问题列表：

1. 启动start-yarn.sh提示spark\_shuffle涉及类不存在解决办法

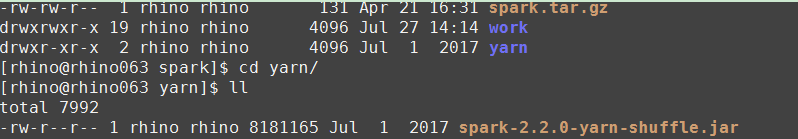
办法一：

进入spark安装包目录找到如下文件：

spark-2.2.0-yarn-shuffle.jar

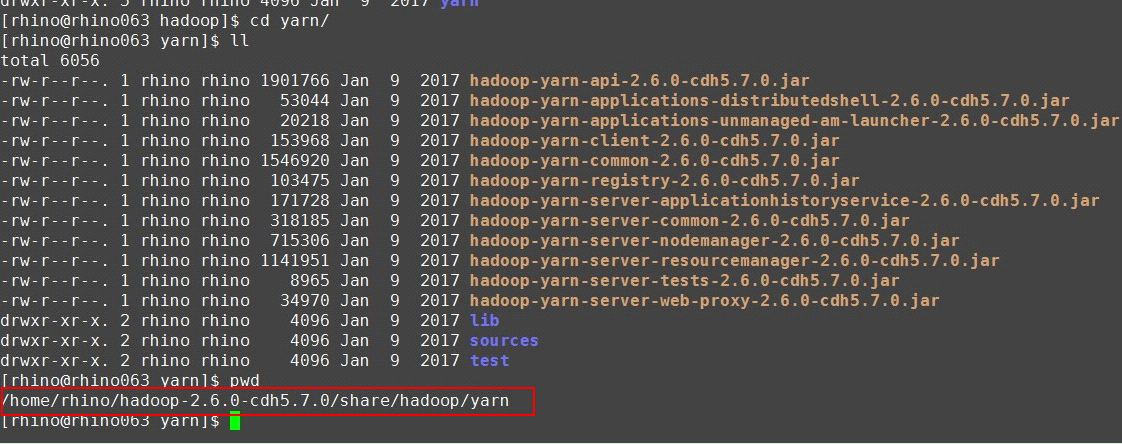
路径：

/home/rhino/spark/yarn



将spark-2.2.0-yarn-shuffle.jar拷贝至hdaoop如下路径：

/home/rhino/hadoop-2.6.0-cdh5.7.0/share/hadoop/yarn



重启yarn

办法二：

上传jar包

* 将spark环境中的jar包放到HDFS

[rhino@rhino001 ~]$hdfs dfs -mkdir hdfs://sinovatiocluster/sharkjars-2.0/

[rhino@rhino001 ~]$hdfs dfs -put ~/spark-2.2.0-bin-hadoop2.6/jars/\* hdfs://sinovatiocluster/sharkjars-2.0/

进入hadoop如下路径：

/home/rhino/hadoop-2.6.0-cdh5.7.0/bin

* 修改yarn文件

[rhino@rhino001 ~]$vim ~/hadoop-2.6.0-cdh5.7.0/bin/yarn

|  |
| --- |
| elif [ "$COMMAND" = "nodemanager" ] ; then  CLASSPATH=${CLASSPATH}:$YARN\_CONF\_DIR/nm-config/log4j.properties  **for f in ${SPARK\_HOME}/yarn/\*.jar; do**  **CLASSPATH=${CLASSPATH}:${f}**  **done**  **#在下面这句前面增加上述内容**  CLASS='org.apache.hadoop.yarn.server.nodemanager.NodeManager' |

* 将yarn文件拷贝到所有Nodemanager节点

[rhino@rhino001 ~]$ scp ~/hadoop-2.6.0-cdh5.7.0/bin/yarn rhino@rhino002:~/ hadoop-2.6.0-cdh5.7.0/bin

[rhino@rhino001 ~]$ scp ~/hadoop-2.6.0-cdh5.7.0/bin/yarn rhino@rhino003~/ hadoop-2.6.0-cdh5.7.0/bin

[rhino@rhino001 ~]$ scp~/hadoop-2.6.0-cdh5.7.0/bin/yarnrhino@rhino004~/ hadoop-2.6.0-cdh5.7.0/bin

1. 调整yarn集群内存量方法

以下配置用于调整yarn集群可以使用的内存量：

1、/home/rhino/hadoop/etc/hadoop/yarn-site.xml:

yarn.nodemanager.resource.memory-mb

说明：每台nodenamager提供多大内存给yarn集群，建议120GB或更多，根据离线流程和impala查询业务比例调整。

2、/home/rhino/hadoop/etc/hadoop/fair-scheduler.xml

配置项：所有队列maxResources配置的内存大小

说明：root队列maxResources为整个yarn集群的cpu和内存资源的总和，在这里总的内存大小即为yarn.nodemanager.resource.memory-mb \* nodemanager数量；

其他队列根据使用情况灵活调整，这些队列资源的总和基本和root队列资源一致即可。

以下配置用于用于调整离线流程每个executor使用的内存量：

1、/home/rhino/.shark/jetty/WEB-INF/conf/spark.properties

配置项：spark.executor.memory

2、/home/rhino/.shark/jetty/WEB-INF/conf/spark-submitTfPy.conf

配置项：executor-memory

3、/home/rhino/.shark/jetty/WEB-INF/classes/spark-submitTfPy.properties

配置项：executor-memory

1. 如何通过命令查看hdfs是否为active还是standby

hdfs haadmin -getServiceState nn1

hdfs haadmin -getServiceState nn2

1. 提示journal没有格式化" Journal Storage Directory \*\* not formatted"

