

## Hive 常见问题维护手册 V1.0

# Hive 常见问题维护手册 V1.0

文档版本 01  
发布日期 2016-03-31

华为技术有限公司



**版权所有 © 华为技术有限公司 2016。 保留一切权利。**

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 商标声明



和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

# 华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编： 518129

网址： <http://www.huawei.com>

客户服务邮箱： [support@huawei.com](mailto:support@huawei.com)

客户服务电话： 4008302118

---

# 目 录

---

HIVE.....	4
1、基本概念.....	4
【Hive架构概述】 .....	4
【日志概述】 .....	5
Hive运行日志。 .....	5
【日志查看】 .....	6
【Hive_LOG_00001】 遇到错误怎么知道查看哪个日志? .....	6
【Hive_LOG_00002】 各个日志里都存放了什么信息? .....	15
【Hive_LOG_00003】 怎么查看HiveServer日志? .....	16
【Hive_LOG_00004】 有哪些快速定位搜索关键字? .....	18
【实战案例】 如何通过日志分析,快速定位问题? .....	19
2、常见问题.....	22
【参数及配置类】 .....	22
[Hive-10001]执行set命令的时候报Cannot modify xxx at runtime .....	22
[Hive-10002]怎样在Hive提交任务的时候指定队列? .....	23
[Hive-10003]客户端怎么设置Map/Reduce内存? .....	23
[Hive-10004]如何在导入表时指定输出的文件压缩格式 .....	24
[Hive-10005]desc描述表过长时,无法显示完整 .....	25
[Hive-10006]增加分区列后再insert数据显示为NULL .....	26
【权限管理类】 .....	26
[Hive-20001]创建新用户,执行查询时报无权限。 .....	26
[Hive-20002]执行SQL提交任务到指定队列报错。 .....	29
[Hive-20003]执行load data inpath命令报错 .....	29
[Hive-20004]执行load data local inpath命令报错 .....	30
[Hive-20005]执行insert overwrite directory 命令报错 .....	31
[Hive-20006]执行insert overwrite local directory 命令报错 .....	31
[Hive-20007]执行create external table报错 .....	32
[Hive-20008]在beeline客户端执行dfs -put命令报错 .....	32
[Hive-20009]执行set role admin报无权限 .....	33
[Hive-20010]通过beeline创建UDF时候报错 .....	33
【安装启动类】 .....	34
[Hive-30001]Hive启动失败报Cannot find hadoop installation .....	34
[Hive-30002]Hive启动失败问题的原因有哪些? .....	34
【监控告警类】 .....	35

---

[Hive-40001]Hive状态为Bad问题总结 .....	35
[Hive-40002]Hive服务状态为Partially Healthy总结 .....	35
[Hive-40003]Hive服务状态为Unknown总结 .....	35
[Hive-40004]Hiveserver 或者Metastore 实例的健康状态为unknown .....	36
[Hive-40005]Hiveserver 或者Metastore 实例的健康状态为Concerning .....	36
[Hive-40006]Hive服务健康状态和Hive实例健康状态的区别 .....	36
[Hive-40007]Hive中的告警有哪些以及触发的场景 .....	36

# HIVE

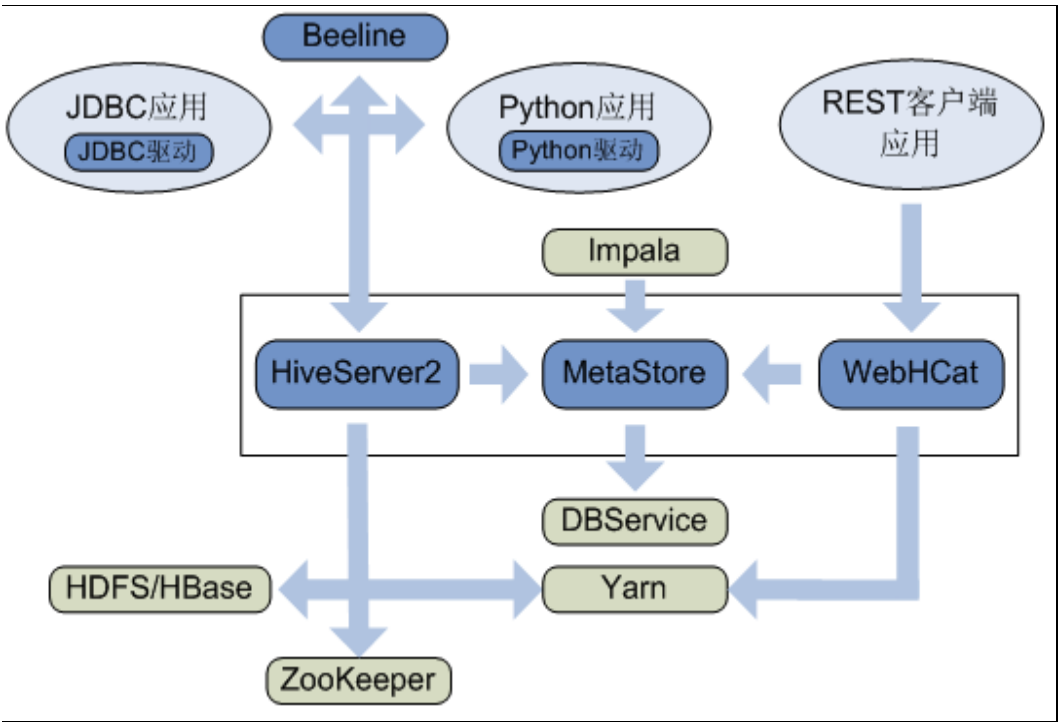
## 1、基本概念

本章简要介绍 Hive 基本概念以及 Hive 日志存放路径，以及常见问题的日志查看方法。

### 【Hive 架构概述】

Hive 是建立在 Hadoop 上的数据仓库框架，提供类似 SQL 的 Hive QL 语言操作结构化数据，其基本原理是将 HQL 语言自动转换成 MapReduce 任务，从而完成对 Hadoop 集群中存储的海量数据进行查询和分析。

架构图如下：



Hive 主要特点如下：

- 海量结构化数据分析汇总。
- 将复杂的 MapReduce 编写任务简化为 SQL 语句。
- 灵活的数据存储格式，支持 JSON，CSV，TEXTFILE，RCFILE，SEQUENCEFILE 这几种存储格式。

Hive 各组件说明：

名称	说明
HiveServer	一个集群内可部署多个 HiveServer，负荷分担。对外提供 Hive 数据库服务，将用户提交的 HQL 语句进行编译，解析成对应的 Yarn 任务或者 HDFS 操作，从而完成数据的提取、转换、分析。

MetaStore	<ul style="list-style-type: none"> <li>一个集群内可部署多个 MetaStore，负荷分担。提供 Hive 的元数据服务，负责 Hive 表的结构和属性信息读、写、维护和修改。</li> <li>提供 Thrift 接口，供 HiveServer、Impala、WebHCat 等 MetaStore 客户端来访问，操作元数据。</li> </ul>
WebHCat	一个集群内可部署多个 WebHCat，负荷分担。提供 Rest 接口，通过 Rest 执行 Hive 命令，提交 MapReduce 任务。
Hive 客户端	包括人机交互命令行 Beeline、提供给 JDBC 应用的 JDBC 驱动、提供给 Python 应用的 Python 驱动、提供给 Mapreduce 的 HCatalog 相关 JAR 包。

## 【日志概述】

Hive 日志包括两类，运行日志、审计日志；同时如果遇到在执行 MR 过程中的失败，还需要去看 MR 日志。【详情请看 MR 日志查看相关章节】

日志路径：

Hive 运行日志位于 /var/log/Bigdata/hive 下；

Hive 审计日志位于 /var/log/Bigdata/audit/hive 下；

各日志功能如下表：

表 1 Hive 日志列表		
日志类型	日志文件名	描述
HiveServer 运行日志	hive.log	<b>Hive 运行日志。</b>
	hive.xxxx-xx-xx.log.zip	运行日志的历史归档，以归档时的日期作为标识。如，hive.2016-02-29_13-08-24.[7].log.zip 表示归档时间点为：2016-02-29_13-08-24。
	hive-omm-gc.log	HiveServer GC 日志。
	hadoop-omm-jar-192-168-1-18.out	控制台日志，程序中以 System.out 输出的信息均会在此日志中体现
	cleanupDetail.log	Hive 在启动、停止过程中相关日志。如遇到启动失败或停止 Hive 失败等问题，相关信息均会体现在这写日志文件内。
	prestartDetail.log	
	startDetail.log	
	stopDetail.log	
MetaStore	metastore.log	<b>MetaStore 运行日志。</b>

