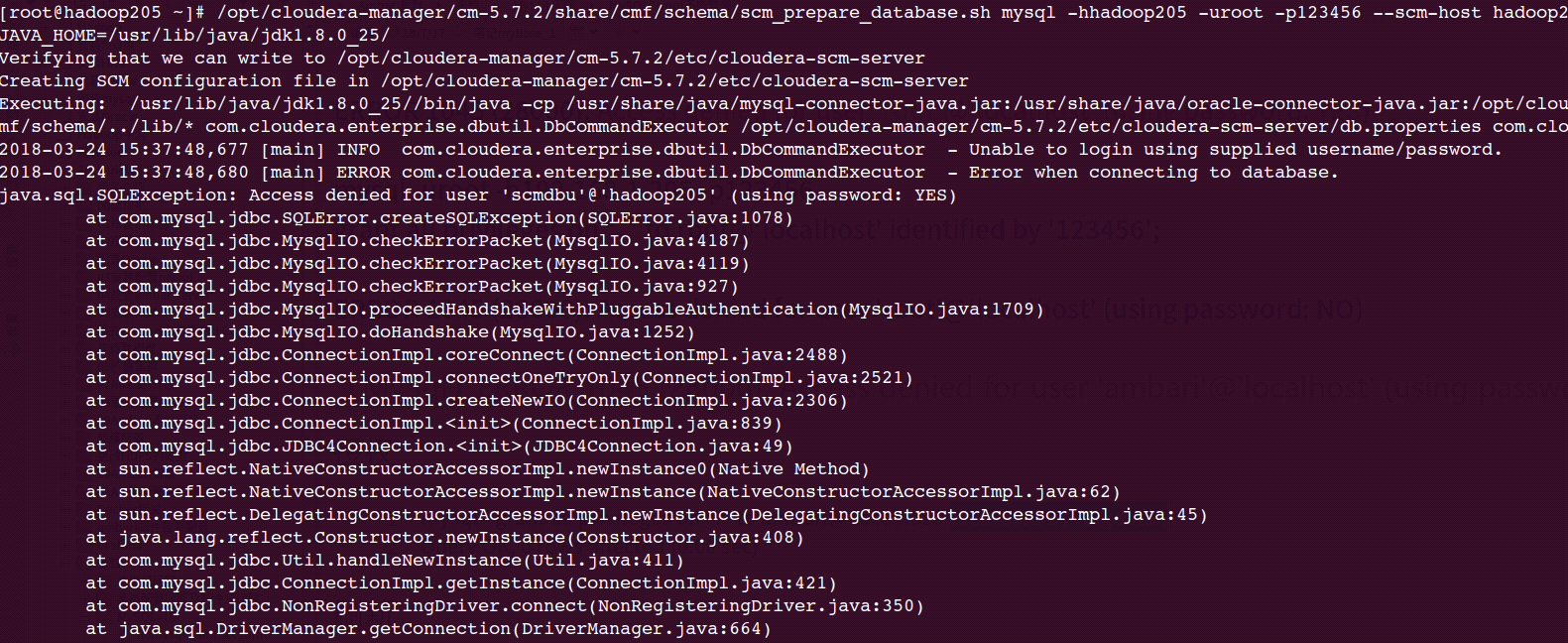


图：执行结果图

请登录mysql，删除该scmdbn数据库，再重新执行上面的初始化数据库的SQL命令，删除该数据库的SQL语句为：

|  |
| --- |
| drop database scmdbn; |

（3）java.sql.SQLException: Access denied for user 'scmdbu'@'hadoop205' (using password: YES)



图：执行结果图

以root登录mysql重新赋权

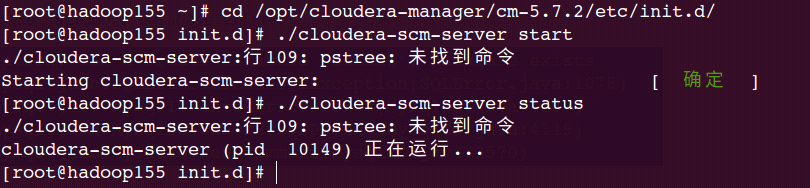
|  |
| --- |
| GRANT ALL PRIVILEGES ON \*.\* TO 'scmdbu'@'hadoop1' identified by 'scmdbp';  GRANT ALL PRIVILEGES ON \*.\* TO 'scmdbu'@'%' identified by 'scmdbp';  GRANT ALL PRIVILEGES ON \*.\* TO 'scmdbu'@'localhost' identified by 'scmdbp';  drop database scmdbn; |

7、启动CM server。

在hadoop155主机上启动cloudera-scm-server。

|  |
| --- |
| /opt/cm-5.15.1/etc/init.d/cloudera-scm-server start  /opt/cm-5.15.1/etc/init.d/cloudera-scm-server status |

执行结果如下：



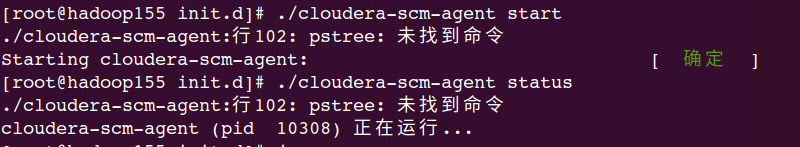
图：执行结果图

8、启动CM agent。

在所有主机上启动cloudera-scm-agent。

|  |
| --- |
| /opt/cm-5.15.1/etc/init.d/cloudera-scm-agent start  /opt/cm-5.15.1/etc/init.d/cloudera-scm-agent status |

执行结果如下：



图：执行结果图

9、停止CM的命令。

如果需要停止CM，应分别停止CM Server和所有的CM agent，操作如下：

|  |
| --- |
| /opt/cm-5.15.1/etc/init.d/cloudera-scm-agent stop |

|  |
| --- |
| /opt/cm-5.15.1/etc/init.d/cloudera-scm-server stop |

【说明】：如果需要重新启动CM，只需执行上面的9、7、8三个个步骤。

至此，CM已装完成，在个人电脑的浏览器中输入10.110.200.155:7180访问Admin Console（如果已配置hosts，可以使用hadoop155:7180进行访问），出现登录页面则说明安装成功，默认的用户名密码为admin/admin。（如果CM正常启动，但仍然无法页面，请检查防火墙和SELinux是否关闭）

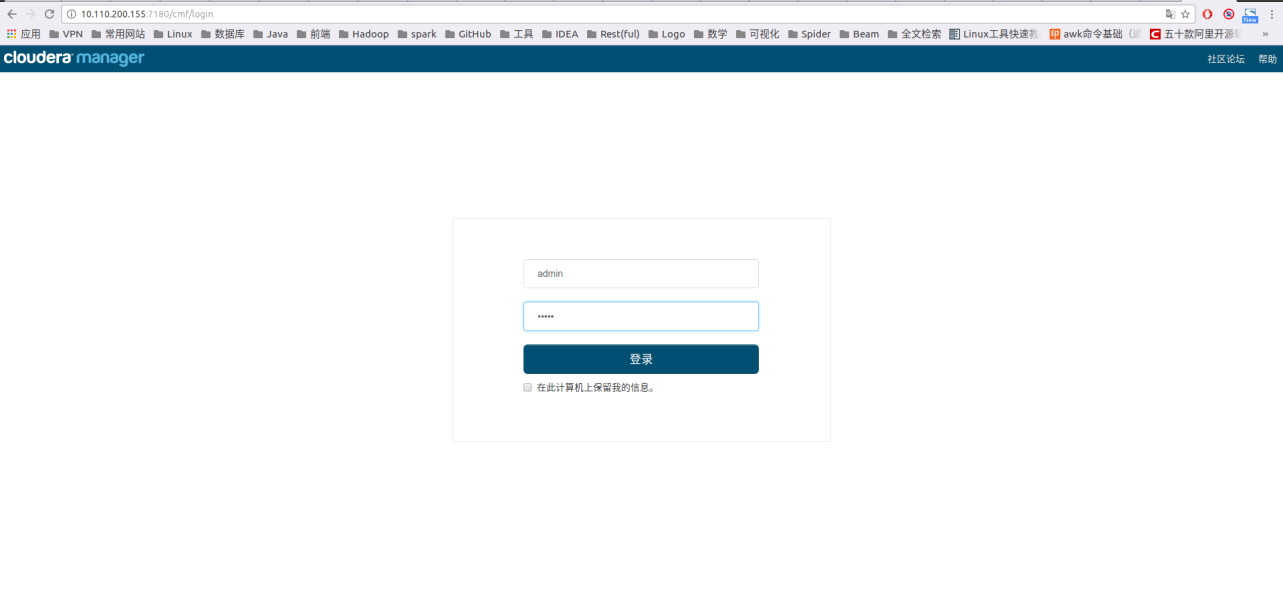
接下来通过访问Admin Console来安装CDH及相关服务。

* **5.2 安装CDH及相关服务**

登录后，在UI界面选择CDH安装方式的页面，只能选择使用Parcels或者Packages其中一种方式进行安装，如果之前已经手动安装了CDH以及其它组件的Packages，那么就无法使用parcel的方式进行安装。parcel是一个序列化后的二进制打包文件，附带着编程文件和一些元数据文件，是官方推荐使用的CDH安装方式之一。