执行：

hdfs namenode -initializeSharedEdits

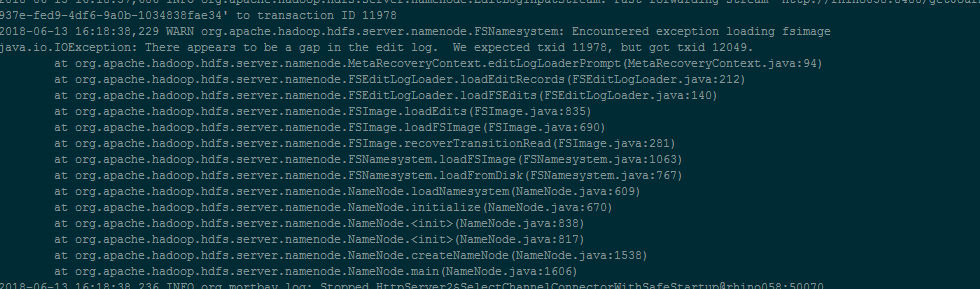
1. 两个NameNode节点均为standby问题，提示“Operation category JOURNAL is not supported in state standby”

解决办法：手动在一个节点执行命令

hdfs haadmin -transitionToActive --forcemanual nn1

1. NameNode节点启动提示“There appears to be a gap in the edit log”

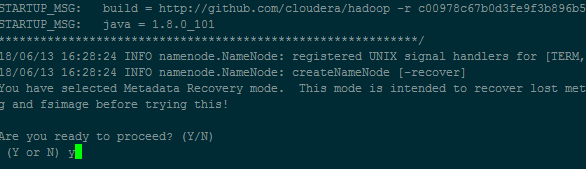
java.io.IOException: There appears to be a gap in the edit log. We expected txid 11978, but got txid 12049.

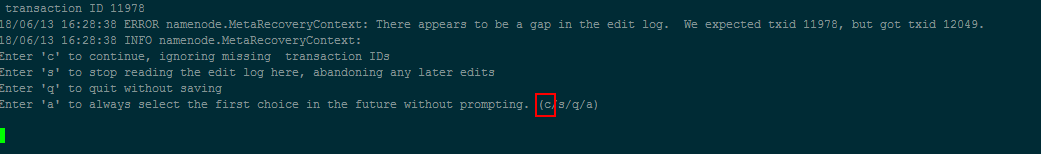


解决办法：

hadoop namenode -recover

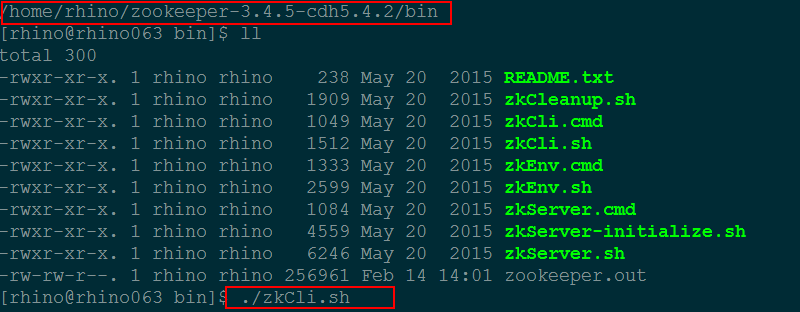
一般选择c。恢复OK





1. Zookeeper中如何清理kafka主题信息

在zookeeper中执行zkCli.sh进入环境



删除如下主题（注意删除前需要 ls查看里面有什么信息，例如：ls /config [changes, clients, topics]）

rmr /brokers

rmr /kafka

rmr /config

1. impalasql查询数据时区相差8小时导致原因

按照时间查询，结果不在查询时间内-----时区相差8小时导致，在start-impala.sh中增加-user\_local\_tz\_for\_unix\_timestamp\_conversions=true -convert\_legacy\_hive\_parquet\_utc\_timestamps=true，重启impala解决

1. 定位异常掉电后hadoop启动失败问题,Journal进程通信出现问题，一直提示9020端口无响应

解决思路：

1. 先停止journalnode服务，删除此节点journalnode元数据存储路径中current文件夹中version以外的文件，启动journalnode。

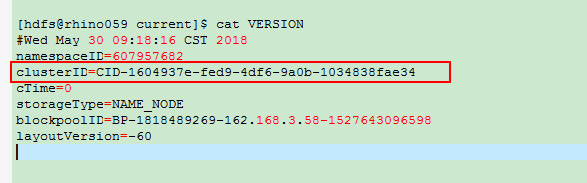
2. 如果某节点的journalnode元数据出现了问题，需要从其他正常节点的journalnode的元数据拷贝过来重启服务，需要注意各个文件的权限。

执行下 hdfs zkfc -formatZK

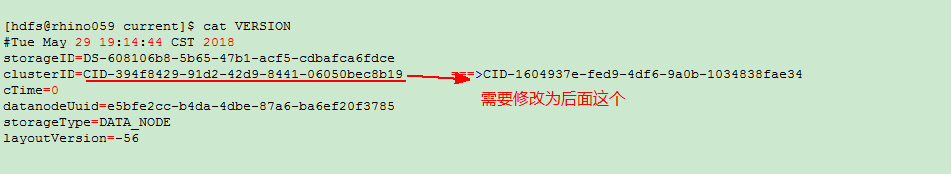
3. datanode节点的clusterID需要同namenode的clusterID保持一致