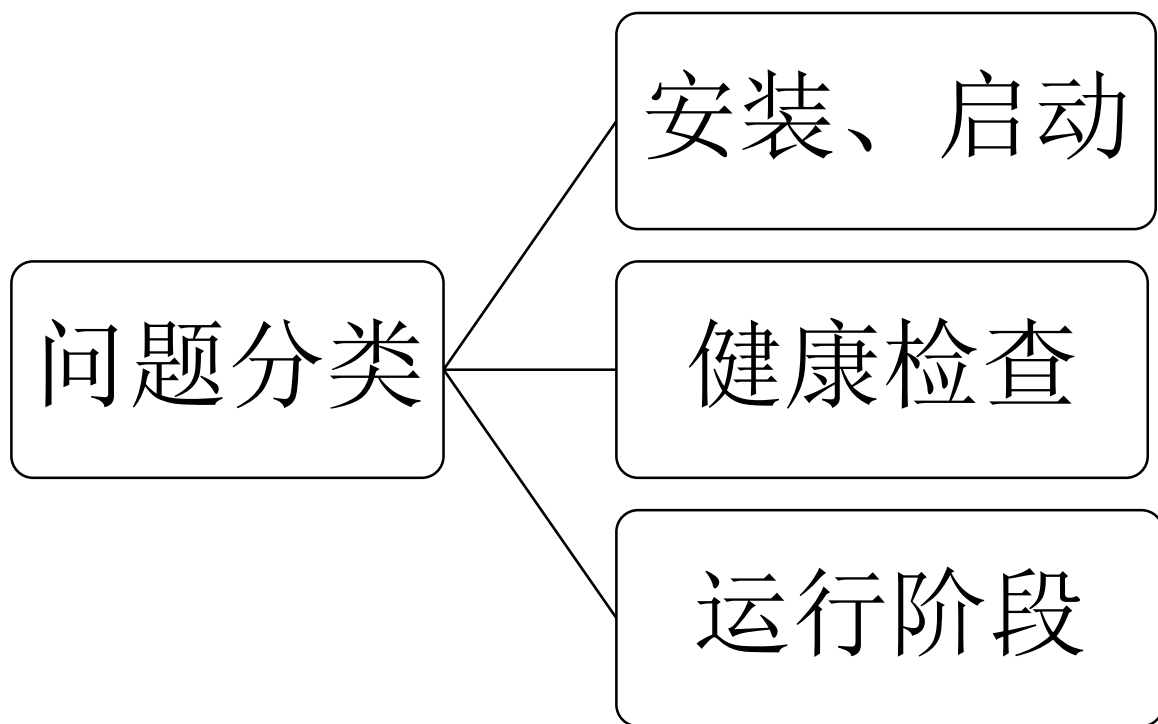
表 1 Hive 日志列表		
日志类型	日志文件名	描述
运行日志	metastore.xxxx-xx-xx.log.zip	运行日志的历史归档, 以归档时的日期作为标识。如, metastore.2016-02-29_13-08-24.[7].log.zip 表示归档时间点为: 2016-02-29_13-08-24。
	hive-omm-gc.log.0.current	MetaStore GC 日志。
	postinstallDetail.log	MetaStore 在启动、停止过程中相关日志。如遇到启动失败或停止 MetaStore 失败等问题, 相关信息均会体现在这写日志文件内。
	prestartDetail.log	
	startDetail.log	
	stopDetail.log	
HiveServer 审计日志	hive-audit.log	HiveServer 审计日志。
	hive-audit.2016-03-09_17-57-12.[1].log.zip	HiveServer 审计日志归档文件。
MetaStore 审计日志	metastore-audit.log	MetaStore 审计日志。
	metastore-audit.2016-03-03_22-02-31.[3].log.zip	MetaStore 审计日志归档文件。

## 【日志查看】

### 【Hive\_LOG\_00001】 遇到错误怎么知道查看哪个日志？

#### 【问题定界】

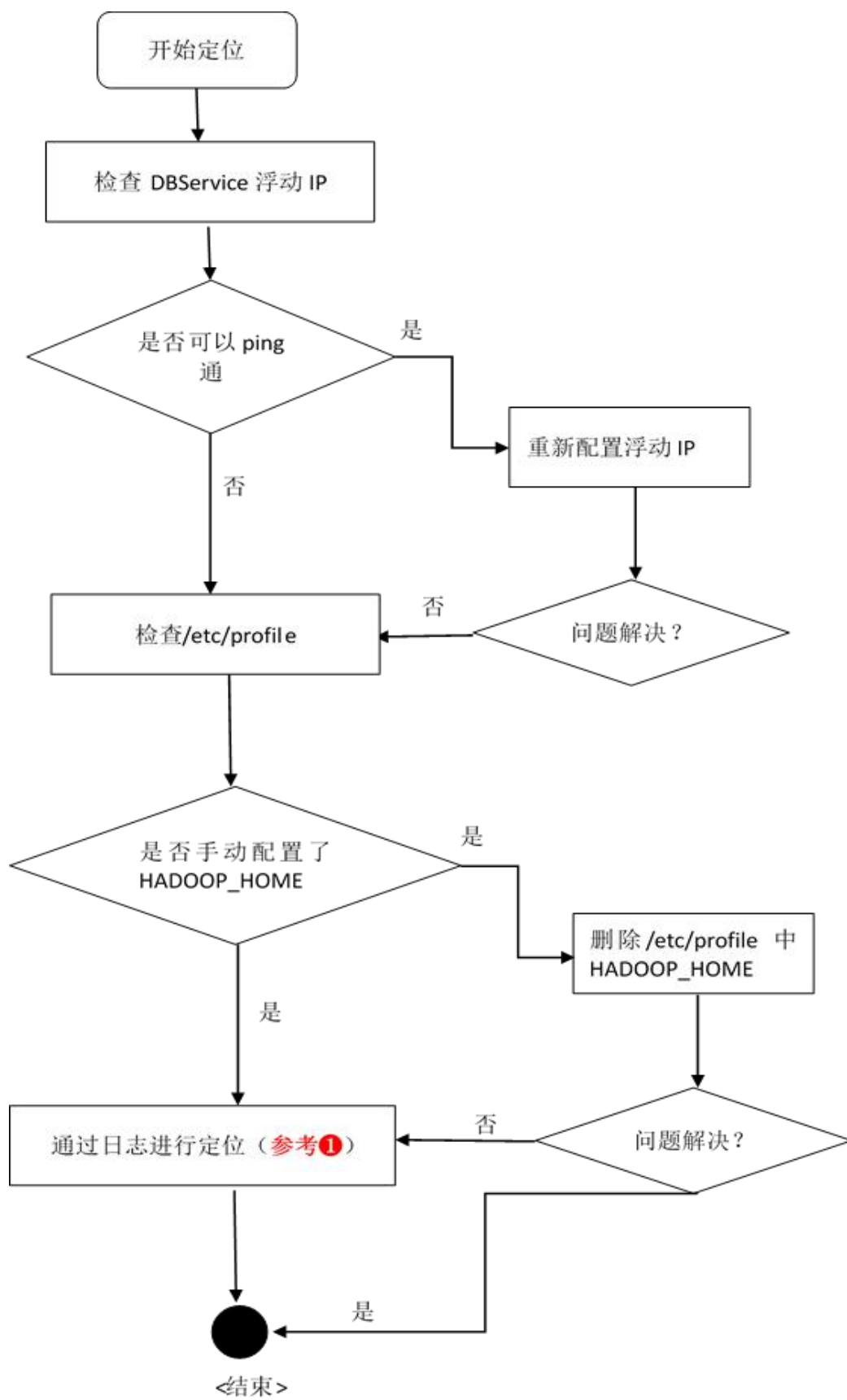
当遇到问题时, 首先需要确认是在哪一个步骤出的问题。Hive 的问题可以简单分为以下几个大类:



#### 【安装、启动过程问题定位思路】

Hive 安装启、启动时一个连续的过程；当我们在安装、启动过程中遇到问题一般在 OM 页面会有提示错误信息。当我们在此过程中遇到错误可参考以下排查步骤：





## 参考①:

遇到此类问题，需要综合查看以下日志，根据日志中提供的线索进行定位：

/var/log/Bigdata/hive/metastore/下：

metastore.log 、 prestartDetail.log 、 startDetail.log

/var/log/Bigdata/hive/hiveserver/下：

hive.log、 prestartDetail.log、 startDetail.log

日志中常见错误及解决方案：

错误信息 1:

```
org.postgresql.util.PSQLException: FATAL: no pg_hba.conf entry for host "160.164.0.146",
database "HIVEMETA"
```

解决方案：

登录 dbservice 所在的机器上：

1. 修改 /srv/BigData/dbdata\_service/data/pg\_hba.conf，在文件最后面追加 host hivemeta hive 0.0.0.0/0 sha256 （hivemeta 为使用的修改文件 pg\_hba.conf 加入授信
2. 执行 source /opt/huawei/Bigdata/dbservice/.dbservice\_profile，再执行 gs\_ctl -D \$GAUSSDATA reload # 使修改后的配置生效。
3. 修改后重启 metastore 服务。

错误信息 2:

```
dbservice in a read-only transactionsql
```

解决方案：

遇到此问题，一般原因是由于 metastore 连到的 dbservice 不是 active 的（主节点），造成这个原因可能是 dbservice 正在切换主备。等 dbservie 好了后，重启 metastore。

其他信息排查：

```
a) source /opt/huawei/Bigdata/dbservice/.dbservice_profile
b) gsql -h 192.168.xx.xx (DB 的浮动 ip) -p 20051 -d hivemeta -U hive -W HiveUser@
```

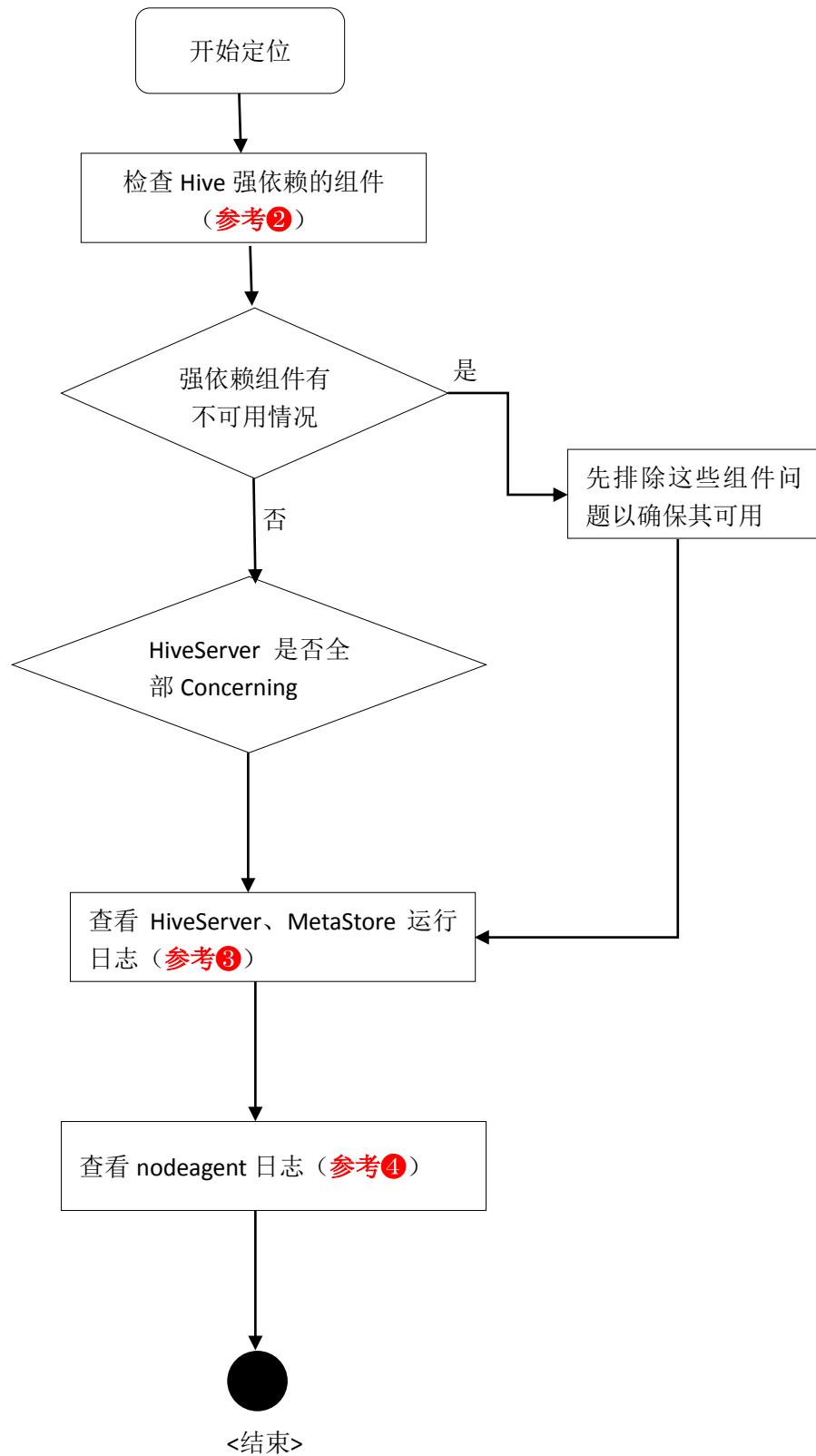
执行以上步骤，确保 DBService 可以联通，并且 hivemeta 数据库可以正常访问。如果以上步骤连不同，请参考 DBService 常见问题定位指导。