这里选择使用本地的parcel repository进行安装。后续只要根据Admin Console的界面指引选择符合自己需求的配置，完成安装就可以了。CM Server会帮助用户完成绝大多数的配置。

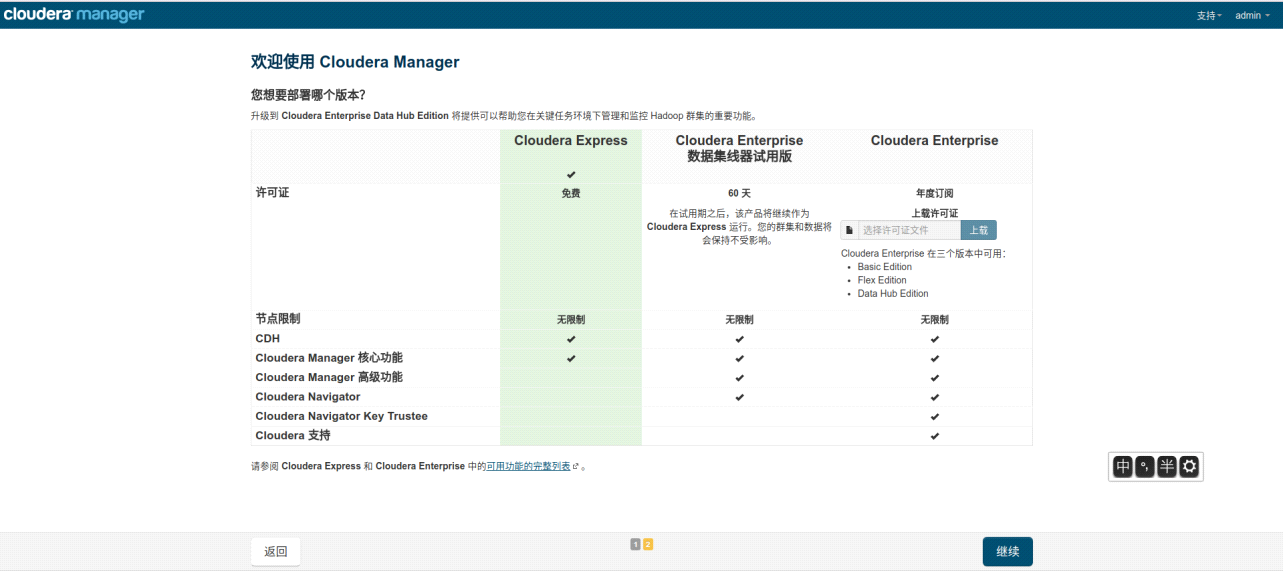
以下仅对主要步骤进行介绍。

1、输入用户名、密码进行登录。（admin/admin）

图：系统安装过程图

2、选择需要安装的版本，本次安装选择免费版本。

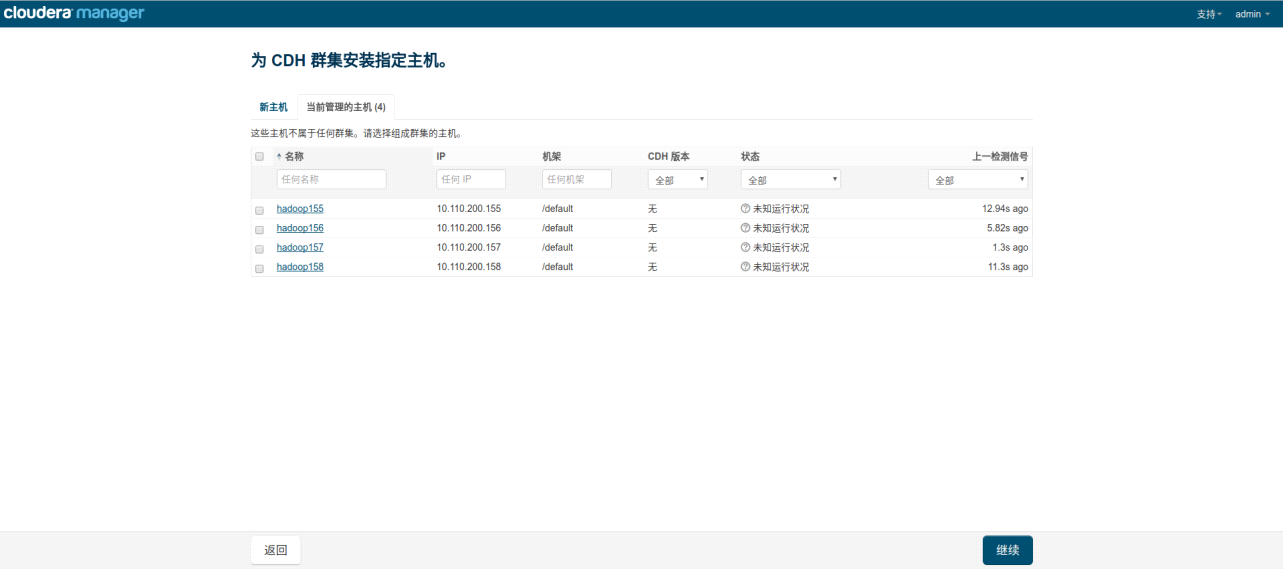
如果安装失败，可以重启CM Server进程重新安装：



图：系统安装过程图

3、为CDH集群安装指定主机。

这里我们可以在“当前管理的主机”中看到4个主机信息，全部勾选并继续下一步。



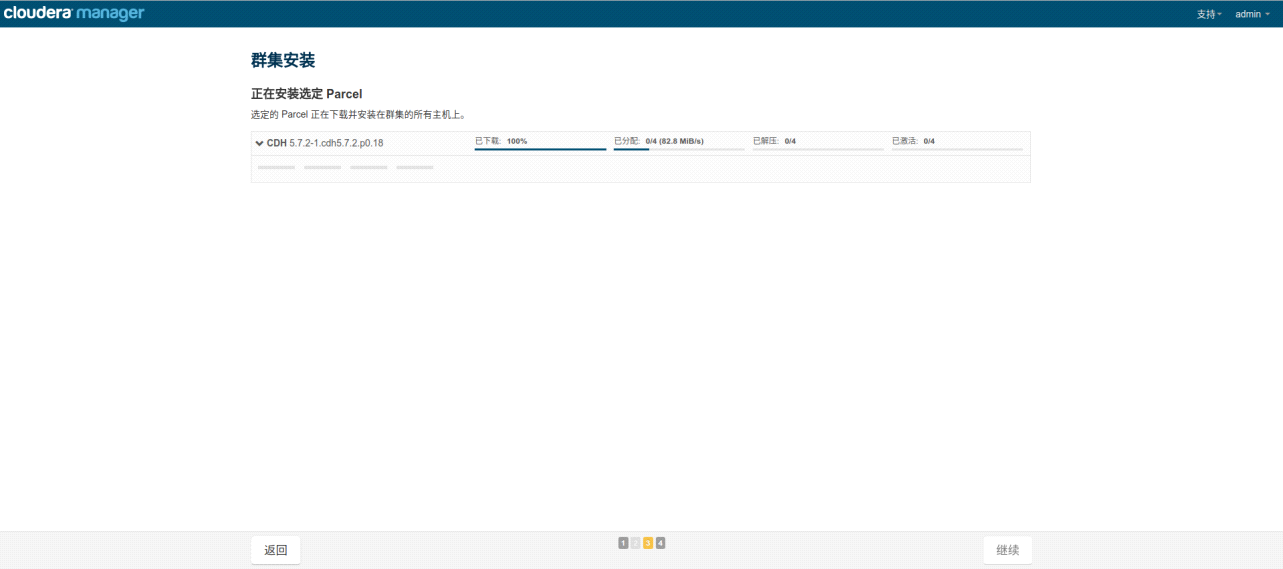
图：系统安装过程图

4、集群安装的选择方式。



图：系统安装过程图

5、安装Parcel到所选的主机。



图：系统安装过程图

如果这一步骤完全就不能把文件发送到某个节点,请检查主节点（hadoop155）与问题节点的ssh连接是否正常。

6、Parcel安装完成。



图：系统安装过程图

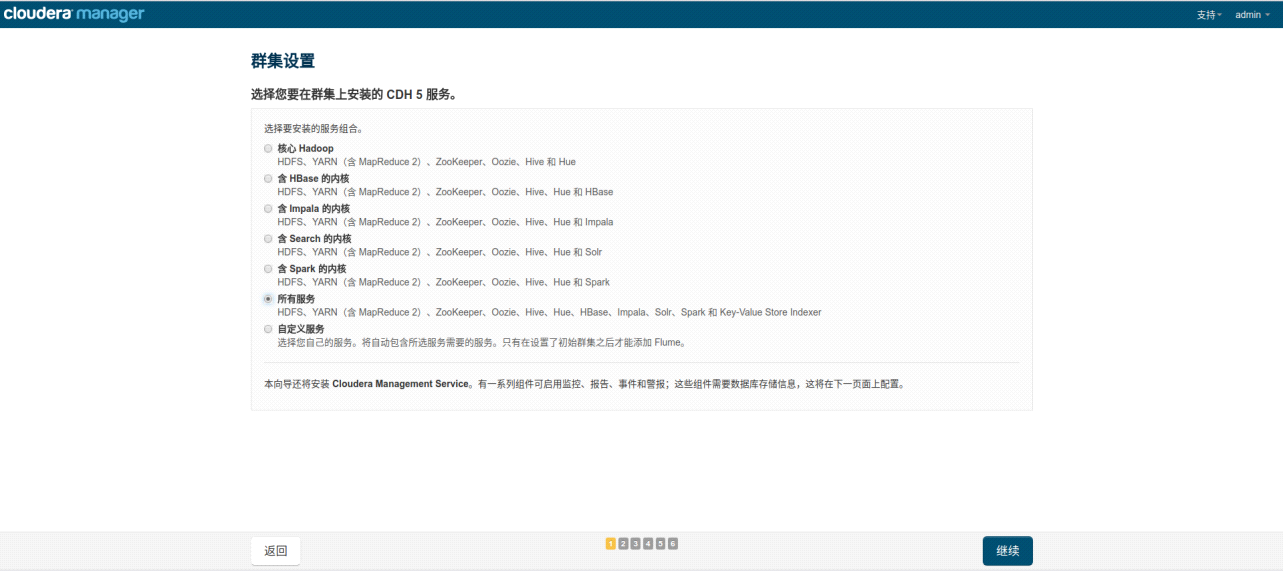
7、检查主机设置的正确性。



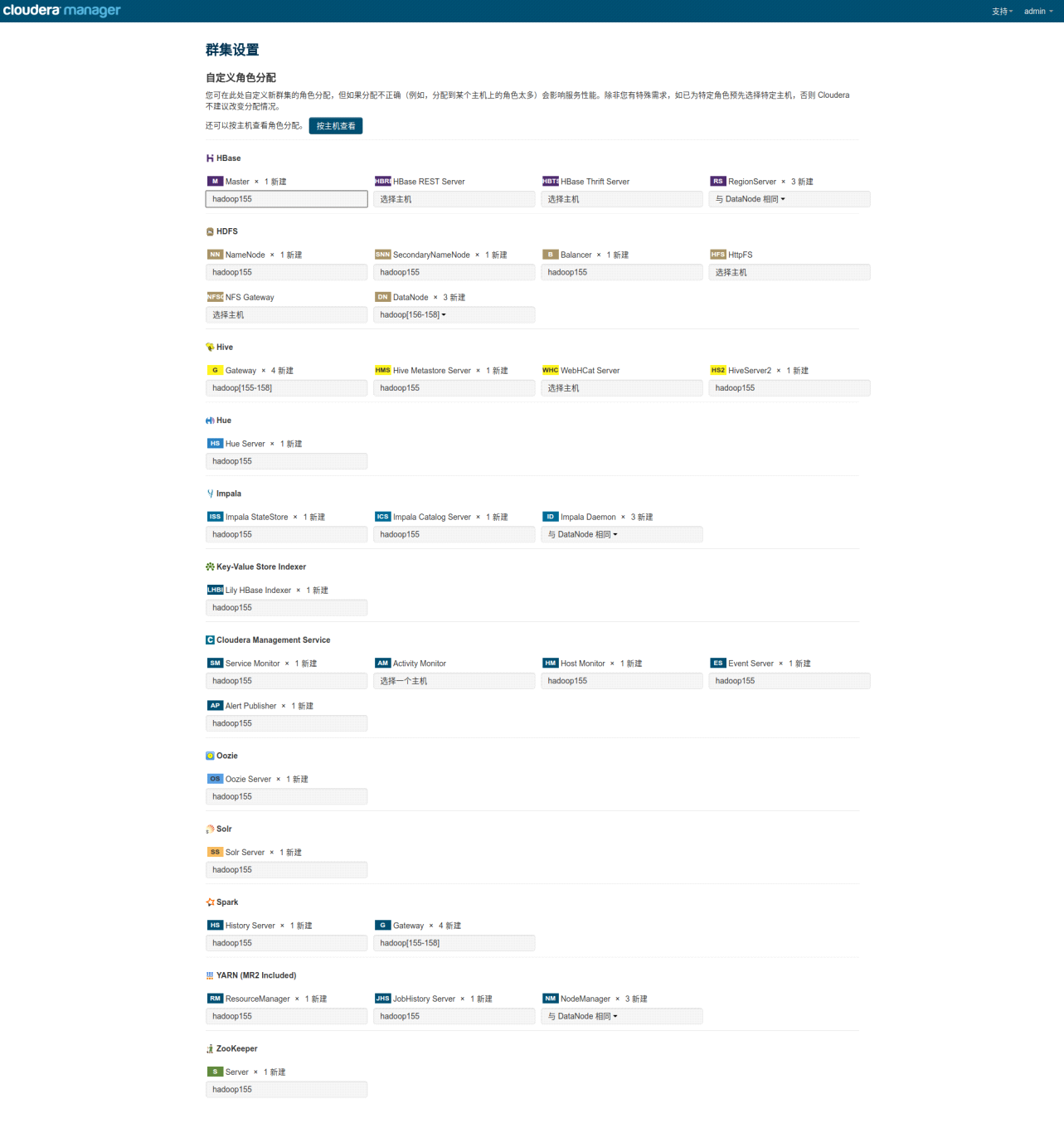