例如：

namenode节点



datanode节点

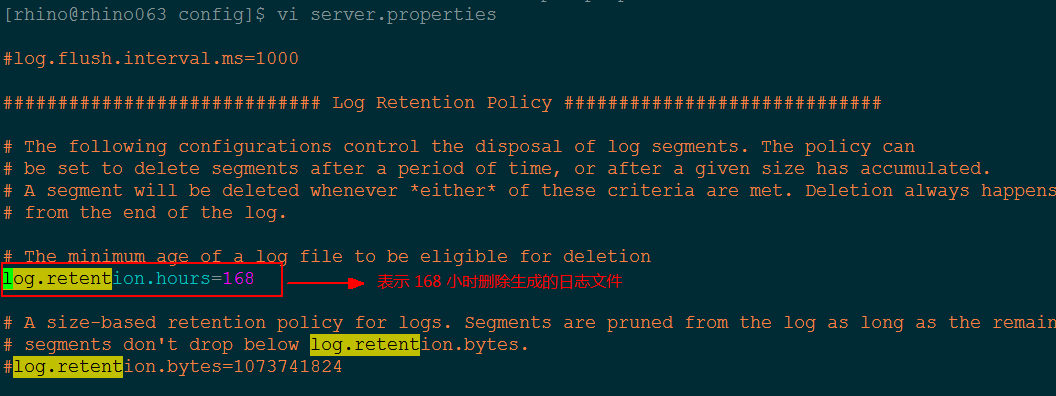


1. Kafka配置数据老化参数设置" log.retention.hours"

配置文件：server.properties

配置参数：log.retention.hours

单位时间：小时



1. 资源队列分配错误

给用户组分配资源队列后，后台tomcat报错：shark.log:java.lang.IllegalArgumentException: java.net.UnknownHostException: rhino060，shark.log:Caused by: java.net.UnknownHostException: rhino060

原因分析：

1)可能后台tomcat-shark配置文件里面主机名配置错误

2)可能jetty/stream下配置文件里面主机名配置错误

解决思路：

1)在shark的class目录中使用命令 grep -rn 'rhino060' \*去遍历所有配置文件，查看配置文件中该报错rhino060主机名所处位置，进行修改并重启tomcat后台

2)在jetty和stream的class目录下使用命令 grep -rn 'rhino060' \*去遍历所有配置文件，查看配置文件中该报错rhino060主机名所处位置，进行修改后，使用提供的wartool工具进行打包./wartool c jetty/stream ，然后重启jetty或者stream

1. 重装kafka前，清理kafka的操作步骤

关键点：

1.1kafka以及前后台进程全部停止

1.2清除kafka配置的log目录下所有日志文件

1.3./zkCli.sh连接zookeeper客户端，删除zookeeper下的注册信息：consumers/admin/config/controller/brokers/controller\_epoch

1.4删除/var/run/kafka/kafkaserver.pid

1.5重启kafka和前后台

1. datanode扩容和缩容实现方式

1.扩容

1)将新增机器安装操作系统、格式化挂载磁盘并添加开机自启动

2)修改操作系统参数

3)添加无密码访问，保证与集群其它机器无密码互连

4)修改/etc/hosts，将新增加的机器IP和主机信息添加进去

5)修改slave文件，将新增机器主机信息添加到slave文件中

6)将hadoop文件夹拷贝到新添加的节点

7)对新添加的节点执行hadoop-daemon.sh start datanode

2.缩容

注意：一台一台执行

1)修改hdfs-site.xml文件

在文件中添加/home/rhino/hadoop/etc/hadoop/excludes配置，指明要缩容的主机名称

将修改后的hdfs-site.xml及excludes文件分别copy至其他机器

2)执行hdfs dfsadmin -refreshNodes刷新hdfs节点

更新hdfs节点状态，观察被排除的datanode节点状态是否变为Decommissioning，Under-replicated blocks逐渐变为0代表恢复正常，副本恢复完成

3)将/home/rhino/hadoop-2.6.0-cdh5.7.0/etc/hadoop/slaves文件中需要缩容的主机剔除，重启集群，完成一个节点的缩容

4)依次类推，依次完成其余节点，注意一个节点一个节点操作，为了保证缩容后数据恢复正常

1. 配置Yarn资源多租户后执行查询sparksql语句时，日志一直提示资源为分配不够问题

17/04/06 16:32:59 INFO cluster.YarnScheduler: Adding task set 0.0 with 1 tasks

17/04/06 16:33:14 WARN cluster.YarnScheduler: Initial job has not accepted any resources; check your cluster UI to ensure that workers are registered and have sufficient resources

17/04/06 16:33:21 INFO thriftserver.SparkExecuteStatementOperation: Running query 'quit' with 96d16e4a-eab1-426e-82cf-a57d9b84b5a6

17/04/06 16:33:21 INFO parse.ParseDriver: Parsing command: quit

NoViableAltException(26@[])

at org.apache.hadoop.hive.ql.parse.HiveParser.statement(HiveParser.java:1071)

at org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:202)

at org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:166)

at org.apache.spark.sql.hive.HiveQl$.getAst(HiveQl.scala:276)

at org.apache.spark.sql.hive.HiveQl$.createPlan(HiveQl.scala:303)

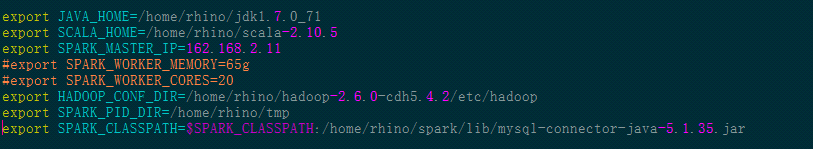
at org.apache.spark.sql.hive.ExtendedHiveQlParser$$anonfun$hiveQl$1.apply(ExtendedHiveQlParser.scala:41)