## 【健康检查问题定位思路】

健康检查一般出问题就表现为在 OM 页面出现服务不可用，在 OM 页面一般呈现为以下几种形式：

- Bad
- 双Active(C30版本为主备模式，出现双主均为不正常)
- Concerning
- UnKnown

定位思路请参考下图：



**参考②：**

Hive 强依赖组件:

- Ldap
- KrbServer
- Zookeeper
- HDFS
- YARN
- DBService

### 参考③:

通常, 通过 HiveServer 和 MetaStore 运行日志可以查看到有用的信息, 常见到位安全认证问题和 DBService 问题。

安全认证问题:

1. 进入路径/opt/huawei/Bigdata/hive/hive-1.1.0/bin
2. 查看是否 hive.keytab 是否存在
3. 查看 hive.keytab 是否有读写权限
4. hive.keytab 是否可用, 判断方法如下:

```
[root@192.168-1-18 bin]# kinit -kt hive.keytab hive/hadoop.hadoop.com
[root@192.168-1-18 bin]# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: hive/hadoop.hadoop.com@HADOOP.COM
Valid starting    Expires          Service principal
03/18/16 15:19:10 03/19/16 15:19:10 krbtgt/HADOOP.COM@HADOOP.COM
```

连接 DBService 问题:

1. 进入 MetaStore 所在节点
2. 查看是否 hive.keytab 是否存在

执行以下命令, 确定可以连上 DBService

```
a) source /opt/huawei/Bigdata/dbservice/.dbservice_profile
b) gsql -h 192.168.xx.xx (DB 的浮动 ip) -p 20051 -d hivemeta -U hive -W HiveUser@
```

### 参考④:

如通过上面步骤均无异常, 那么很可能就是流程机制类的问题了。

1. 首先确认各个 HiveServer 的 nodeagent 的 monitor.log 日志, 是否存在取到非零状态码的日志,

路径: /var/log/Bigdata/nodeagent/monitorlog/monitor.log

```
2015-11-12 12:00:08,153 INFO [ServiceCheckThread1_10_HiveServer] errordes:
15001
com.huawei.hadoop.om.agent.monitoring.serviceChecker.JmxServiceStatusChecker.j
mxCheckServiceStatus(JmxServiceStatusChecker.java:90)
```

2. 如果存在, 检查对应的 hive.log, 肯定会出现类似下面的失败日志, 即可针对性处理对应问题。

```
2015-11-10 12:23:56,756 | ERROR | HiveServiceHealthChecker | failed to connection
to hiveserver. |
org.apache.hadoop.hive.om.monitor.HiveServiceHealthChecker.checkHiveServer(Hive
```

---

```
ServiceHealthChecker.java:145)
```

3. 查看 Hive 对应的 JMX 服务和健康检查线程是否正常。该线程的相关日志在 hive.log 中:

```
2015-11-10 12:23:56,756 | ERROR | HiveServiceHealthChecker | failed to connection  
to hiveserver.  
org.apache.hadoop.hive.om.monitor.HiveServiceHealthChecker.checkHiveServer(Hive  
ServiceHealthChecker.java:145)
```

4. 确认 nodeagent 的健康检查双线程是正常运行的: 开启 nodeagent DEBUG 级别日志, 查看是否存在以下运行记录。如果不存在, 可能是 agent 的运行或者配置问题。

```
2015-11-12 11:55:08,121 DEBUG [ServiceCheckThread1_10_HiveServer]  
objectName:org.apache.hadoop.hive.om.hiveservice:type=HiveServiceStatusMBean,s  
erviceKey:ServiceStatusCode  
com.huawei.hadoop.om.agent.monitoring.serviceChecker.JmxServiceStatusChecker.j  
mxCheckServiceStatus(JmxServiceStatusChecker.java:66)
```

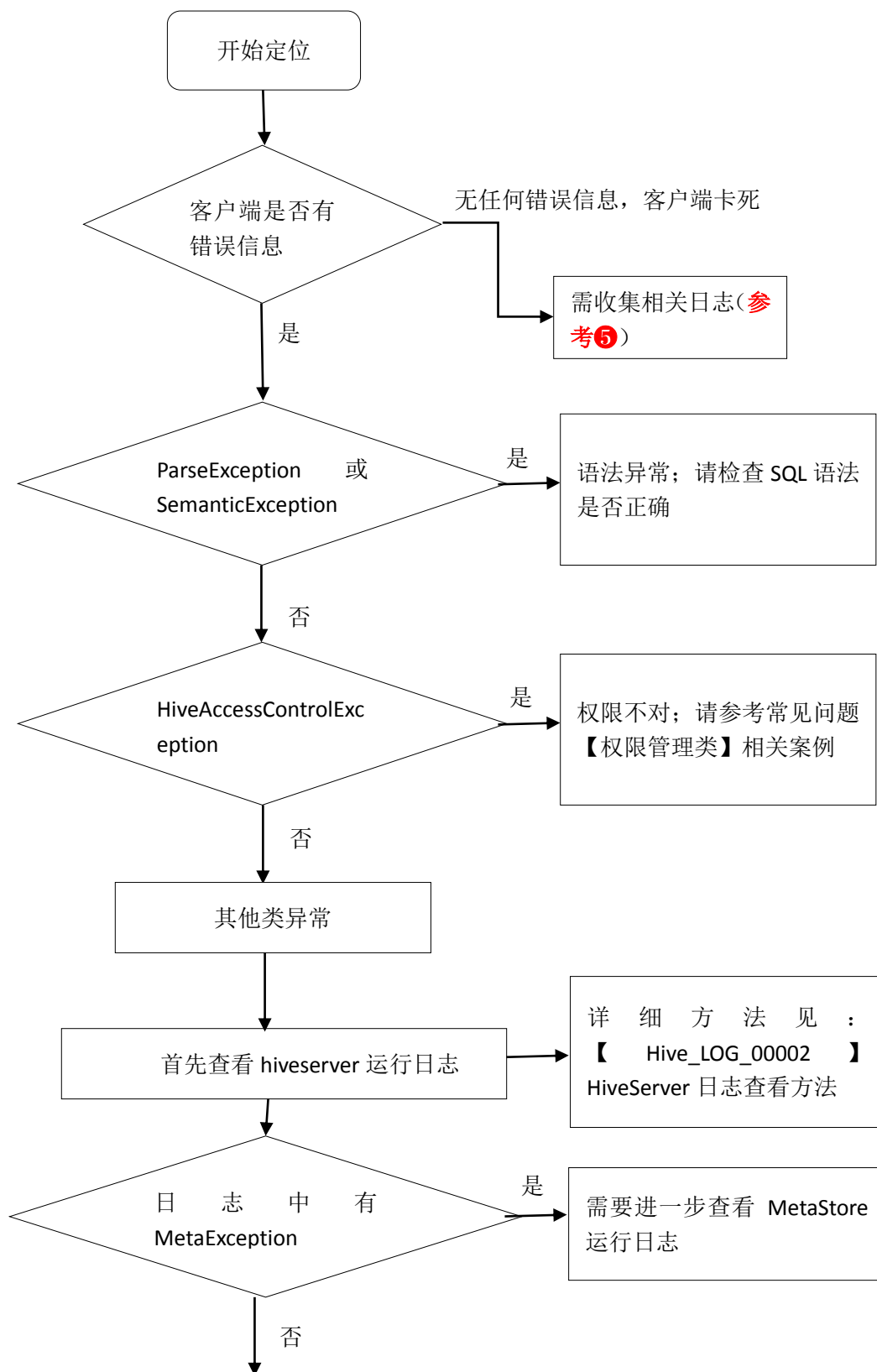
5. 确认健康检查双线程是否正常启动的, 查看 nodeagent 的 agent.log 日志, 是否有正常初始化日志。