图：系统安装过程图

如果3.9关闭THP，3.10修改swappiness没有正确配置，可能会出现上面的警告，请执行相应的操作进行解决。

8、安装服务，这里根据需要选择安装的服务。

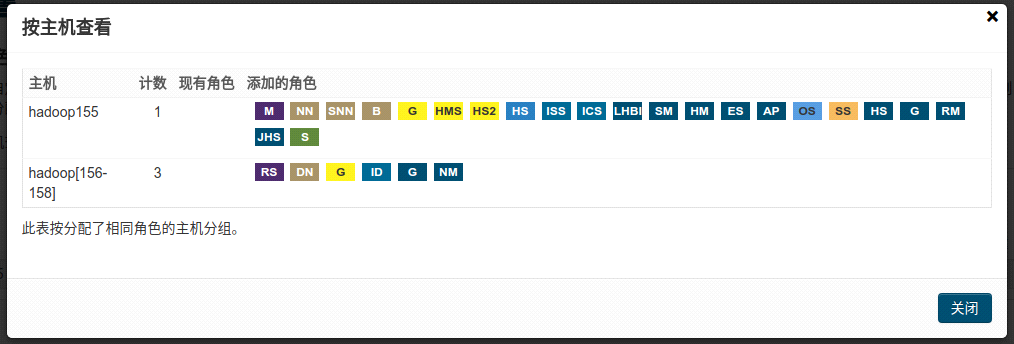


图：系统安装过程图

9、根据规划选择相应服务安装到指定的主机上。

图：系统安装过程图

10、查看服务。选择“按主机查看”，查看不同主机上装了哪些服务。



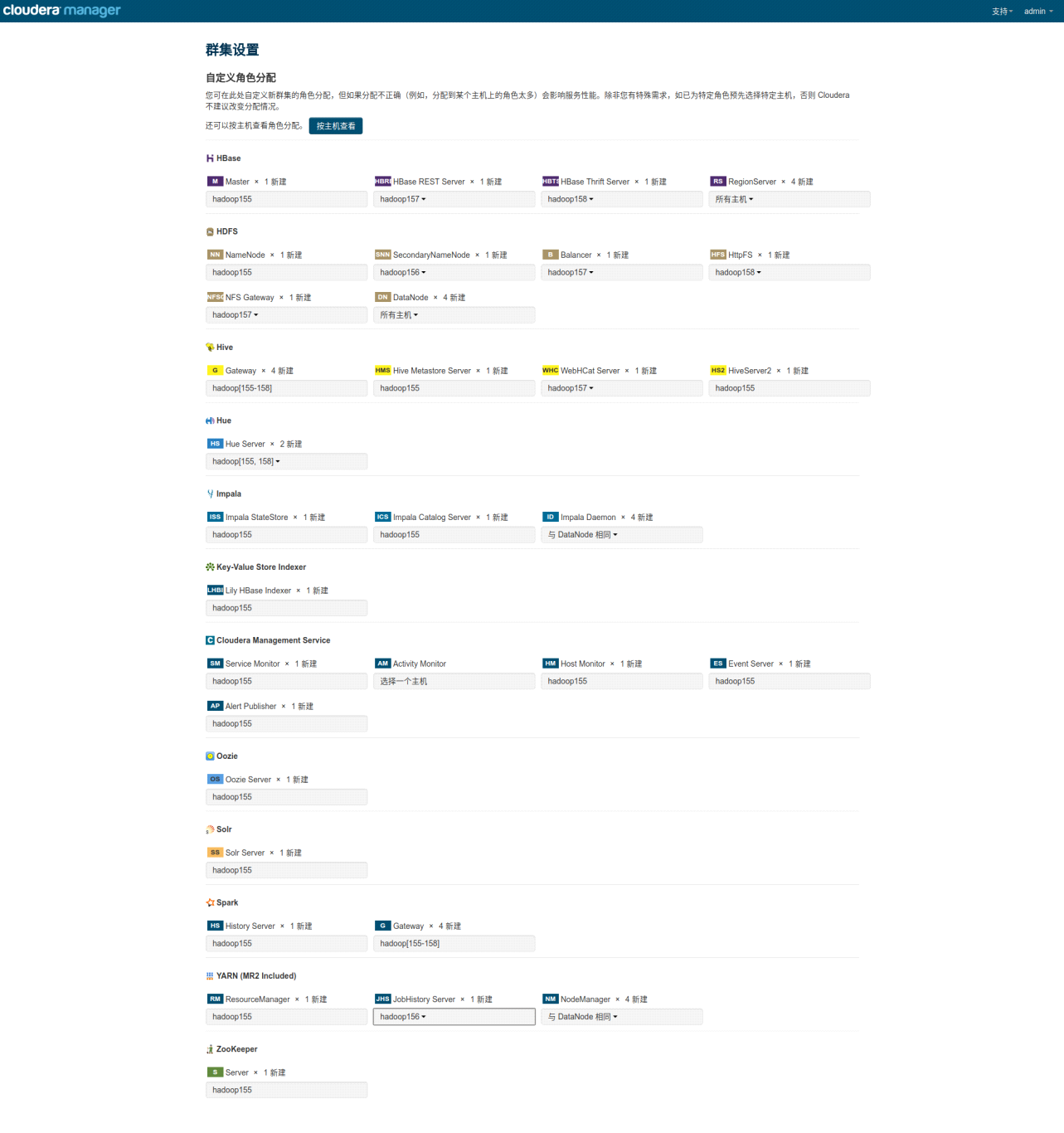
图：系统安装过程图

11、服务确认。确认各主机上所安装的服务。



图：系统安装过程图

12、最终，各主机上安装的服务如下。



图：系统安装过程图

服务安装一般规则：

（1）一般来说，Master-Slave结构的服务Master只能安装到一台主机上（例如NameNode），而Slave则可以同时安装到多台(>=2)主机上（例如DataNode）。

（2）NameNode和SecondaryNameNode不能安装在同一主机上，这两台主机的硬件配置需要一致。

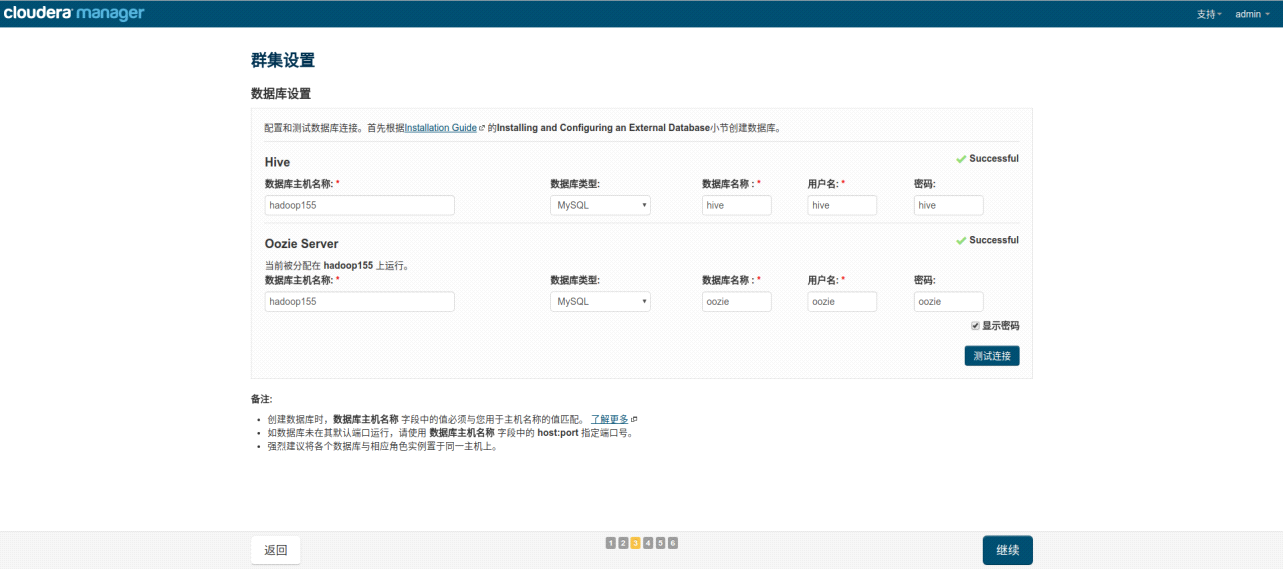
（3）建议所有DataNode配置一致，并且DataNode主机都安装Yarn的RegionServer。

（4）各服务最好合理分配到不同节点，避免单个节点同时运行多种任务，导致负载过大。