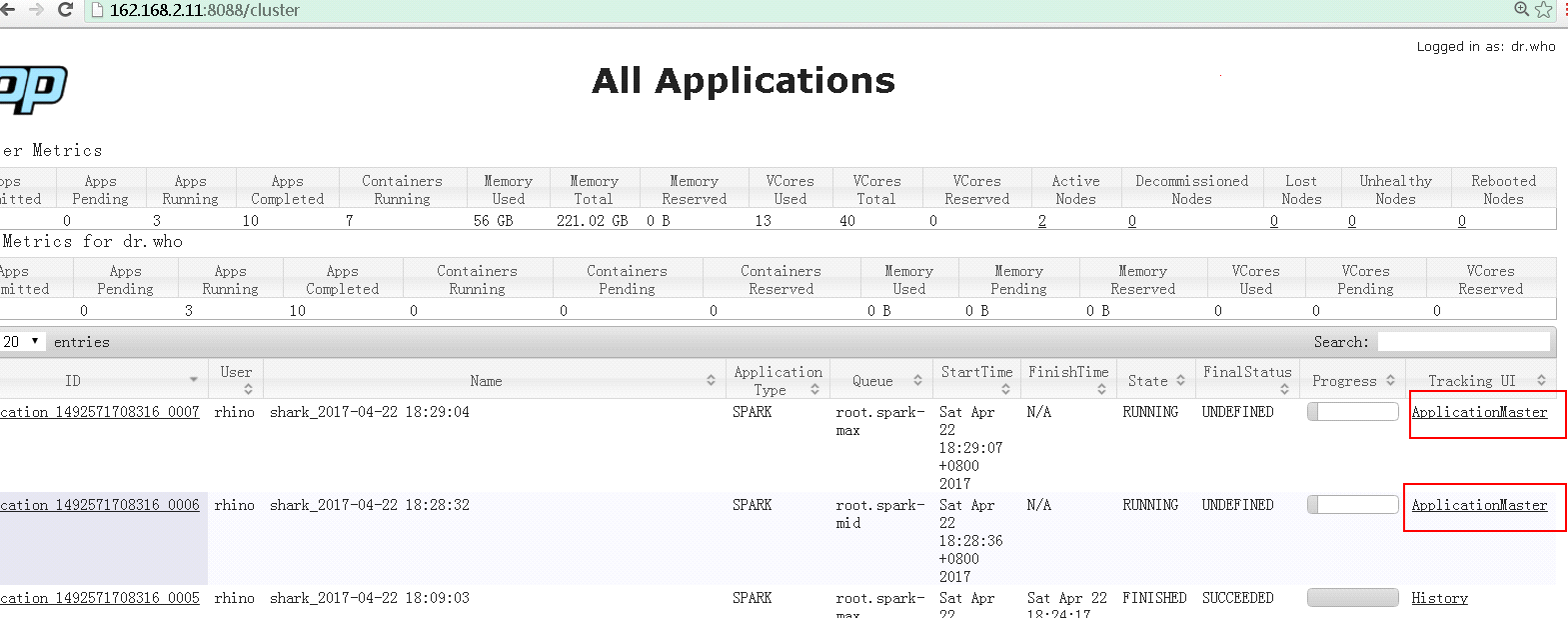
17/04/06 16:33:29 WARN cluster.YarnScheduler: Initial job has not accepted any resources; check your cluster UI to ensure that workers are registered and have sufficient resources

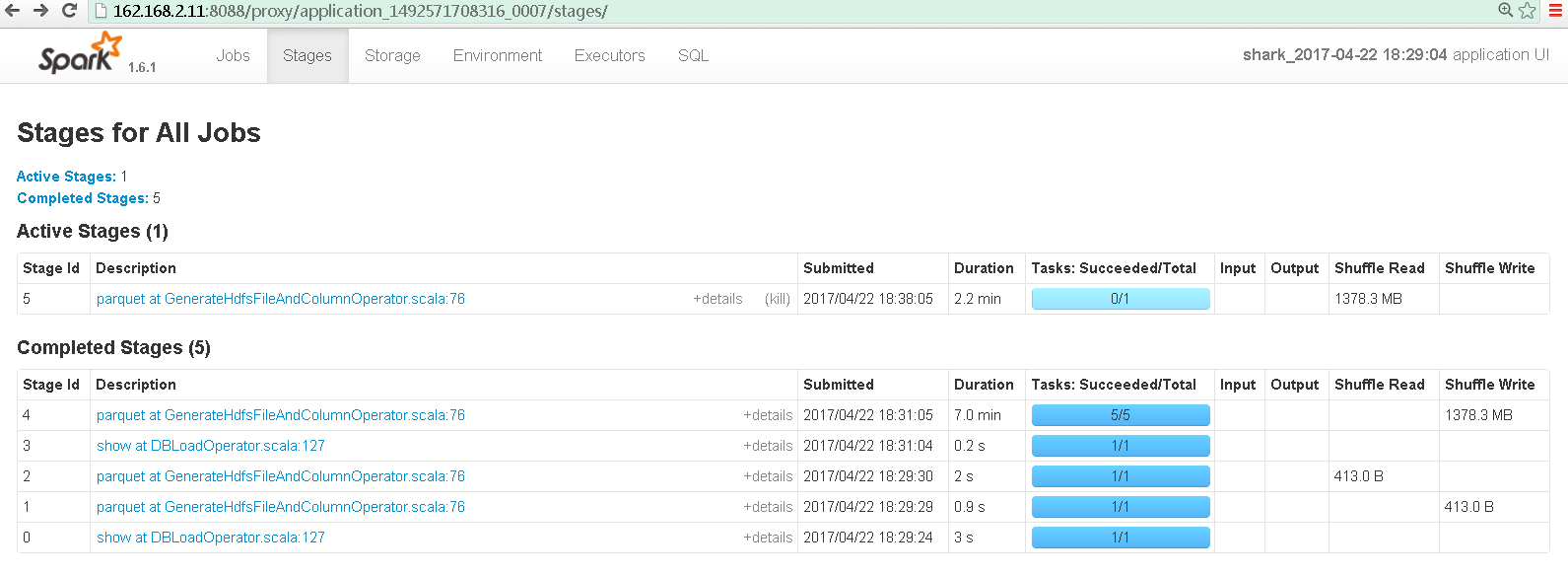
17/04/06 16:33:44 WARN cluster.YarnScheduler: Initial job has not accepted any resources; check your cluster UI to ensure that workers are registered and have sufficient resources