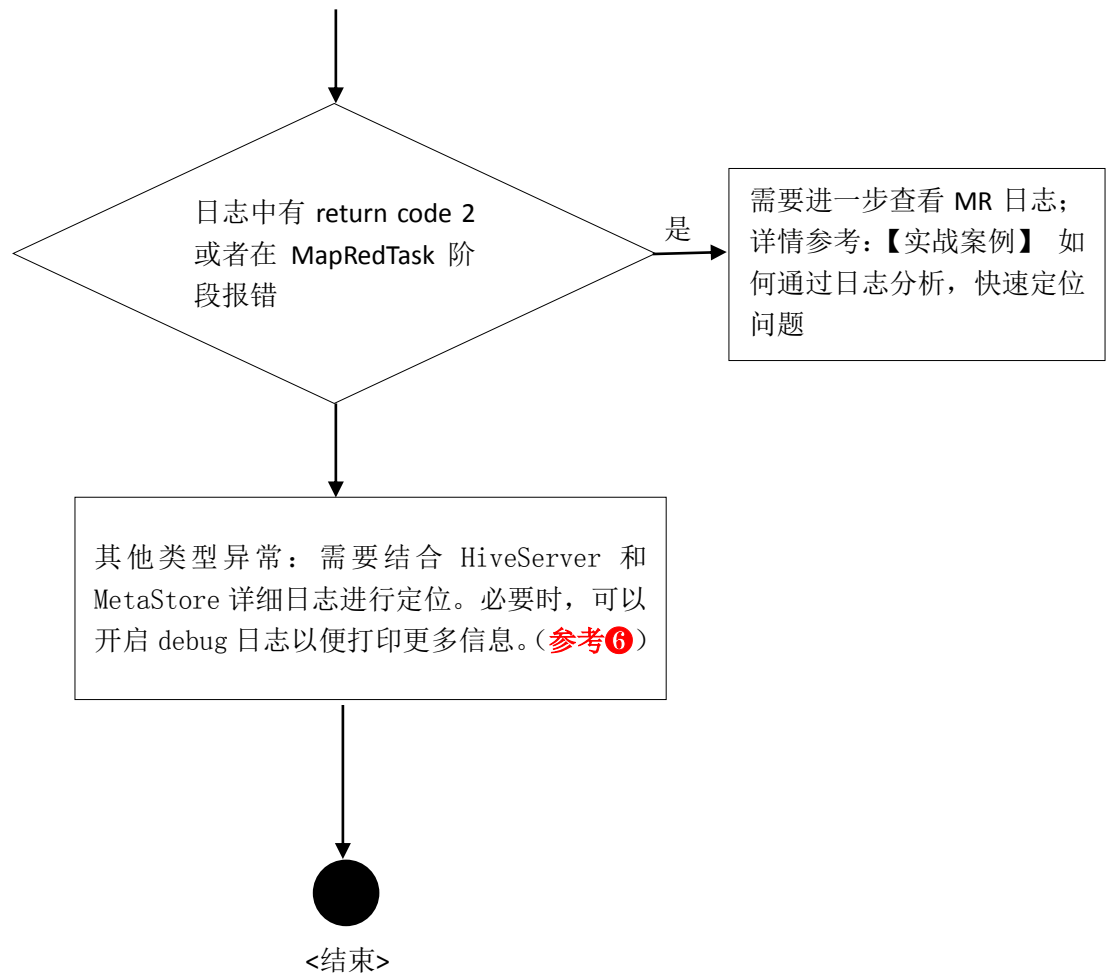
```
2015-11-12 12:01:33,297 INFO [HealthMonitorThread1_10_HiveServer]  
1_10_HiveServer state changed from UNKNOWN to GOOD  
com.huawei.hadoop.om.agent.services.HealthMonitorService$ProcessHealthMonitor  
Thread.changeProcessState(HealthMonitorService.java:631)
```

6. 如果上述均正常, 则需要收集对应 controller, nodeagent 的日志, 寻求 OM 维护的协助。

### 【运行阶段问题定位思路】

运行阶段问题占到 Hive 问题的 90%以上, 遇到的问题种类繁多, 下图将整理这类问题的大致定位思路以给出方向:





### 参考⑥:

遇到此类无任何错误码返回的卡死问题, 一般处理比较困难, 需要收集以下日志结合日志分析:

Beeline 的 jstack 信息, 以确定是否是客户端卡死;

1. 首先, 执行下述命令, 以找到 beeline 的 pid, 即 28487

```
[root@192-168-1-18 monitorlog]# ps -ef|grep beeline
root          28487  28427    0 15:54 pts/0        00:00:05 java -cp
/opt/duan/client/Hive/Beeline/conf:/opt/duan/client/Hive/Beeline
```

2. 每各 20 秒左右执行下述命令, 并重定向到不同的文件。采集 3 次以上的信息。

```
[root@192-168-1-18 monitorlog]# jstack -F 28487 > /tmp/1.txt
```

3. 查看 HiveServer 审计日志。如果审计日志里并没有收到请求, 那么问题很可能是卡住在客户端。
4. HiveServer 运行日志以及 jstack 信息
5. MetaStore 运行日志以及 jstack 信息
6. HiveServer 以及 MetaStore GC 日志

### 参考⑥:

1. 进入 HiveServer 所在节点。（两个节点均需要修改）
2. 修改 /opt/huawei/Bigdata/etc/\*\_HiveServer/hive-log4j.properties 文件中 log4j.rootLogger = INFO,DRFA,EventCounter，将 INFO 改成 DEBUG

## 【Hive\_LOG\_00002】 各个日志里都存放了什么信息？

### 【审计日志】

首先，对于审计日志来说，记录了某个时间点某个用户从哪个 IP 发起对 HiveServer 或者 MetaStore 的请求以及记录执行的语句是什么。

如下的 HiveServer 审计日志，表示在 2016-02-01 14:51:22 用户 user\_chen 向 HiveServer 发起了 show tables 请求，客户端 IP 为 192.168.1.18

```
2016-02-01 14:51:22,335 | INFO | HiveServer2-Handler-Pool: Thread-37815 | UserN  
ame=user_chen | UserIP=192.168.1.18 | Time=2016/02/01 14:51:22 | Operati  
on=ExecuteStatement | stmt={show tables} | Resource= | Result= Detail=  
| org.apache.hive.service.cli.thrift.ThriftCLIService.logAuditEvent(ThriftCLISer  
vice.java:350)
```

在如,下面 MetaStore 审计日志表示在 2016-01-29 11:31:15 用户 hive 向 MetaStore 发起 shutdown 请求，客户端 ip 为 192.168.1.18。

```
2016-01-29 11:31:15,451 | INFO | pool-6-thread-70648 | ugi=hive/hadoop.hadoop.c  
om@HADOOP.COM | IP=192.168.1.18 | cmd=Shutting down the object store...  
| org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.logAuditEvent(HiveM  
etaStore.java:375)
```

通常情况下，审计日志对我们定位实际错误信息并无太大帮助。但在遇到诸如下述类问题时，需要查看审计日志：

- 1) 如客户端发起请求，但是迟迟未得到响应。由于不确定到底是任务是卡在客户端还是服务端，可以通过审计日志查看。如果审计日志根本没有相关信息，那么说明卡死在客户端；如审计日志有相关打印，那么就需要去运行日志里看到底程序卡在哪一步了。
- 2) 查看指定时间段的任务请求个数。可通过审计日志查看在指定时间段到底有多少个请求。

### 【HiveServer 运行日志】

简言之，HiveServer 负责接收客户端请求（SQL 语句），然后编译、执行（提交到 YRAN 或跑 local MR）、与 MetaStore 交互获取元数据信息等。HiveServer 运行日志记录了一个 SQL 完整的执行过程。

通常情况下，当我们遇到 SQL 语句运行失败，我们首先要看的就是 HiveServer 运行日志。



## 【MetaStore 运行日志】

通常情况下，当我们遇到查看 HiveServer 运行日志时，如遇到 MetaException 或者连接 MetaStore 失败，我们就需要去看 MetaStore 运行日志了。

## 【GC 日志查看】

HiveServer 和 MetaStore 均有 GC 日志，当我们遇到 GC 问题可以 GC 日志以快速定位是否是 GC 导致。如，当遇到 HiveServer 或 MetaStore 频繁重启就需要去看下对应的 GC 日志了。

## 【Hive\_LOG\_00003】 怎么查看 HiveServer 日志？

我们可以简单的将一个 SQL 语句在 Hive 中的执行分为**编译阶段**和**执行阶段**。

### 【编译阶段日志查看】

如下所示，编译阶段的开始标志关键字是“Parsing command:”

结束关键字：“Parsing Completed:”。

如果我们在编译阶段报错，那么日志就位于上述开始与结束之间。

值得注意的是，对于一个 SQL 语句我们会在日志看到两个对应的线程号。编译阶段会看到类似：**HiveServer2-Handler-Pool: Thread-64**；执行阶段会看到类似 **HiveServer2-Background-Pool: Thread-9502**。由于 HiveServer 里存放所有任务的运行日志，所以我们通常需要根据线程号进行搜索。

```
2016-02-04 15:30:37,536 | INFO | HiveServer2-Handler-Pool: Thread-64 | <PERFLOG
method=parse from=org.apache.hadoop.hive.ql.Driver> |
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogBegin(PerfLogger.java:121)

2016-02-04 15:30:37,536 | INFO | HiveServer2-Handler-Pool: Thread-64 |

Parsing command: select count(*) from src |

org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:185)

2016-02-04 15:30:37,538 | INFO | HiveServiceHealthChecker | Will try to open client
transport with JDBC Uri:
jdbc:hive2://160.138.1.18:21066/default;principal=hive/hadoop.hadoop.com@HADOOP.C
OM;sasl.qop=auth-conf;auth=kerberos;inner.client.marker=1454571037511 |
org.apache.hive.jdbc.HiveConnection.openTransport(HiveConnection.java:253)
.....此处省略 N 行
2016-02-04 15:30:37,549 | INFO | HiveServer2-Handler-Pool: Thread-64 | Parse
Completed | org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:206)
```

```

2016-02-04 15:30:37,549 | INFO | HiveServer2-Handler-Pool: Thread-64 | </PERFLOG
method=parse start=1454571037536 end=1454571037549 duration=13
from=org.apache.hadoop.hive.ql.Driver> |
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogEnd(PerfLogger.java:148)
2016-02-04 15:30:37,549 | INFO | HiveServer2-Handler-Pool: Thread-64 | <PERFLOG
method=semanticAnalyze from=org.apache.hadoop.hive.ql.Driver> |
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogBegin(PerfLogger.java:121)

```

总结：对于编译阶段的错误，我们在客户端会看到 `ParseException` 或 `SemanticException`；此类错误最好定位，多数情况下 SQL 语法错误、或输入不支持的关键字等。