（5）注意DataNode至少要安装3台机器，HDFS默认备份因子为3，如果DataNode少于3台会报错“CDH副本不足的块，集群中有293个副本不足的块。集群众共有296个块。百分比副本不足的块：98.99%。临界阈值：40%  Under-Replicated Blocks” 。

12、配置hive和oozie数据库信息。

填写之前已创建的hive和oozie数据库名称、数据库用户名、密码，点击“测试连接”，如果数据库创建及赋权都正确，则会出现如下所示的绿色对勾和Successful提示：



图：系统安装过程图

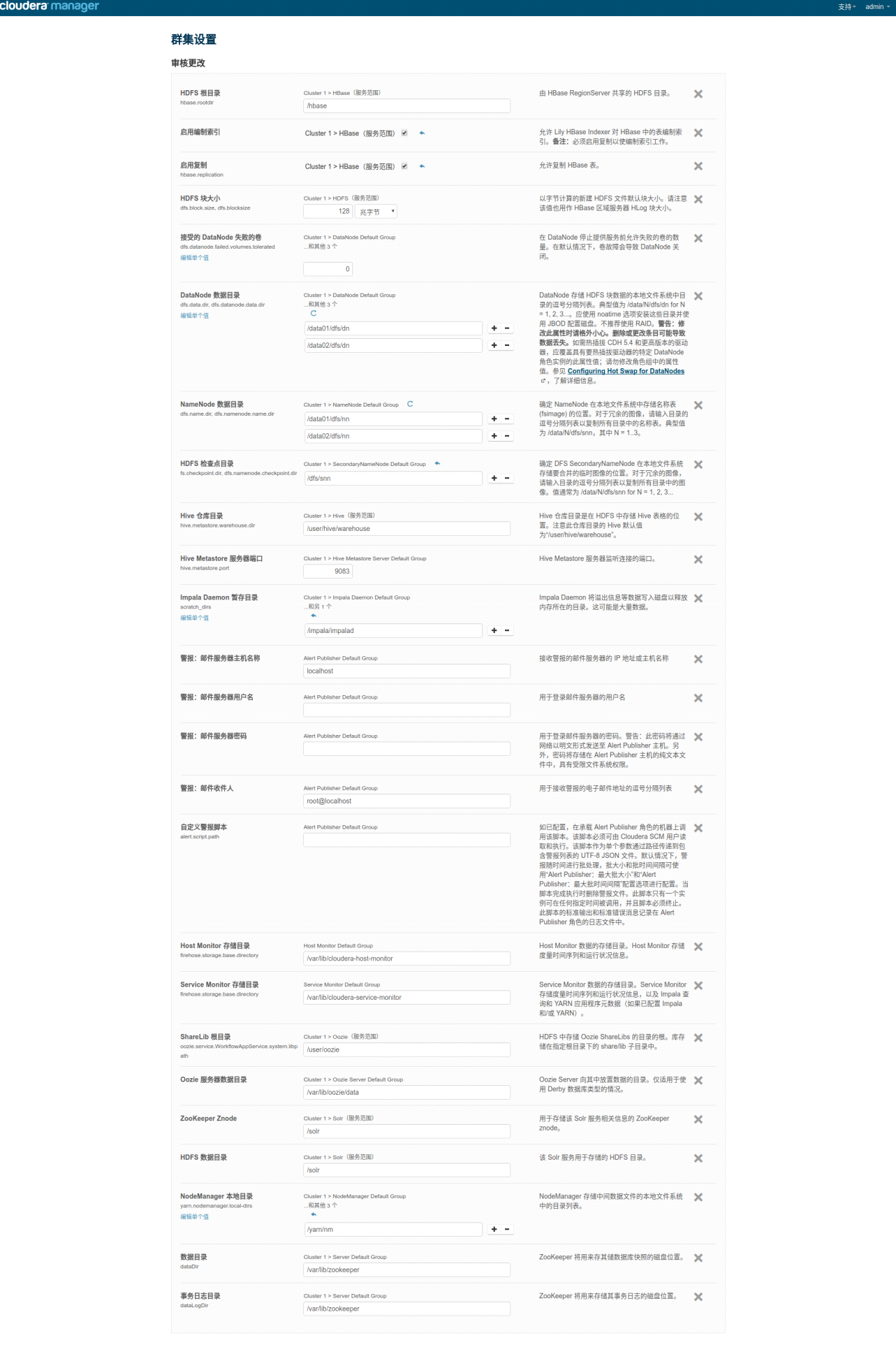
如果hive和oozie的数据库、用户创建以及赋权不正确，则会出现如下错误：

（1）Logon denied for user/password. Able to find the database server and database, but logon request was rejected

（2）Unable to connect to database with provided credential. Able to find the Database server, but not the specified database. Please check if the database name is correct and make sure that the user can access the database.