问题解决方法：

使用yarn资源分配后，同之前的处理方式有所区别，不需要启动spark的master、worker进程，即需要将~/spark/conf/spark-env.sh如下配置项注释







1. Spark-sql中执行查询语句提示” spark.kryoserializer.buffer.max value”

0: jdbc:hive2://162.168.2.11:10000> select \* from table5;

Error: org.apache.spark.SparkException: Job aborted due to stage failure: Task 0 in stage 191.0 failed 4 times, most recent failure: Lost task 0.3 in stage 191.0 (TID 210, rhino002): org.apache.spark.SparkException: Kryo serialization failed: Buffer overflow. Available: 0, required: 186. To avoid this, increase spark.kryoserializer.buffer.max value.

at org.apache.spark.serializer.KryoSerializerInstance.serialize(KryoSerializer.scala:299)

at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:240)

at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1145)

at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:615)

at java.lang.Thread.run(Thread.java:745)

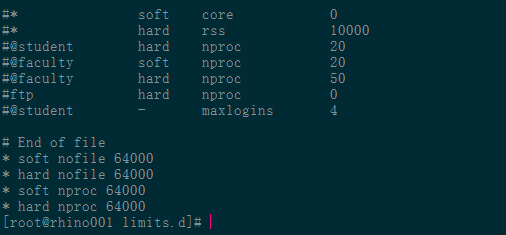
问题解决方法：

启动thrift进程指定的内存小了导致，还有就是这个表查询的数据量特别大，用下面的语句启动thrift进程

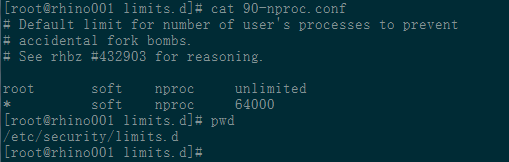
~/spark/sbin/start-thriftserver.sh --executor-memory 5g --total-executor-cores 2 --conf spark.kryoserializer.buffer.max=256m --conf spark.kryoserializer.buffer=64m

1. 环境中执行”java -version”命令提示memory不够的问题

检查文件/etc/security/limits.conf



检查文件etc/security/limits.d/ 90-nproc.conf



1. JDBC接口为impala时，数据仓库落地如何确定是内部表还是外部表，内外部区别如何

1)impala数据仓库落地默认是外部表，通过show create table可以查看有external标识的为外部表

2)区别

1.存储位置：

内部表存储默认spark.sql.warehouse.dir=hdfs://sinovatiocluster/user/hive/warehouse，由Hive自身管理

外部表数据由HDFS管理，外部表存储位置由自己制定，可以通过show create table的location位置确定，location里面指明了存储路径和分属的库和表

2.drop 表区别

删除内部表会直接删除元数据及存储数据，删除外部表仅仅会删除元数据，hdfs存储路径下的数据依然存在

1. 启动impala失败，Cause: org.apache.hive.service.cli.HiveSQLException: Couldn't open transport for localhost:26000 (connect() failed: Connection refused)

原因分析：  
1)impala进程的启动是依赖catalog和statestored，可能是由于这两个进程异常导致

2)26000端口为catalog的服务端口

解决思路：

1)检查依赖进程是否正常，查看impala/lib/impala/logs日志分析具体原因

1. 使用beeline命令连接sparksql提示不能打开会话” Error opening session”

**错误提示：**

[rhino@rhino001 shark-data]$ beeline -u jdbc:hive2://162.168.2.11:10000 -n rhino -p rhino

Connecting to jdbc:hive2://162.168.2.11:10000

16/12/21 14:30:27 INFO jdbc.Utils: Supplied authorities: 162.168.2.11:10000

16/12/21 14:30:27 INFO jdbc.Utils: Resolved authority: 162.168.2.11:10000

16/12/21 14:30:27 INFO jdbc.HiveConnection: Will try to open client transport with JDBC Uri: jdbc:hive2://162.168.2.11:10000

16/12/21 14:31:52 ERROR jdbc.HiveConnection: Error opening session

org.apache.thrift.transport.TTransportException

at org.apache.thrift.transport.TIOStreamTransport.read(TIOStreamTransport.java:132)

at org.apache.thrift.transport.TTransport.readAll(TTransport.java:86)

at org.apache.thrift.transport.TSaslTransport.readLength(TSaslTransport.java:376)

at org.apache.thrift.transport.TSaslTransport.readFrame(TSaslTransport.java:453)

at org.apache.thrift.transport.TSaslTransport.read(TSaslTransport.java:435)

at org.apache.thrift.transport.TSaslClientTransport.read(TSaslClientTransport.java:37)

at org.apache.thrift.transport.TTransport.readAll(TTransport.java:86)

at org.apache.thrift.protocol.TBinaryProtocol.readAll(TBinaryProtocol.java:429)

at org.apache.thrift.protocol.TBinaryProtocol.readI32(TBinaryProtocol.java:318)