### 【运行阶段日志查看】

如下图，运行阶段日志简单分为三部分，任务提交、任务执行、任务完成标记。

```

org.apache.hadoop.hive.ql.Driver> |
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogBegin(PerfLogger.java:121)
2016-02-04 15:30:39,424 | INFO | HiveServer2-Background-Pool: Thread-9502 | Starting command: select
count(*) from src | org.apache.hadoop.hive.ql.Driver.execute(Driver.java:1401)
2016-02-04 15:30:39,424 | INFO | HiveServer2-Background-Pool: Thread-9502 | <PERFLOG
method=PreHook.org.apache.hadoop.hive.ql.security.auth Hook from=org.apache.hadoop.hiv
Hook from=org.apache.hadoop.hiv |
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogBegin(PerfLogger.java:121)
2016-02-04 15:30:39,424 | INFO | HiveServer2-Background-Pool: Thread-9502 | </PERFLOG
method=PreHook.org.apache.hadoop.hive.ql.security.authorization.plugin.DisallowTransform
Hook start=1454571039424 end=1454571039424 duration=0
from=org.apache.hadoop.hive.ql.Driver> |
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogEnd(PerfLogger.java:148)
.....此处省略 N 行

```

```

2016-02-04 15:30:39,425 | INFO | HiveServer2-Background-Pool: Thread-9502 | <PERFLOG
method=runTasks from=org.apache.hadoop.hive.ql.Driver> |
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogBegin(PerfLogger.java:121)
2016-02-04 15:30:39,425 | INFO | HiveServer2-Background-Pool: Thread-9502 | <PERFLOG
method=task.MAPRED.Stage-1 from=org.apache.hadoop.hive.ql.Driver> |
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogBegin(PerfLogger.java:121)
2016-02-04 15:30:39,447 | INFO | HiveServer2-Background-Pool: Thread-9502 | Launching
Job 1 out of 1 |
org.apache.hadoop.hive.ql.session.SessionState$LogHelper.printInfo(SessionState.java:855)
Tarting
.....此处省略 N 行

```

```

2016-02-04 15:32:26,525 | INFO | HiveServer2-Background-Pool: Thread-9502 | Total
MapReduce CPU Time Spent: 8 seconds 860 msec |
org.apache.hadoop.hive.ql.session.SessionState$LogHelper.printInfo(SessionState.java:855)
2016-02-04 15:32:26,525 | INFO | HiveServer2-Background-Pool: Thread-9502 | OK |
org.apache.hadoop.hive.ql.session.SessionState$LogHelper.printInfo(SessionState.java:855)

```

```
2016-02-04 15:32:26,525 | INFO | HiveServer2-Background-Pool: Thread-9502 | <PERFLOG
method=releaseLocks from=org.apache.hadoop.hive ql.Driver> |
org.apache.hadoop.hive ql.log.PerfLogger.PerfLogBegin(PerfLogger.java:121)
2016-02-04 15:32:26,525 | INFO | HiveServer2-Background-Pool: Thread-9502 | </PERFLOG
method=releaseLocks start=1454571146525 end=1454571146525 duration=0
from=org.apache.hadoop.hive ql.Driver> |
org.apache.hadoop.hive ql.log.PerfLogger.PerfLogEnd(PerfLogger.java:148)
2016-02-04 15:32:26,526 | INFO | HiveServer2-Background-Pool: Thread-9502 | </PERFLOG
method=Driver.run start=1454571039423 end=1454571039423 |3
from=org.apache.hadoop.hive ql.Driver> |
org.apache.hadoop.hive ql.log.PerfLogger.PerfLogEnd(PerfLogger.java:148)
```

任务执行结束

【Hive\_LOG\_00004】 有哪些快速定位搜索关键字？

- 1. 首先是出错的语句关键字如执行“select count(\*) from src”语句返回失败，那么我们要在 HiveServer 日志首先要搜索的便是“select count(\*) from src”；
- 2. 由于我们所有的执行语句信息均体现在 HiveServer 日志中， 所以可能存在一个语句搜出很多结果的情况。这样我们再需要用时间过滤出其他的信息。
- 3. 再次，当我定位到出错时间段的语句后，对于同一条 SQL 语句，我们会搜索出两个线程号；这个是个我们就要根据线程号来查看相应日志。如下，Thread-64 和 Thread-9502 分别对应编译阶段的线程号和任务执行阶段的任务号。

```
2016-02-04 15:30:37,536 | INFO | HiveServer2-Handler-Pool: Thread-64 | Parsing
command: select count(*) from src |
org.apache.hadoop.hive ql.parse.ParseDriver.parse(ParseDriver.java:185)
2016-02-04 15:30:37,538 | INFO | HiveServiceHealthChecker | Will try to open client
transport with JDBC Uri:
jdbc:hive2://160.138.1.18:21066/default;principal=hive/hadoop.hadoop.com@HADOOP.C
OM;sasl.qop=auth-conf;auth=kerberos;inner.client.marker=1454571037511 |
org.apache.hive.jdbc.HiveConnection.openTransport(HiveConnection.java:253)
.....此处省略 N 行
2016-02-04 15:30:39,424 | INFO | HiveServer2-Background-Pool: Thread-9502 | Starting
command: select count(*) from src |
org.apache.hadoop.hive ql.Driver.execute(Driver.java:1401)
2016-02-04 15:30:39,424 | INFO | HiveServer2-Background-Pool: Thread-9502 |
<PERFLOG
method=PreHook.org.apache.hadoop.hive ql.security.authorization.plugin.DisallowTransfor
mHook from=org.apache.hadoop.hive ql.Driver> |
org.apache.hadoop.hive ql.log.PerfLogger.PerfLogBegin(PerfLogger.java:121)
2016-02-04 15:30:39,424 | INFO | HiveServer2-Background-Pool: Thread-9502 |
</PERFLOG
method=PreHook.org.apache.hadoop.hive ql.security.authorization.plugin.DisallowTransfor
```

```
mHook      start=1454571039424      end=1454571039424      duration=0
from=org.apache.hadoop.hive.ql.Driver>
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogEnd(PerfLogger.java:148)
```

4. 然后我们只需要用简单的命令既可将我们需要的日志信息找出来，如， `grep "Thread-64" hive.log > bak.log`

其他常用搜索关键字：

**Exception**：异常堆栈信息，或可以根据具体异常类型搜索，如 `MetaException`, `ConnectionLossException` 等。

**Error**: 打印 Error 日志的信息

**Parsing command**: SQL 语句解析开始标记

**Starting command**: SQL 语句执行开始标记。

**Launching Job**: MR 任务开始提交。

**Trying to connect to metastore**: HiveServer 开始尝试连接 MetaStore 的开始标记

**Full GC**: 查看 GC 日志中是否有 Full GC。

## 【实战案例】 如何通过日志分析，快速定位问题？

### 【问题现象】

用户的在执行某个作业时，客户端报以下错误：

```
Error: Error while processing statement: FAILED: Execution Error, return code 2 from
org.apache.hadoop.hive.ql.exec.mr.MapRedTask (state=08S01,code=2)
```

### 【定位步骤】

步骤 1: 问题定界（确定到底问题是在客户端、HiveServer、MetaStore 还是 MR?）

通过日志我可以看到报错是在 `MapRedTask` 里，那么我们可以肯定问题是在 MR 里，所以肯定是需要去看 MR 日志。

#### 提示：

一般情况下 针对跑 MR 任务的返回码 有以下的意义：

`code=1` 提交 job 失败或者 job 运行时抛出异常时会返回的异常码， 这些错误通常在 HiveServer 侧有具体的异常堆栈信息

`code=2` 提交 job 和运行 job 时没有抛出异常，但是 job 运行时 map 阶段或者 reduce 阶段抛出 `RuntimeException` 时会抛出的异常，一般是各 operator 处理具体每一行数据时出现的 Hive 异常

这个问题定位就需要根据该 job 的 job id 去 job history 去查看该 job 的历史日志来定位问题

`code=3` 动态分区，job close 阶段 第一种情况是涉及动态分区的 job 在 close 阶段失败会返回的异常码，

该阶段主要工作是将数据移动至目标表或分区中时；第二种情况是涉及到 Join 的 job 在 close 阶段失败时会返回的异常码，该阶段主要工作是移动数据到最终路径下。

`code=5` 加载 Job 时创建临时文件夹失败

步骤 2: 通过 Hive 日志, 查看对应任务在 MR 的 Job Id.

为了找到 MR 对应的 Job Id, 我们需要首先去看 Hive 日志。

在 HiveServer 日志里, 以我们执行的语句为关键字进行搜索:

如下所示, 根据我们的搜索关键字可以搜索出来两条记录, 分别对应两个线程号:

1) 以 Parsing command 开头。排除掉, 此线程是在编译阶段。

2) 以 Starting command 开头。找到对应的线程号既 Thread-919.

```
2016-03-15 15:26:51,916 | INFO | HiveServer2-Handler-Pool: Thread-114 | Parsing
command: select count(*) from d1 a join d1 b join d1 c |
org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:185)
2016-03-15 15:26:51,917 | INFO | HiveServer2-Handler-Pool: Thread-114 | Parse
Completed | org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:206)
此处省略 N 行...
2016-03-15 15:26:53,375 | INFO | HiveServer2-Background-Pool: Thread-919 |
Concurrency mode is disabled, not creating a lock manager |
org.apache.hadoop.hive.ql.Driver.checkConcurrency(Driver.java:159)2016-03-15
15:26:53,375 | INFO | HiveServer2-Background-Pool: Thread-919 | <PERFLOG
method=Driver.execute from=org.apache.hadoop.hive.ql.Driver> |
org.apache.hadoop.hive.ql.log.PerfLogger.PerfLogBegin(PerfLogger.java:121)2016-03-
15 15:26:53,375 | INFO | HiveServer2-Background-Pool: Thread-919 | Starting
command: select count(*) from d1 a join d1 b join d1 c |
org.apache.hadoop.hive.ql.Driver.execute(Driver.java:1401)
```