出现这种情况请检查hive和oozie的数据库配置，主要是赋权操作。

13、服务具体配置。



图：系统安装过程图

重点对以下参数进行设置，其他默认即可：

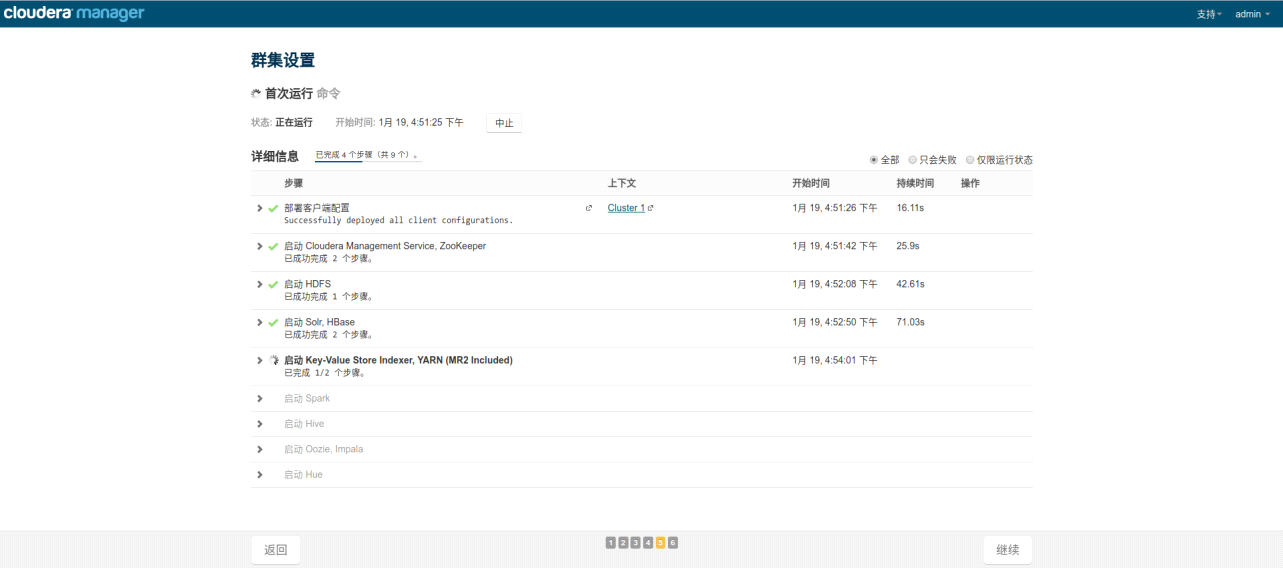
（１）设置HDFS的NN和DN的数据目录。

NN数据目录，/data01/dfs/nn 、 /data02/dfs/nn；

DN数据目录，/data01/dfs/dn 、 /data02/dfs/dn ；

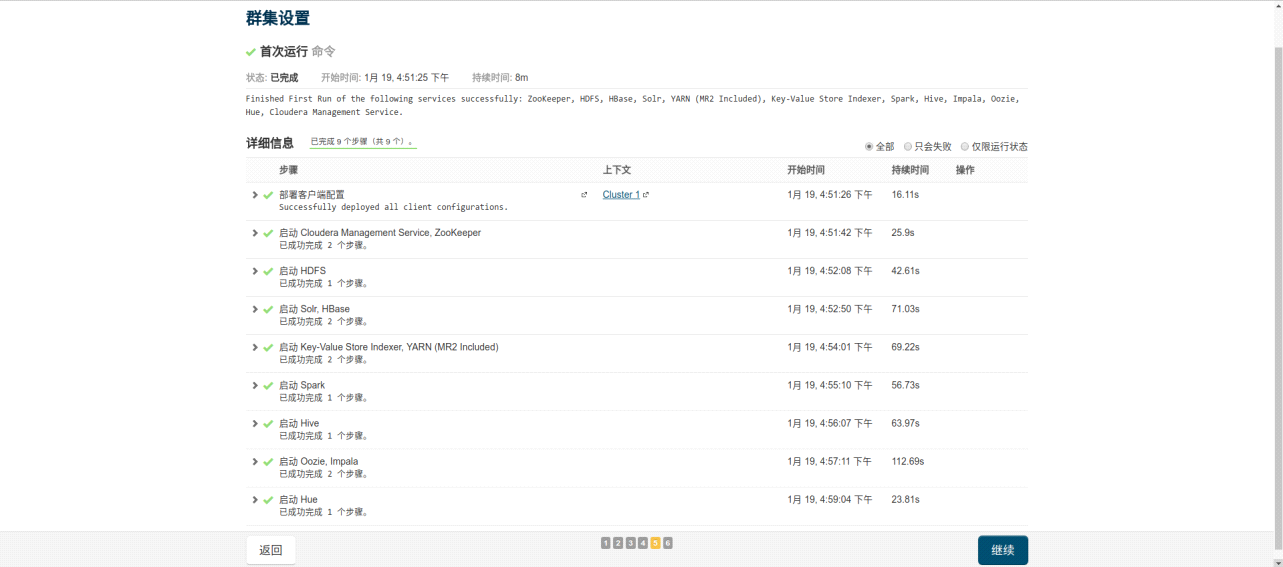
（２）Hbase根目录：/data01/hbase。

14、配置完成后，开始安装服务。



图：系统安装过程图

15、等待安装完成。



图：系统安装过程图

16、恭喜！服务安装成功。



图：系统安装过程图

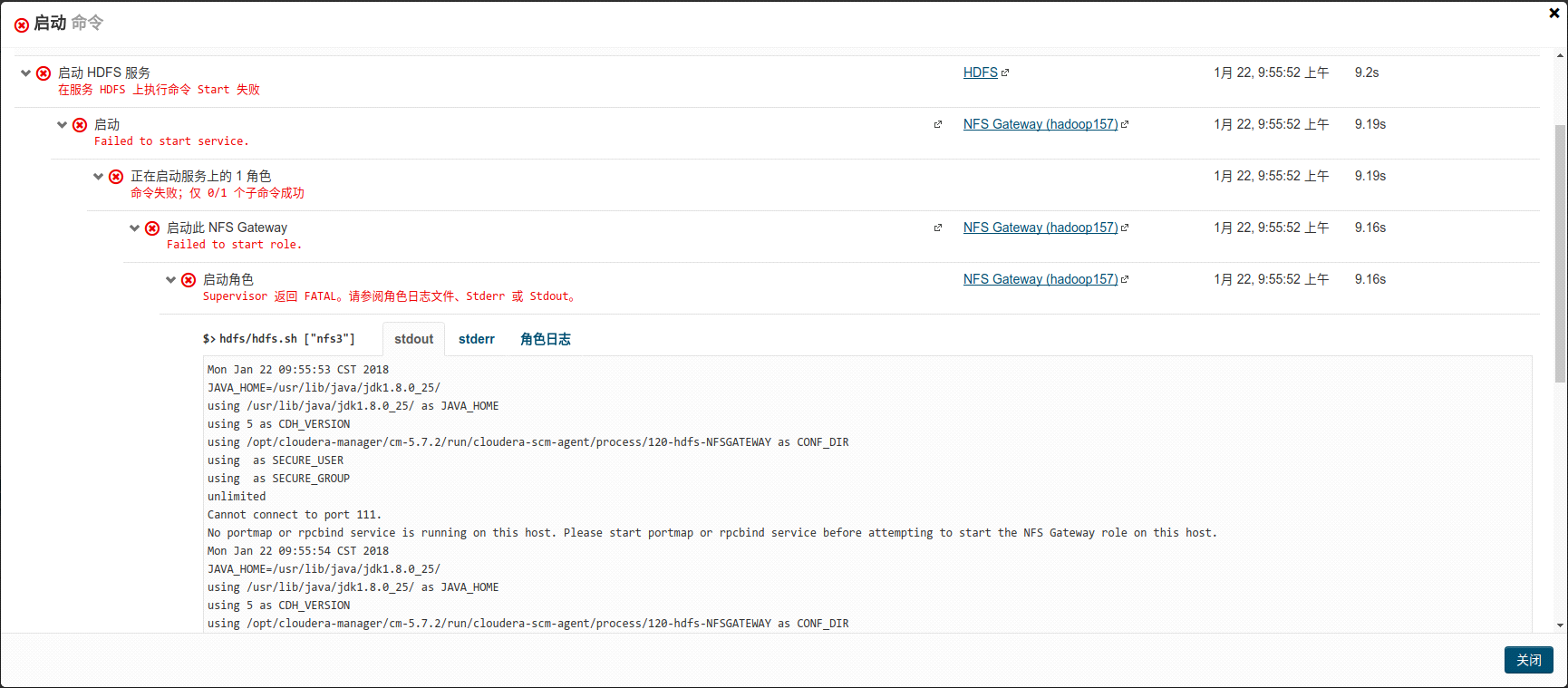
17、安装完成后可能出现的问题。

一般来说，安装完成后会出现一些错误或警告，此时不用担心，点击 “所有运行状况问题”后面的数字，查看相应的报错信息，逐个解决这些问题。



图：系统安装过程图

一般情况下会出现HDFS下的NFS GATEWAY启动失败，报错如下：



图：系统安装过程图

解决办法：

在NFS-GATEWAY所在主机上安装 nfs-utils portmap，启动并且设置rpcbind开机启动，不要启动nfs，然后再启动NFS-GATEWAY。

|  |
| --- |
| systemctl stop nfs  systemctl start rpcbind  systemctl enable rpcbind |