at org.apache.thrift.protocol.TBinaryProtocol.readMessageBegin(TBinaryProtocol.java:219)

at org.apache.thrift.TServiceClient.receiveBase(TServiceClient.java:69)

at org.apache.hive.service.cli.thrift.TCLIService$Client.recv\_OpenSession(TCLIService.java:156)

at org.apache.hive.service.cli.thrift.TCLIService$Client.OpenSession(TCLIService.java:143)

at org.apache.hive.jdbc.HiveConnection.openSession(HiveConnection.java:583)

at org.apache.hive.jdbc.HiveConnection.<init>(HiveConnection.java:192)

at org.apache.hive.jdbc.HiveDriver.connect(HiveDriver.java:105)

at java.sql.DriverManager.getConnection(DriverManager.java:571)

at java.sql.DriverManager.getConnection(DriverManager.java:187)

at org.apache.hive.beeline.DatabaseConnection.connect(DatabaseConnection.java:142)

at org.apache.hive.beeline.DatabaseConnection.getConnection(DatabaseConnection.java:207)

**解决办法：**

进入spark安装路径

/home/rhino/spark-1.6.1-bin-hadoop2.6/sbin

执行命令：

./start-thriftserver.sh --total-executor-cores 1 --executor-memory 512M



1. hadoop启动时报错22端口连接不上

原因：现网机器的ssh端口修改，默认端口不是22

解决方法：

/home/rhino/hadoop-2.6.0-cdh5.7.0/etc/hadoop/hadoop-env.sh里增加如下配置后重启hadoop：

export HADOOP\_SSH\_OPTS="-p 5122"

备注:5122是修改后的ssh端口

1. spark启动报错22端口连接不上

原因：现网机器的ssh端口修改，默认端口不是22

解决方法：

/home/rhino/spark/conf/spark-env.sh 中添加如下配置后重启spark:

export SPARK\_SSH\_OPTS="-p 5122"

备注:5122是修改后的ssh端口

1. spark启动报错无法指定被请求的地址

spark日志:

Exception in thread "main" java.net.BindException: 无法指定被请求的地址: Service 'sparkMaster' failed after 16 retries (starting from 7077)! Consider explicitly setting the appropriate port for the service 'sparkMaster' (for example spark.ui.port for SparkUI) to an available port or increasing spark.port.maxRetries.

解决方法:

/home/rhino/spark/conf/spark-env.sh 中添加如下配置后重启spark:

export SPARK\_MASTER\_HOST=172.20.139.202

export SPARK\_LOCAL\_IP=127.0.0.1

备注: 172.20.139.202是机器 IP

1. start-dfs.sh无法启动datanode

错误日志：

2018-12-14 15:24:11,006 WARN org.apache.hadoop.hdfs.server.common.Storage: java.io.IOException: Incompatible clusterIDs in /data03/datanode: namenode clusterID = CID-53428a93-324d-4372-8b22-435abe112b9b; datanode clusterID = CID-839d2a7d-d1b7-4360-b6fe-75d1491cad26

2018-12-14 15:24:11,006 INFO org.apache.hadoop.hdfs.server.common.Storage: Lock on /data04/datanode/in\_use.lock acquired by nodename 8186@rhino215

2018-12-14 15:24:11,006 WARN org.apache.hadoop.hdfs.server.common.Storage: java.io.IOException: Incompatible clusterIDs in /data04/datanode: namenode clusterID = CID-53428a93-324d-4372-8b22-435abe112b9b; datanode clusterID = CID-839d2a7d-d1b7-4360-b6fe-75d1491cad26

2018-12-14 15:24:11,006 INFO org.apache.hadoop.hdfs.server.common.Storage: Lock on /data05/datanode/in\_use.lock acquired by nodename 8186@rhino215

2018-12-14 15:24:11,006 WARN org.apache.hadoop.hdfs.server.common.Storage: java.io.IOException: Incompatible clusterIDs in /data05/datanode: namenode clusterID = CID-53428a93-324d-4372-8b22-435abe112b9b; datanode clusterID = CID-839d2a7d-d1b7-4360-b6fe-75d1491cad26

2018-12-14 15:24:11,007 INFO org.apache.hadoop.hdfs.server.common.Storage: Lock on /data06/datanode/in\_use.lock acquired by nodename 8186@rhino215

2018-12-14 15:24:11,007 WARN org.apache.hadoop.hdfs.server.common.Storage: java.io.IOException: Incompatible clusterIDs in /data06/datanode: namenode clusterID = CID-53428a93-324d-4372-8b22-435abe112b9b; datanode clusterID = CID-839d2a7d-d1b7-4360-b6fe-75d1491cad26

2018-12-14 15:24:11,007 INFO org.apache.hadoop.hdfs.server.common.Storage: Lock on /data07/datanode/in\_use.lock acquired by nodename 8186@rhino215

2018-12-14 15:24:11,007 WARN org.apache.hadoop.hdfs.server.common.Storage: org.apache.hadoop.hdfs.server.common.InconsistentFSStateException: Directory /data07/datanode is in an inconsistent state: cluster Id is incompatible with others.

2018-12-14 15:24:11,008 INFO org.apache.hadoop.hdfs.server.common.Storage: Lock on /data08/datanode/in\_use.lock acquired by nodename 8186@rhino215

2018-12-14 15:24:11,008 WARN org.apache.hadoop.hdfs.server.common.Storage: org.apache.hadoop.hdfs.server.common.InconsistentFSStateException: Directory /data08/datanode is in an inconsistent state: cluster Id is incompatible with others.