继续往下看 Thread-919 的日志, 可以看到以 Starting job 开头的关键字, 从而找到对应的 jobId, 既 job\_1458004189946\_0005。

```
2016-03-15 15:27:00,683 | INFO | HiveServer2-Background-Pool: Thread-919 | Starting
Job = job_1458004189946_0005, Tracking URL = https://160-138-1-
182:26001/proxy/application_1458004189946_0005/ |
org.apache.hadoop.hive.ql.session.SessionState$LogHelper.printInfo(SessionState.java:
855)
```

步骤 3: 根据 jobId 去 Yarn 上查看对应的日志。

进入 OM 页面→Services→Yarn→ResourceManagers,如图

Services
Yarn Status

Status
Instances
Configuration
Resource Distribution

Start Service
Stop Service
Download Client
More Actions

### Yarn Summary

Health Status	Good
Configuration Status	Synchronized
Version	2.7.1
Running App	0
Pending App	0
ResourceManager WebUI	<a href="#">ResourceManager(Standby)</a> <a href="#">ResourceManager(Active)</a>

进入 Yarn 的页面后，找到我们刚才找到的 jobId，如下：

About
Nodes
Node Labels
Applications
NEW
NEW SAVING
SUBMITTED
ACCEPTED
RUNNING
FINISHED
FAILED
KILLED
Scheduler
Tools

Apps Submitted	Apps Pending	Apps Running	Apps Complet
5	0	0	5

### Scheduler Metrics

Scheduler Type

Capacity Scheduler

Show 20 entries

ID	User	Na
<a href="#">application 1458004189946 0005</a>	admin	select coun a join dl b join...c(St
<a href="#">application 1458004189946 0004</a>	admin	select coun a join dl b join...c(St
<a href="#">application 1458004189946 0003</a>	admin	select coun a join dl b join...c(St

点击链接进去→History→logs，打开日志信息，如下：

2016-03-15
15:27:26,844
INFO
[main]

org.apache.hadoop.conf.Configuration.deprecation: map.input.start is deprecated.  
Instead, use mapreduce.map.input.start

```

2016-03-15      15:27:26,844      INFO      [main]
org.apache.hadoop.conf.Configuration.deprecation: map.input.length is deprecated.
Instead, use mapreduce.map.input.length
2016-03-15  15:27:26,844  INFO  [main]  org.apache.hadoop.mapred.MapTask:
numReduceTasks: 1
2016-03-15  15:27:27,325  FATAL [main] org.apache.hadoop.mapred.YarnChild: Error
running child :java.lang.OutOfMemoryError: Java heap space
    at org.apache.hadoop.mapred.MapTask$MapOutputBuffer.init(MapTask.java:986)
    at org.apache.hadoop.mapred.MapTask.createSortingCollector(MapTask.java:402)
    at org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:442)
    at org.apache.hadoop.mapred.MapTask.run(MapTask.java:343)

```

至此，我们已经找到问题原因是 Map 阶段的 OOM。

### 【解决方案】

调大 Map 使用内存。(详细见【Hive-F100003】)

## 2、常见问题

### 【参数及配置类】

[Hive-10001]执行 set 命令的时候报 Cannot modify xxx at runtime.

### 【问题现象】

执行 set 命令时报以下错误：

```

0: jdbc:hive2://160.138.1.18:21066/> set mapred.job.queue.name=QueueA;
Error: Error while processing statement: Cannot modify mapred.job.queue.name at list of
params that are allowed to be modified at runtime (state=42000,code=1)

```

### 【解决方案】

方案 1：

1. 登录 OM 页面—>Hive→Configuration→All→HiveServer→Security
2. 将要添加的参数添加到配置项 hive.security.authorization.sqlstd.confwhitelist 中，
3. 点击保存并重启 HiveServer 后即可。如下图所示：

Parameter	Value
hive.security.authorization.sqlstd.confw...	hive.optimize.index.filter,hive.optimize.index.filter.compact.minsize,hive.index.co
* hive.security.temporary.function.need.a...	<input checked="" type="radio"/> true <input type="radio"/> false
* hive.security.transform.disallow	<input checked="" type="radio"/> true <input type="radio"/> false
hive.security.whitelist.switch	<input type="checkbox"/> ON <input checked="" type="checkbox"/> OFF

方案 2：



1. 登录 OM 页面—>Hive→Configuration→All→HiveServer→Security
2. 找到选项 hive.security.whitelist.switch，选择 OFF,点击保存并重启即可。

## [Hive-10002]怎样在 Hive 提交任务的时候指定队列？

### 【解决方案】

1. 如下，在执行语句前通过下述参数设置：

```
set mapred.job.queue.name=QueueA;  
select count(*) from rc;
```

1. 提交任务后，可在 Yarn 页面看到，如下任务已经提交到队列 QueueA 了。

User:	admin
Name:	select count(*) from rc(Stage-1)
Application Type:	MAPREDUCE
Application Tags:	
YarnAppState:	FINISHED
Queue:	QueueA
FinalStatus Reported by AM:	SUCCEEDED
Started:	Thu Mar 03 09:01:58 +0800 2016
Elapsed:	1mins, 0sec
Tracking URL:	History
Log Aggregation Status	Status
Diagnostics:	

**注意：队列的名称区分大小写，如写成 queueA,Queuea 均无效。**

## [Hive-10003]客户端怎么设置 Map/Reduce 内存？

### 【解决方案】

1. Hive 在执行 SQL 语句前，可以通过 set 命令 来设置 M/R 相关客户端参数
2. 以下为与 M/R 内存相关的参数：

```
set mapreduce.map.memory.mb=4096; // 每个Map Task需要的内存量  
set mapreduce.map.java.opts=-Xmx3276M; // 每个Map Task 的Java最大使用内存  
set mapreduce.reduce.memory.mb=4096; // 每个Map Task需要的内存量  
set mapreduce.reduce.java.opts=-Xmx3276M; // 每个Reduce Task 的Java最大使用内存  
set mapred.child.java.opts=-Xms1024M -Xmx3584M; //此参数为全局参数，既对Map和
```



**注意：**参数设置只对当前 session 有效。

## [Hive-10004]如何在导入表时指定输出的文件压缩格式

### 【解决方案】

1. 当前 Hive 支持以下几种压缩格式：

```
org.apache.hadoop.io.compress.BZip2Codec
org.apache.hadoop.io.compress.Lz4Codec
org.apache.hadoop.io.compress.DeflateCodec
org.apache.hadoop.io.compress.SnappyCodec
org.apache.hadoop.io.compress.GzipCodec
```

2. 如需要全局设置，既对所有表都进行压缩；可以在 OM 页面上进行全局配置，如下：

The screenshot shows the Hive OM configuration interface. The 'hive.exec.compress.output' setting is set to 'true'. Below it, the 'mapreduce.output.fileoutputformat.compress.codec' setting is configured with a selection of codecs. The 'SnappyCodec' option is selected.

**注意：**hive.exec.compress.output=true; 这个一定要选择 true，否则下面选项不会生效。

3. 如需在 session 级设置，只需要在执行命令前做如下设置即可：

```
set hive.exec.compress.output=true;
set mapred.output.compression.codec=org.apache.hadoop.io.compress.SnappyCodec;
```

## [Hive-10005]desc 描述表过长时，无法显示完整

### 【解决方案】

1. 启动时，设置参数 `maxWidth=20000` 即可，如下：

```
[root@192-168-1-18 logs]# beeline --maxWidth=2000
scan complete in 3ms
Connecting to
省略.....
Beeline version 1.1.0 by Apache Hive
```

2. 扩展：可通过 `beeline -help` 看到很多关于客户端显示的设置。如下：

```
-u <database url>          the JDBC URL to connect to
-n <username>              the username to connect as
-p <password>              the password to connect as
-d <driver class>          the driver class to use
-i <init file>              script file for initialization
-e <query>                 query that should be executed
-f <exec file>             script file that should be executed
--hiveconf property=value  Use value for given property
--color=[true/false]       control whether color is used for display
--showHeader=[true/false]  show column names in query results
--headerInterval=ROWS;     the interval between which headers are displayed
--fastConnect=[true/false] skip building table/column list for tab-completion
--autoCommit=[true/false] enable/disable automatic transaction commit
--verbose=[true/false]     show verbose error messages and debug info
--showWarnings=[true/false] display connection warnings
--showNestedErrs=[true/false] display nested errors
--numberFormat=[pattern]   format numbers using DecimalFormat pattern
--force=[true/false]       continue running script even after errors
--maxWidth=MAXWIDTH        the maximum width of the terminal
--maxColumnWidth=MAXCOLWIDTH the maximum width to use when displaying
columns
--silent=[true/false]      be more silent
--autosave=[true/false]    automatically save preferences
--outputformat=[table/vertical/csv2/tsv2/dsv/csv/tsv] format mode for result display
Note that csv, and tsv are deprecated - use
csv2, tsv2 instead
--truncateTable=[true/false] truncate table column when it exceeds length
--delimiterForDSV=DELIMITER specify the delimiter for delimiter-separated values
output format (default: |)
--isolation=LEVEL          set the transaction isolation level
--nullemptystring=[true/false] set to true to get historic behavior of printing null as
empty string
```

---

--socketTimeout=n The default value is 300.	socket connection timeout interval, in second.
--	--

## [Hive-10006]增加分区列后再 insert 数据显示为 NULL

### 【问题现象】

执行如下命令：

```
create table test_table(
col1 string,
col2 string
)
PARTITIONED BY(p1 string)
STORED AS orc tblproperties('orc.compress'='SNAPPY');
alter table test_table add partition(p1='a');
insert into test_table partition(p1='a') select col1,col2 from temp_table;
alter table test_table add columns(col3 string);
insert into test_table partition(p1='a') select col1,col2,col3 from temp_table;
这个时候 select * from test_table where p1='a' 看见的列 col3 全为 NULL
alter table test_table add partition(p1='b');
insert into test_table partition(p1='b') select col1,col2,col3 from temp_table;
select * from test_table where p1='b' 能看见 col3 有值
```

### 【解决方案】

add column 的时候加入 cascade 关键字即可，如下：

```
alter table test_table add columns(col3 string) cascade;
```