执行结果如下：

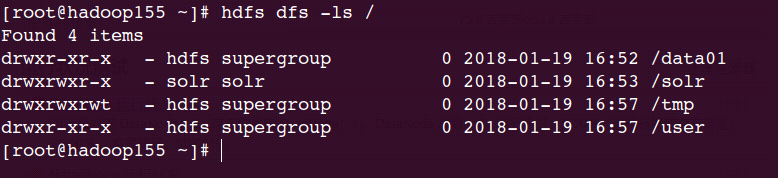


图：执行结果图

* **Hadoop测试及应用**
* **6.1 Hadoop测试**

选择hadoop155主机作为HDFS Client，执行hdfs命令。

1、通过命令查看HDFS文件目录：



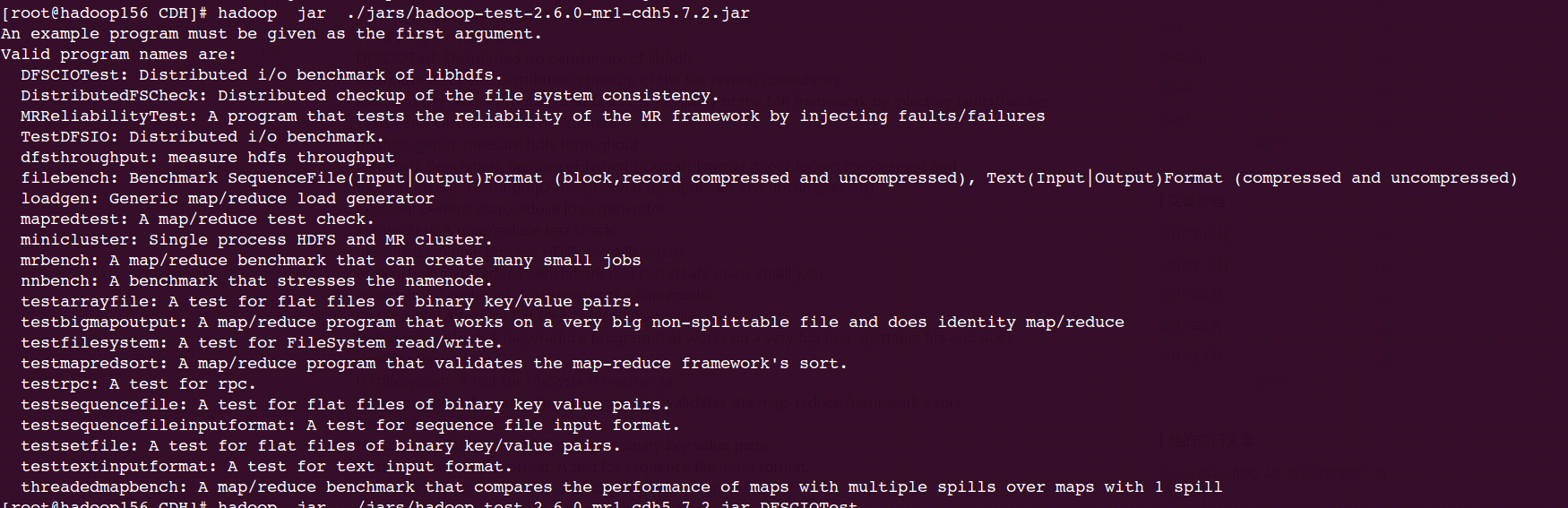
图：执行结果图

2、运行自带的测试程序：

（1）查看有哪些测试程序：

|  |
| --- |
| cd /opt/cloudera/parcels/CDH  hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar |

执行结果如下：



图：执行结果图

（2）TestDFSIO write 测试HDFS的IO性能：

用一个MapReduce作业来并发地执行读写操作，每个map任务用于读或写每个文件，map的输出用于收集与处理文件相关的统计信息，reduce用于累积统计信息，并产生summary。

Usage: TestDFSIO [genericOptions] -read [-random | -backward | -skip [-skipSize Size]] | -write | -append | -clean [-compression codecClassName] [-nrFiles N] [-size Size[B|KB|MB|GB|TB]] [-resFile resultFileName] [-bufferSize Bytes][-rootDir]

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar TestDFSIO -wriOresults.txt10 -fileSize 1000 -resFile /tmp/TestDFSI |

【说明】：如果使用root用户，在cloudera下root用户没有HDFS的根目录/写权限，会报错 java.io.IOException: Permission denied: user=root, access=WRITE, inode="/":hdfs:supergroup:drwxr-xr-x ，所以要切换到hdfs用户（su - hdfs）执行。

|  |
| --- |
| su - hdfs; |

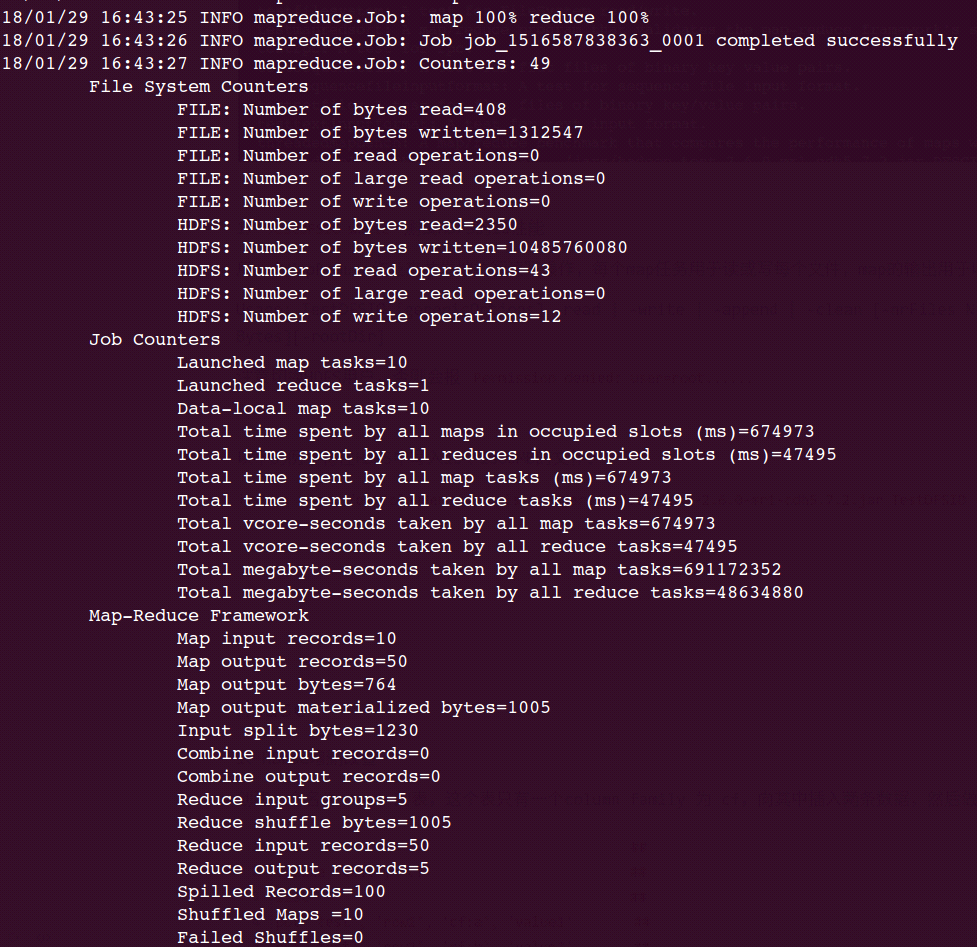
测试Write:

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar TestDFSIO -write -nrFiles 10 -fileSize 1000 -resFile /tmp/TestDFSI |

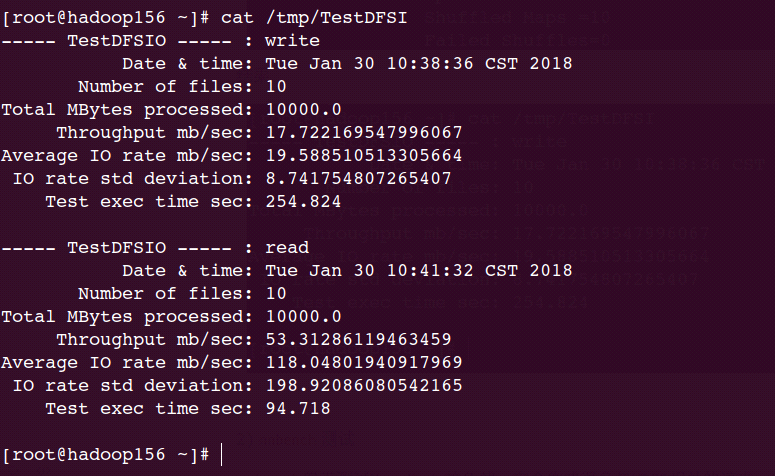
测试Read:

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar TestDFSIO -read -nrFiles 10 -fileSize 1000MB -resFile /tmp/TestDFSI |

执行结果如下：



图：执行结果图



图：执行结果图

清除测试文件Clean:

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar TestDFSIO -clean |

最后，测试写入10个1000MB的文件：

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar TestDFSIO -write -nrFiles 10 -fileSize 1000 |

（3）nnbench测试：

nnbench用于测试NameNode的负载，它会生成很多与HDFS相关的请求，给NameNode施加较大的压力。这个测试能在HDFS上模拟创建、读取、重命名和删除文件等操作。