2018-12-14 15:24:11,008 FATAL org.apache.hadoop.hdfs.server.datanode.DataNode: Initialization failed for Block pool <registering> (Datanode Uuid unassigned) service to rhino214/192.168.50.214:9020. Exiting.

java.io.IOException: All specified directories are failed to load.

at org.apache.hadoop.hdfs.server.datanode.DataStorage.recoverTransitionRead(DataStorage.java:478)

at org.apache.hadoop.hdfs.server.datanode.DataNode.initStorage(DataNode.java:1394)

at org.apache.hadoop.hdfs.server.datanode.DataNode.initBlockPool(DataNode.java:1355)

at org.apache.hadoop.hdfs.server.datanode.BPOfferService.verifyAndSetNamespaceInfo(BPOfferService.java:317)

at org.apache.hadoop.hdfs.server.datanode.BPServiceActor.connectToNNAndHandshake(BPServiceActor.java:228)

at org.apache.hadoop.hdfs.server.datanode.BPServiceActor.run(BPServiceActor.java:829)

at java.lang.Thread.run(Thread.java:745)

2018-12-14 15:24:11,008 FATAL org.apache.hadoop.hdfs.server.datanode.DataNode: Initialization failed for Block pool <registering> (Datanode Uuid unassigned) service to rhino213/192.168.50.213:9020. Exiting.

java.io.IOException: All specified directories are failed to load.

at org.apache.hadoop.hdfs.server.datanode.DataStorage.recoverTransitionRead(DataStorage.java:478)

at org.apache.hadoop.hdfs.server.datanode.DataNode.initStorage(DataNode.java:1394)

at org.apache.hadoop.hdfs.server.datanode.DataNode.initBlockPool(DataNode.java:1355)

at org.apache.hadoop.hdfs.server.datanode.BPOfferService.verifyAndSetNamespaceInfo(BPOfferService.java:317)

at org.apache.hadoop.hdfs.server.datanode.BPServiceActor.connectToNNAndHandshake(BPServiceActor.java:228)

at org.apache.hadoop.hdfs.server.datanode.BPServiceActor.run(BPServiceActor.java:829)

at java.lang.Thread.run(Thread.java:745)

2018-12-14 15:24:11,010 WARN org.apache.hadoop.hdfs.server.datanode.DataNode: Ending block pool service for: Block pool <registering> (Datanode Uuid unassigned) service to rhino214/192.168.50.214:9020

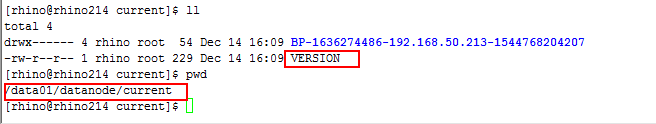
2018-12-14 15:24:11,010 WARN org.apache.hadoop.hdfs.server.datanode.DataNode: Ending block pool service for: Block pool <registering> (Datanode Uuid unassigned) service to rhino213/192.168.50.213:9020

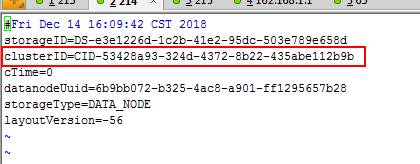
2018-12-14 15:24:11,111 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Removed Block pool <registering> (Datanode Uuid unassigned)

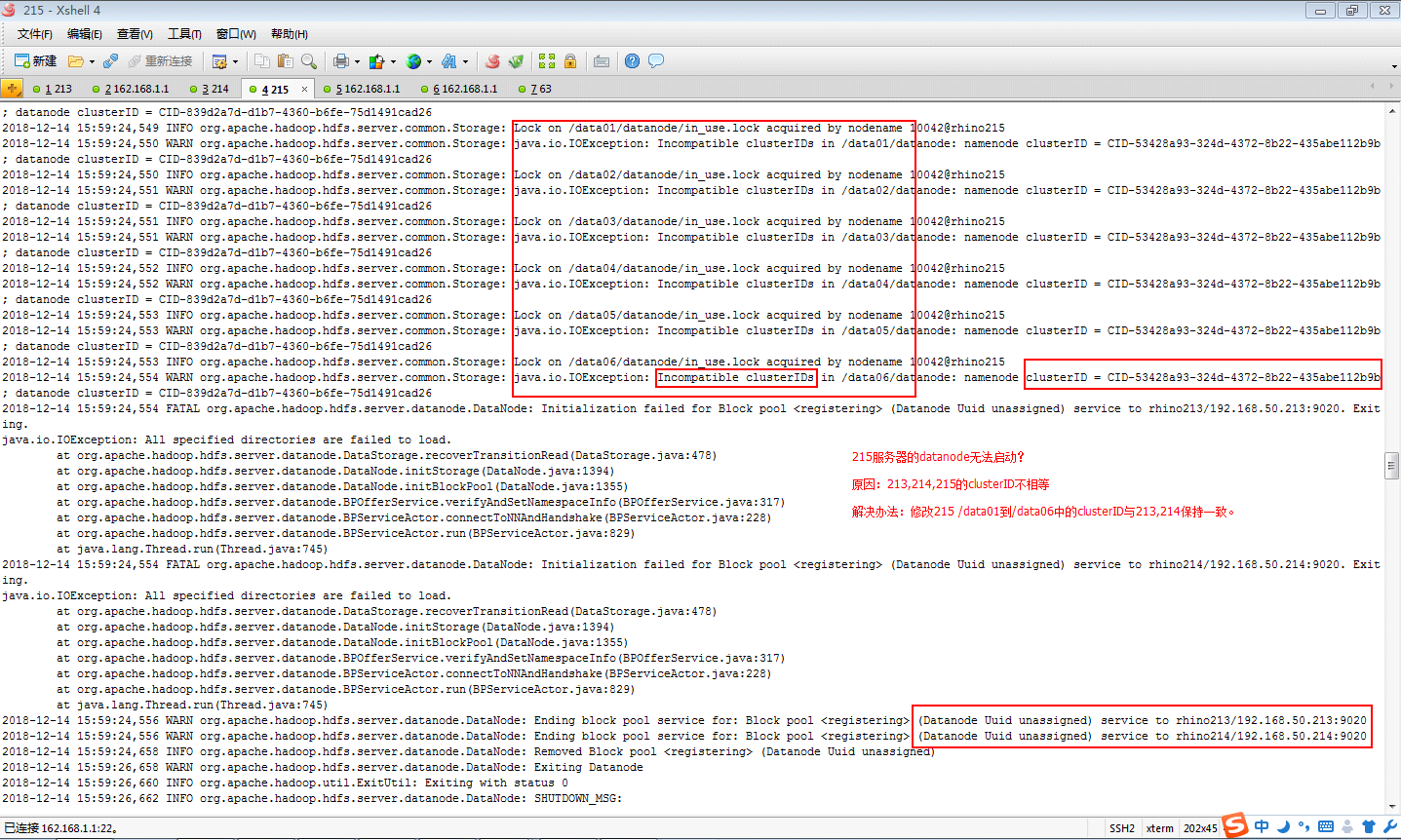
2018-12-14 15:24:13,111 WARN org.apache.hadoop.hdfs.server.datanode.DataNode: Exiting Datanode