### 【权限管理类】

## [Hive-20001]创建新用户，执行查询时报无权限。

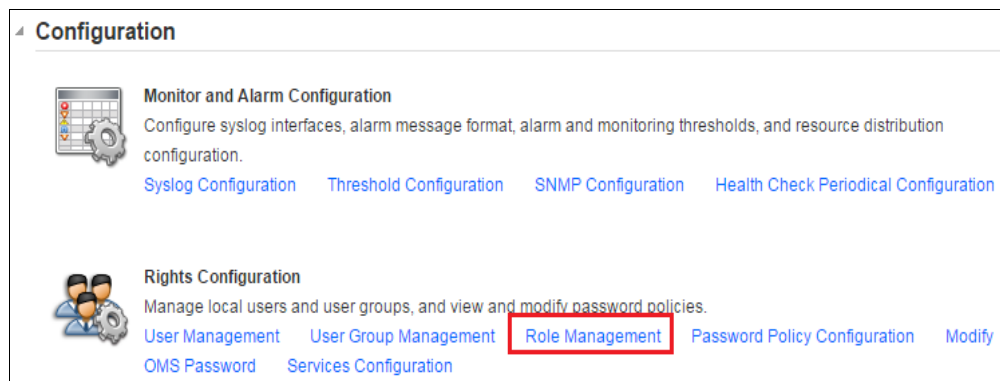
### 【问题现象】

创建了新用户，但是执行查询的时候报错无权限。

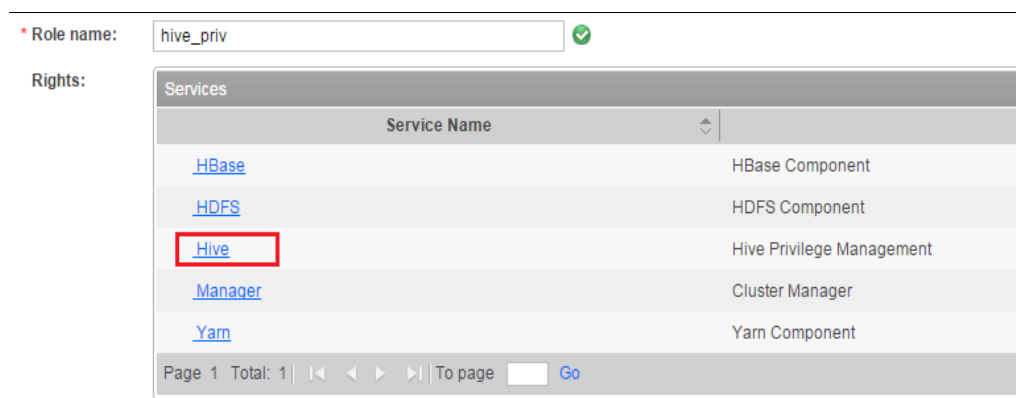
```
Error: Error while compiling statement: FAILED: HiveAccessControlException Permission
denied: Principal [name=hive, type=USER] does not have following privileges for operation
QUERY  [[SELECT] on Object [type=TABLE_OR_VIEW, name=default.t1]]
(state=42000,code=40000)
```

## 【解决方案】

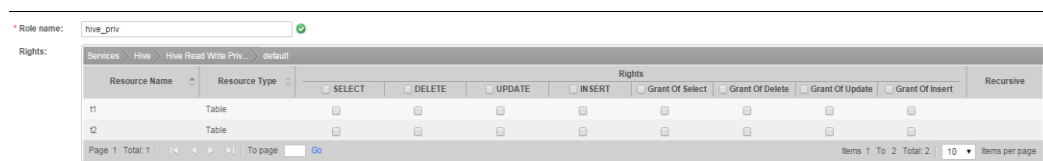
1. 登录 OM 平台
2. 进入 System 标签页
3. 点击 Configuration 下的 Role Management 进入到增加角色界面



4. 点击 Add Role 进入到角色编辑界面
5. 输入角色名称
6. 点开 Hive，这时候会列出 Hive 的两种权限，Hive 管理员权限和 Hive 表的读写权限



7. 选择 Hive 读写权限，这时候会列出 Hive 中的所有数据库，选择需要授权的数据库并点开，之后会列出该数据库下的所有表，以及当前已经拥有的权限。



勾选需要的权限然后点 OK 完成角色创建。

创建用户并与角色绑定

1. 登录 OM 平台
2. 进入 System 标签页
3. 点击 Configuration 下的 User Management 进入到用户界面

System > User Management >

+ Add User

Username	
admin	super admin user of manager system
user1	
user2	
user3	

Page 1 Total: 1 | < < > > | To page  Go

4. 点击 Add User 进入到增加用户界面，填写用户名，选择用户类型，选择需要加入的用户组（需要使用 Hive 的话必须加入 Hive 组）

System > User Management > Add User >

\* Username:  ✓

\* Usertype:  ✓

\* Password:  ✓

\* Confirm password:  ✓

\* User Group: [Select user groups and join them](#)

hive X

hadoop X

\* Primary Group:

5. 选择需要赋予的权限，绑定刚才创建的角色即可

Assign rights by [Select and bind roles](#)

role:

hive\_priv X

6. 完成操作

---

[Hive-20002]执行 SQL 提交任务到指定队列报错。

【问题现象】

执行 SQL 提交任务到 Yarn 报如下错误：

```
Error: Error while processing statement: FAILED: Execution Error, return code 1 from
org.apache.hadoop.hive.ql.exec.mr.MapRedTask (state=08S01,code=1)
```

【解决方案】

用户无 YARN 队列提交权限，需要赋予 YARN 相应队列的提交权限  
权限配置步骤请参考【Hive-F300001】

[Hive-20003]执行 load data inpath 命令报错

【问题现象】

执行 load data inpath 报如下错误：

错误 1：

```
HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does
not have following privileges on Object [type=DFS_URI,
name=hdfs://hacluster/tmp/input/mapdata] for operation LOAD : [OBJECT OWNERSHIP]
```

错误 2：

```
HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does
not have following privileges on Object [type=DFS_URI,
name=hdfs://hacluster/tmp/input/mapdata] for operation LOAD : [INSERT, DELETE]
```

错误 3：

```
SemanticException [Error 10028]: Line 1:17 Path is not legal "file:///tmp/input/mapdata":
Move from: file:/tmp/input/mapdata to: hdfs://hacluster/user/hive/warehouse/tmp1 is
not valid. Please check that values for params "default.fs.name" and
"hive.metastore.warehouse.dir" do not conflict.
```

【解决方案】

Hive 对 load data inpath 命令有如下权限要求，请对照下述要求是否满足：

1. 文件的 owner 需要为执行命令的用户
2. 当前用户需要对该文件有读、写权限
3. 当前用户需要对该文件的目录有执行权限
4. 由于 load 操作会将该文件移动到表对应的目录中，所以要求当前用户需要对该文件的目录有写权限

- 
5. 要求文件的格式与表指定的存储格式相同。如创建表时指定 `stored as rcfile`，但是文件格式为 `txt`，则不符合要求。
  6. 文件必须是 **HDFS** 上的文件，不可以用 `file://` 的形式指定本地文件系统上的文件
  7. 文件名不能以下横线（`_`）或点（`.`）开头，以这些开头的文件会被忽略
- 如下所示，如果用用户 `test_hive` load 数据，正确的权限如下：

```
[root@192-168-1-18 duan]# hdfs dfs -ls /tmp/input2
16/03/21 14:45:07 INFO hdfs.PeerCache: SocketCache disabled.
Found 1 items
-rw-r--r--   3 hive_test hive          6 2016-03-21 14:44 /tmp/input2/input.txt
```

## [Hive-20004]执行 `load data local inpath` 命令报错

### 【问题现象】

执行 `load data local inpath` 报如下错误：

#### 错误 1：

```
HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does
not have following privileges on Object [type=LOCAL_URI, name=file:/tmp/input/mapdata]
for operation LOAD : [SELECT, INSERT, DELETE]
```

#### 错误 2：

```
HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does
not have following privileges on Object [type=LOCAL_URI, name=file:/tmp/input/mapdata]
for operation LOAD : [OBJECT OWNERSHIP]
```

#### 错误 3：

```
SemanticException Line 1:23 Invalid path "/tmp/input/mapdata": No files matching path
file:/tmp/input/mapdata
```

### 【解决方案】

Hive 对 `load data inpath` 命令有如下权限要求，请对照下述要求是否满足：

1. 由于所有的命令都是发送的主 **HiveServer** 上去执行的，所以要求此文件在主 **HiveServer** 节点上
2. **HiveServer** 进程是以操作系统上的 `omm` 用户启动的，所以要求 `omm` 用户对此文件有读权限，对此文件的目录有读、执行权限
3. 文件的 `owner` 需要为执行命令的用户
4. 当前用户需要对该文件有读、写权限
5. 要求文件的格式与表指定的存储格式相同。如创建表时指定 `stored as rcfile`，但是文件格式为 `txt`，则不符合要求。
6. 文件名不能以下横线（`_`）或点（`.`）开头，以这些开头的文件会被忽略

---

## [Hive-20005]执行 insert overwrite directory 命令报错

### 【问题现象】

执行 insert overwrite directory 报如下错误:

#### 错误 1 :

```
HiveAuthzPluginException Error getting permissions for /tmp/out/abc: Permission denied:
user=user1, access=EXECUTE, inode="/tmp/out":admin:hive:drwxr-----
```

#### 错误 2 :

```
HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does
not have following privileges on Object [type=DFS_URI, name=/tmp/out/abc] for operation
QUERY : [INSERT]r1, access=EXECUTE, inode="/tmp/out":admin:hive:drwxr-----
```

#### 错误 3 :

```
HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does
not have following privileges on Object [type=DFS_URI, name=/tmp/out/abc] for operation
QUERY : [INSERT]
```

### 【解决方案】

Hive 对 insert overwrite directory 命令有如下权限要求, 请对照下述要求是否满足:

**如果目录不存在:**

1. 要求对此目录的父目录有读、写、执行权限

**如果目录存在:**

1. 要求对此目录有读、写、执行权限 (如果此目录下有子目录或子文件, 要求对子目录有读、写、执行权限, 对子文件有写权限)
2. 由于会先将此目录删掉, 所以要求对此目录的父目录有读、写、执行权限