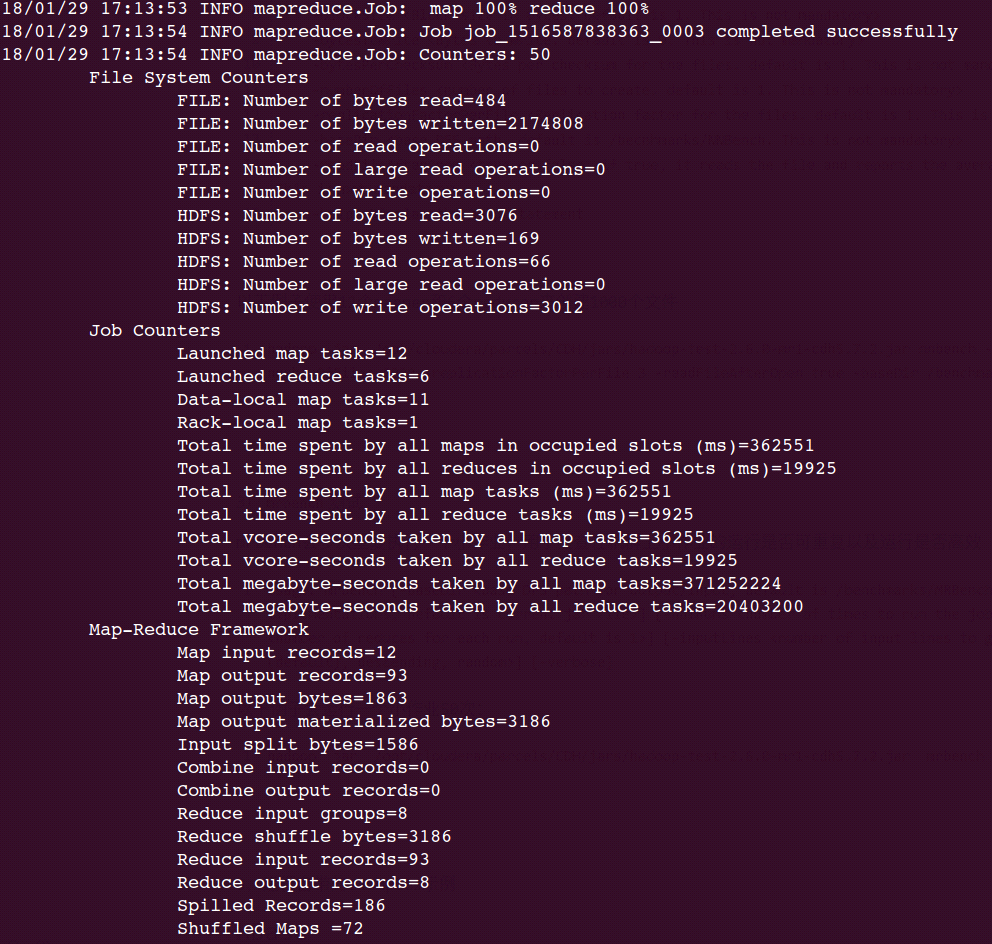
执行以下命令查看用法：

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar nnbench |

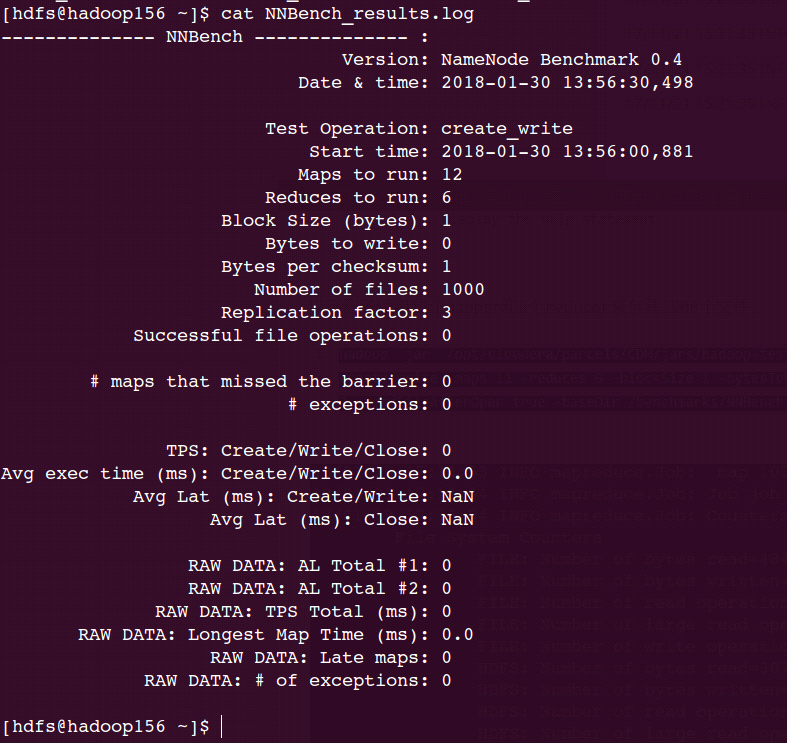
下面例子使用12个mapper和6个reducer来创建1000个文件：

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar nnbench -operation create\_write -maps 12 -reduces 6 -blockSize 1 -bytesToWrite 0 -numberOfFiles 1000 -replicationFactorPerFile 3 -readFileAfterOpen true -baseDir /benchmarks/NNBench-`hostname -s` |

执行结果如下：



图：执行结果图



图：执行结果图

（4）mrbench测试

mrbench会多次重复执行一个小作业，用于检查在机群上小作业的运行是否可重复以及运行是否高效。

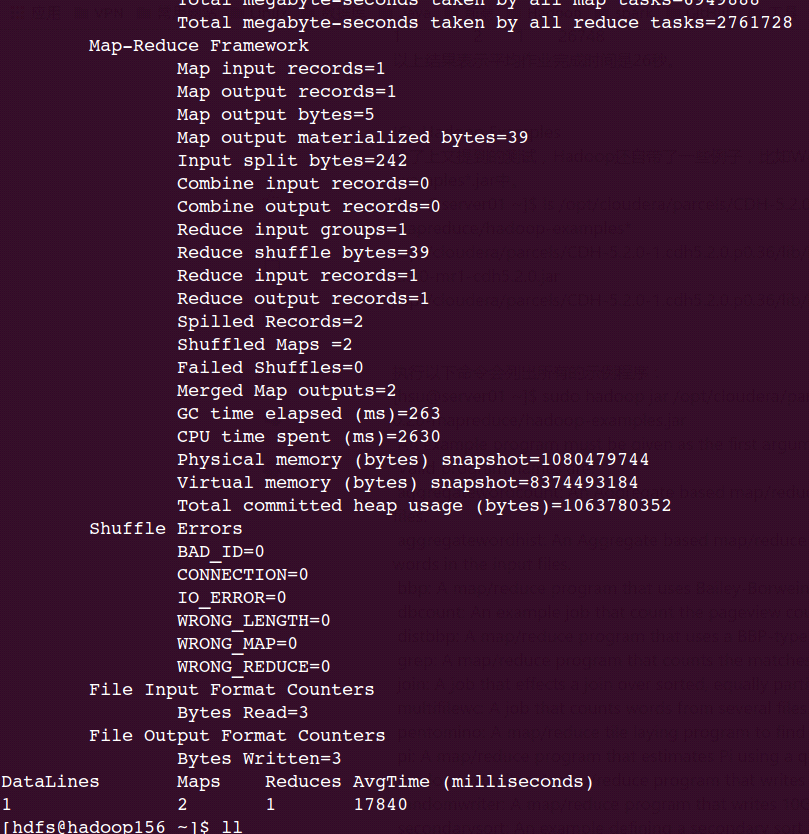
执行以下命令查看用法：

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar mrbench |

以下例子将运行一个小作业50次（运行时间可能会很长）：

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-test-2.6.0-mr1-cdh5.7.2.jar mrbench -numRuns 50 |

执行结果如下:



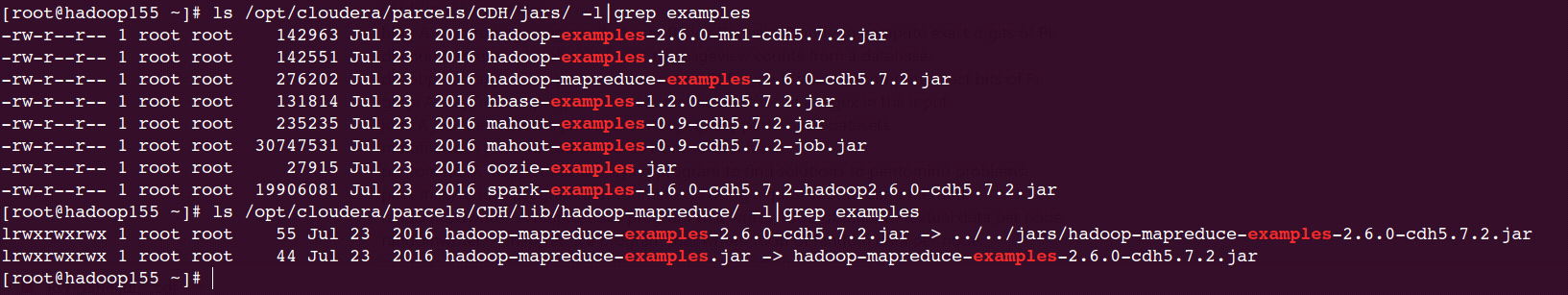
图：执行结果图

* **6.2 Hadoop Examples示例**

1、查找examples示例位置：

|  |
| --- |
| ls /opt/cloudera/parcels/CDH/jars/ -l|grep examples  ls /opt/cloudera/parcels/CDH/lib/hadoop-mapreduce/ -l|grep examples |