2018-12-14 15:24:13,114 INFO org.apache.hadoop.util.ExitUtil: Exiting with status 0

2018-12-14 15:24:13,116 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: SHUTDOWN\_MSG:

原因：如果有三台服务器，213,214,215，那么每台clusterID都要相同





解决方法：如果215的datanode无法启动，需要修改从/data01-/data06中的clusterID的值，路径/data01/datanode/current/VERSION，是三台服务器的clusterID相等。

1. 启动NameNode失败

错误日志：

2018-12-17 14:43:58,580 ERROR org.apache.hadoop.hdfs.server.namenode.NameNode: Failed to start namenode.

java.lang.OutOfMemoryError: GC overhead limit exceeded

at java.util.Arrays.copyOfRange(Arrays.java:3664)

at java.lang.String.<init>(String.java:207)

at java.lang.StringBuilder.toString(StringBuilder.java:407)

at org.apache.hadoop.hdfs.server.namenode.RedundantEditLogInputStream.getName(RedundantEditLogInputStream.java:139)

at org.apache.hadoop.hdfs.server.namenode.FSEditLogLoader.loadFSEdits(FSEditLogLoader.java:139)

at org.apache.hadoop.hdfs.server.namenode.FSImage.loadEdits(FSImage.java:835)

at org.apache.hadoop.hdfs.server.namenode.FSImage.loadFSImage(FSImage.java:690)

at org.apache.hadoop.hdfs.server.namenode.FSImage.recoverTransitionRead(FSImage.java:281)

at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.loadFSImage(FSNamesystem.java:1063)

at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.loadFromDisk(FSNamesystem.java:767)

at org.apache.hadoop.hdfs.server.namenode.NameNode.loadNamesystem(NameNode.java:609)

at org.apache.hadoop.hdfs.server.namenode.NameNode.initialize(NameNode.java:670)

at org.apache.hadoop.hdfs.server.namenode.NameNode.<init>(NameNode.java:838)

at org.apache.hadoop.hdfs.server.namenode.NameNode.<init>(NameNode.java:817)

at org.apache.hadoop.hdfs.server.namenode.NameNode.createNameNode(NameNode.java:1538)

at org.apache.hadoop.hdfs.server.namenode.NameNode.main(NameNode.java:1606)

2018-12-17 14:43:58,583 INFO org.apache.hadoop.util.ExitUtil: Exiting with status 1

2018-12-17 14:43:58,584 INFO org.apache.hadoop.hdfs.server.namenode.NameNode:

解决办法：在hadoop-env.sh文件中增加限制内存参数 -Xms6000m -Xmx6000m。然后重启hdfs

vi /home/rhino/hadoop-2.6.0-cdh5.7.0/etc/hadoop/hadoop-env.sh

|  |
| --- |
| export HADOOP\_NAMENODE\_OPTS="-Dhadoop.security.logger=${HADOOP\_SECURITY\_LOGGER:-INFO,RFAS} -Dhdfs.audit.logger=${HDFS\_AUDIT\_LOGGER:-INFO,NullAppender} -Xms6000m -Xmx6000m $HADOOP\_NAMENODE\_OPTS" |

（PS：如果SecondaryNameNode也启动失败，检查一下端口50090是否被占用

netstat -anp | grep 50090）

1. jps命令出现不可用的进程 “22828 -- process information unavailable ”

解决方法：进入tmp/hsperfdata\* 目录下进行删除进程号即可。

1. 在impala里执行命令报错如下：

[rhino032:21000] test> select \* from test3;

Query: select \* from test3

Query submitted at: 2019-01-23 16:08:30 (Coordinator: http://rhino032:25000)

Query progress can be monitored at: http://rhino032:25000/query\_plan?query\_id=8f4c041e6765c036:ab8ef79e00000000

WARNINGS: TransmitData() to 162.168.3.2:27000 failed: Invalid argument: Client connection negotiation failed: client connection to 162.168.3.2:27000: unable to find SASL plugin: PLAIN

解决方法：

root用户下执行如下命令：

|  |
| --- |
| [root@rhino032 ~]# yum install gcc python-devel cyrus-sasl\* -y |

然后重启impala后解决。