执行结果如下：



图：执行结果图

查看有哪些example：

|  |
| --- |
| hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-examples.jar |

（2）WordCount测试。

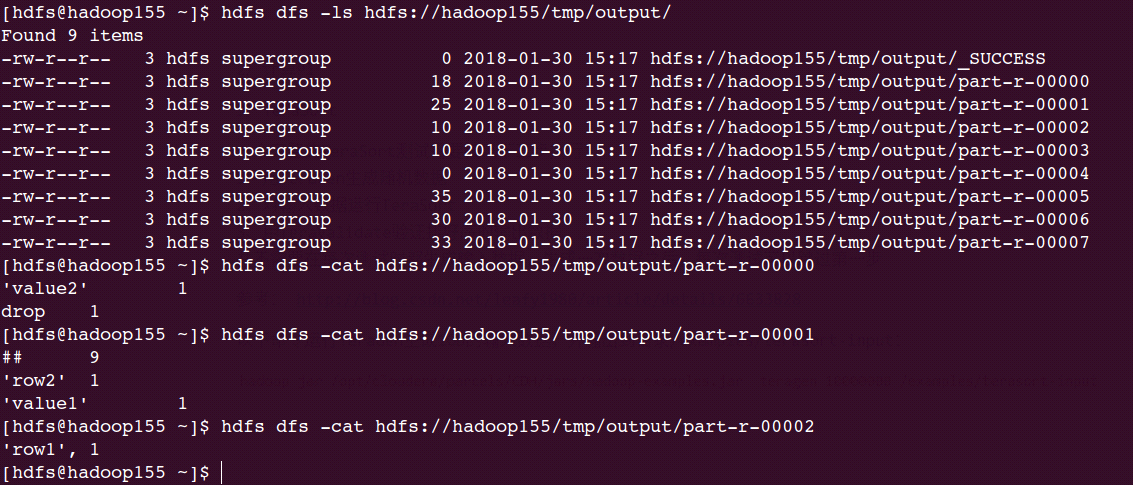
首先创建测试文件：

|  |
| --- |
| su - hdfs  vim /tmp/input/  hello Hadoop  hello java  hello php  hello python  … |

然后上传文件，并运行：

|  |
| --- |
| hdfs dfs -put /tmp/input hdfs://hadoop155/tmp/input  hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-examples.jar wordcount /tmp/input /tmp/output |

查看运行结果：



图：执行结果图

WordCount仅支持统计英文单词或者字符串。

（3）其他测试。

其他example测试：如估算PI的值、TeraSort…请查阅相关资料自行完成测试。

* **6.3 HBase应用示例**

1、使用shell命令测试HBase增删改查。

启动shell：

|  |
| --- |
| hbase shell |

创建表test：

|  |
| --- |
| create 'test', 'cf' |

查看所有表：

|  |
| --- |
| list |

向表中新增数据：

|  |
| --- |
| put 'test', 'row1', 'cf:a', 'value1'  put 'test', 'row2', 'cf:b', 'value2' |

查看表数据：

|  |
| --- |
| scan 'test' |

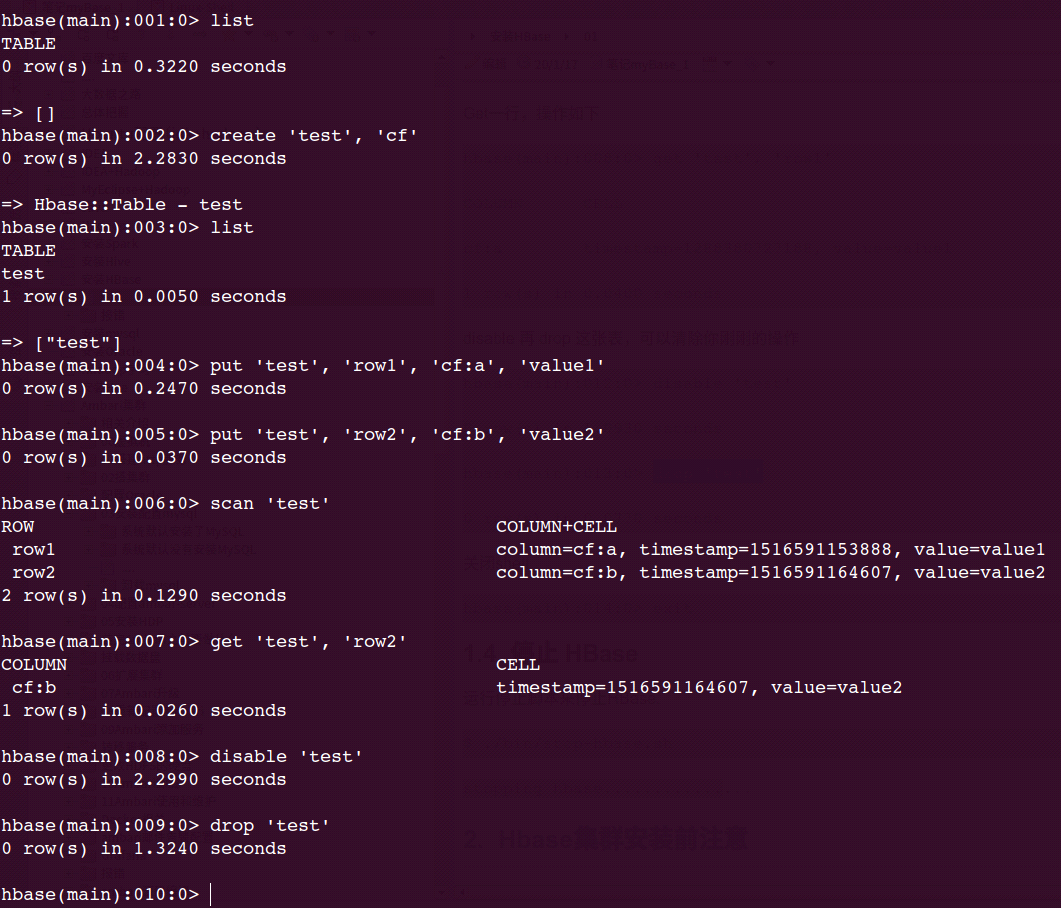
条件查询：

|  |
| --- |
| get 'test', 'row2' |

删除表：

|  |
| --- |
| disable 'test'  drop 'test'  exit |

执行结果如下：



图：执行结果图

2、结合实际场景，完成HBase应用示例。

Tips：

1. 一个节点安装完CentOS系统之后，安装对应服务器的驱动（拷贝驱动文件夹到节点上，修改.sh文件的权限后运行），时间较长，约10~15分钟。
2. 注意:在关闭SELinux时，要修改的项是SELinux,千万不要改成了别的选项,如果没有这一项,就在后面加上一行SELINUX=disabled。如果操作失误修改了SELINUXPOLITY=disabled，会导致无法开机。解决方案是在开机时进入grub模式，添加selinux参数。
3. 节点设置快照的话,尽量在关机或者红线的时候进行。在克隆之前不要设置，不然克隆速度会很慢.
4. 配置ssh免密登陆，可以使用命令：ssh-keygen和ssh-copy-id两条命令来更方便的实现。如A要免密登陆B，则在A节点上执行命令：ssh-keygen来生成自身的密钥，然后执行ssh-copy-id B来把自己的密钥拷贝到B上，这样A就可以免密登陆到B了。

|  |
| --- |
| export JAVA\_HOME=/usr/java/jdk1.8.0\_25  export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar  export PATH=$JAVA\_HOME/bin:$PATH |

1. 这里装了jdk之后，需配置一下PATH。因为安装Spark的时候的时候默认会到/usr/java/default去找javahome，所以一般也安装在/usr/java目录下。设置环境变量并source之后才可以直接使用java相关命令。
2. 后续安装的时候，可能会发生report manager也需要访问数据库的情况，则仿照hive和oozie创建database及设置好权限即可。
3. 如果使用脚本初始化数据库的时候这里出错，错误提示不是上面三种，大概率是因为虽然出错了，但是mysql依然依据有效的命令创建了相应的database，所以需要到mysql中删除掉。
4. 按照上述安装步骤,停止cm进程的命令应为

|  |
| --- |
| /opt/cm-5.7.2/etc/init.d/cloudera-scm-agent stop  /opt/cm-5.7.2/etc/init.d/cloudera-scm-server stop |

1. 在CM界面分配、解压、激活那一步时，可能你的有些节点会出现很多奇怪的问题导致变红,耐心等待,到最后才能确定是否可以完成安装。
2. 如果安装服务的时候发生了错误，就察看安装日志来寻找错误出在哪，具体目录为出问题的节点下的

/opt/cm-5.7.2/run/cloudera-scm-agent/process/

具体是哪个组件出了问题，就到哪个组件的日志目录下看log，大概率是前面安装的一些必要组件没有安装成功，如果没有读log解决问题的相关经验，就到log的最后几行去找相应组件的字眼，并使用yum安装它们。