## [Hive-20006]执行 insert overwrite local directory 命令报错

### 【问题现象】

执行 load data local inpath 报如下错误:

```
HiveAccessControlException Permission denied. Principal [name=user1, type=USER] does not
have following privileges on Object [type=LOCAL_URI, name=file:/tmp/out/1] for operation
QUERY : [INSERT]
```

### 【解决方案】

Hive 对 insert overwrite local directory 命令有如下权限要求, 请对照下述要求是否满足:



1. `dir` 必须加上 `file://` 前缀，否则会有很多权限不正确的错误
2. 由于所有的命令都是发送的主 `HiveServer` 上去执行的，所以要求此目录在主 `HiveServer` 节点上

#### 如果目录不存在

1. `HiveServer` 进程是以操作系统上的 `omm` 用户启动的，所以要求 `omm` 用户对此目录的父目录有读、写、执行权限
2. 要求当前操作用户对指定目录的父目录有写权限
3. 如果父目录中还有其他目录或文件，则要求 `omm` 用户对这些目录有读、写、执行权限，对这些文件有写权限，同时当前操作的用户对这些目录和文件有写权限

#### 如果目录存在

1. `HiveServer` 进程是以操作系统上的 `omm` 用户启动的，所以要求 `omm` 用户对此目录有读、写、执行权限（如果此目录下有子目录或子文件，要求对子目录有读、写、执行权限，对子文件有写权限）
2. 要求当前操作用户对指定目录有写权限（如果此目录下有子目录或子文件，要求对这些目录/文件有写权限）
3. 由于会先将此目录删掉，所以要求对此目录的父目录有读、写、执行权限

## [Hive-20007]执行 `create external table` 报错

### 【问题现象】

执行命令：`create external table xx(xx int)stored as textfile location '/tmp/aaa/aaa'`，报以下错误：

```
Permission denied. Principal [name=fantasy, type=USER] does not have following privileges on Object [type=DFS_URI, name=/tmp/aaa/aaa] for operation CREATETABLE : [SELECT, INSERT, DELETE, OBJECT OWNERSHIP] (state=42000,code=40000)
```

### 【解决方案】

创建外部表时，会判断当前用户对指定的目录以及该目录下其它目录和文件是否有读写权限，如果该目录不存在，会去判断其父目录，依次类推。如果一直不满足就会报权限不足。而不是报指定的目录不存在。

请确认当前用户对路径`"/tmp/aaa/aaa"`有读写权限，如果该路径不存在，确认对其父路径有读写权限。

## [Hive-20008]在 `beeline` 客户端执行 `dfs -put` 命令报错

### 【问题现象】

执行命令：`dfs put /opt/kv1.txt /tmp/kv1.txt`，报以下错误：

```
Permission denied. Principal [name=admin, type=USER] does not have following privileges on Object [type=COMMAND_PARAMS,name=[put, /opt/kv1.txt, /tmp/kv1.txt]]
```

```
for operation DFS : [ADMIN PRIVILEGE] (state=,code=1)
```

### 【解决方案】

当前登录的用户不具备操作此命令的权限，如果登录的当前用户具有 `admin` 角色，请用 `set role admin` 来切换到 `admin` 角色操作。如果不具备 `admin` 角色，在 OM 管理界面 System 下 Role Management 中给对应的角色赋予对应的权限。

## [Hive-20009]执行 set role admin 报无权限

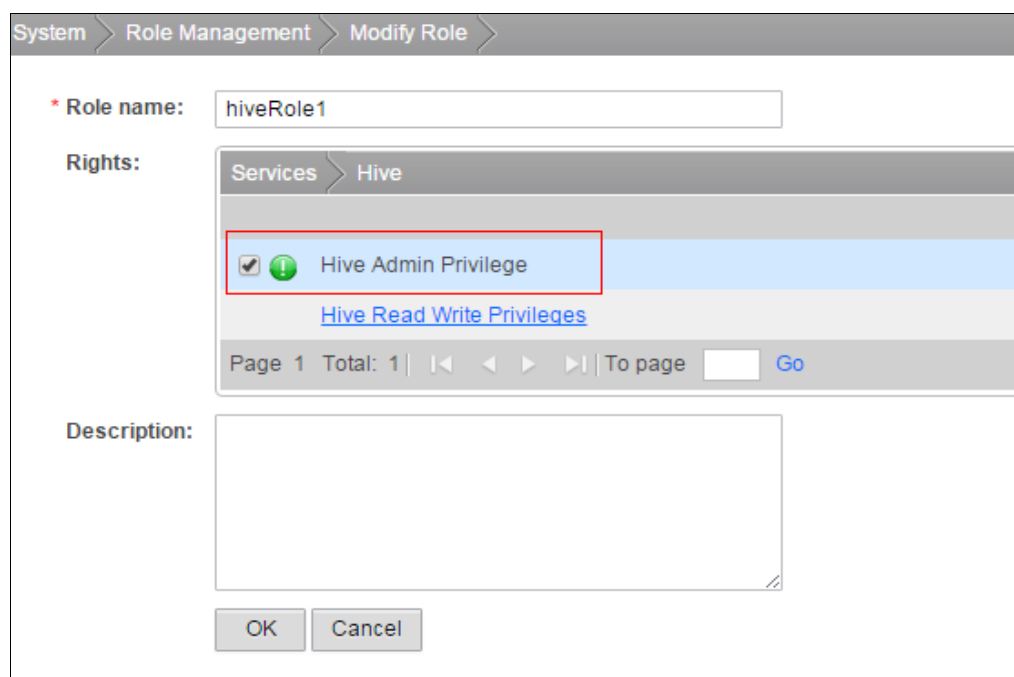
### 【问题现象】

执行命令 `set role admin` 报下述错误：

```
0: jdbc:hive2://10.11.42.26:21066/> set role admin;  
Error: Error while processing statement: FAILED: Execution Error, return code 1 from org.a  
pache.hadoop.hive.ql.exec.DDLTask. dmp_B doesn't belong to role admin (state=08S01,co  
de=1)
```

### 【解决方案】

登录 OM 页面编辑指定用户所绑定的 role,选择 Hive Admin Privilege.



## [Hive-20010]通过 beeline 创建 UDF 时候报错

### 【问题现象】

执行命令：`create function fn_test3 as 'test.MyUDF' using jar 'hdfs:///tmp/udf2/MyUDF.jar';`

报以下错误:

```
Error: Error while compiling statement: FAILED: HiveAccessControlException Permission
denied: Principal [name=admin, type=USER] does not have following privileges for operation
CREATEFUNCTION [[ADMIN PRIVILEGE] on Object [type=DATABASE, name=default], [ADMIN
PRIVILEGE] on Object [type=FUNCTION, name=default.fn_test3]] (state=42000,code=40000)
```

### 【解决方案】

Hive 中创建永久函数需要特殊的 role admin。所以遇到上述问题,在执行语句前执行:  
`set role admin` 即可解决。

## 【安装启动类】

### [Hive-30001]Hive 启动失败报 Cannot find hadoop installation

#### 【问题现象】

客户先按安装 C50 版本,之后卸载重装 C30 版本,然后启动 Hive 起不来。提示下述错误:

```
Cannot find hadoop installation: \${HADOOP_HOME} or \${HADOOP_PREFIX} must be set or
hadoop must be in the path"
```

#### 【解决方案】

Hive 启动时,默认 HADOOP\_HOME 环境变量指向 /opt/huawei/Bigdata/hive-0.13.1/hadoop,但是发现客户客户手动在/etc/profile 下配置了 HADOOP\_HOME 环境变量,最终导致 Hive 启动失败。

手动删除/etc/profile 中 HADOOP\_HOME 配置即可。

### [Hive-30002]Hive 启动失败问题的原因有哪些?

Hive 启动失败最常见的原因是 metastore 实例无法连接上 DBservice。可以查看 metastore.log 中具体的错误信息。目前总结连不上 DBservice 原因主要有:

#### 【可能原因 1】

DBservice 没有初始化好 Hive 的元数据库 hivemeta,可以通过以下方式验证。

#### 【解决方案】

1. 执行以下命令:

```
source /opt/huawei/Bigdata/dbservice/.dbservice_profile
gsql -h 160.172.0.44 (DB 的浮动 ip) -p 20051 -d hivemeta -U hive -W HiveUser@
```

- 
2. 如果不能正确进入交互界面，说明数据库初始化失败。  
如果报如下错误：`org.postgresql.util.PSQLException: FATAL: no pg_hba.conf entry for host "160.164.0.146", database "HIVEMETA"`
  3. 说明 DBservice 的配置文件可能丢失了 hivemeta 的配置，在 DBservice 所在的节点上
  4. `/srv/BigData/dbdata_service/data/pg_hba.conf`，在文件最后面追加 `host hivemeta hive 0.0.0.0/0 sha256`
  5. 执行 `source /opt/huawei/Bigdata/dbservice/.dbservice_profile`
  6. 执行 `gs_ctl -D $GAUSSDATA reload` # 使修改后的配置生效。

### 【可能原因 2】

DBservice 的浮动 IP 配置有误，导致 metastore 节点 IP 无法正确连接浮动 IP，或者是在与该 ip 建立互信的时候失败导致 metastore 启动失败。

### 【解决方案】

DBservice 的浮动 IP 配置需要同网段内没有被使用过的 ip，也就是在配置前 ping 不通的 ip，请修改 DBService 浮动 IP 配置。

## 【监控告警类】

### [Hive-40001]Hive 状态为 Bad 问题总结

#### 【可能原因】

DBservice 服务不可用会导致 Hive 服务状态为 Bad。  
HDFS 服务不可用会导致 Hive 服务状态为 Bad。  
Zookeeper 服务不可用会导致 Hive 服务状态为 Bad。  
LDAP/KrbServer 服务不可用会导致 Hive 服务状态为 Bad。  
metastore 实例不可用会导致 Hive 服务状态为 Bad。

### [Hive-40002]Hive 服务状态为 Partially Healthy 总结

#### 【可能原因】

HBase 服务不可用会导致 Hive 服务的健康状态变为 Partially Healthy。

### [Hive-40003]Hive 服务状态为 Unknown 总结

#### 【可能原因】

Hive 服务停止。  
hiveserver 实例出现双主。

## [Hive-40004]Hiveserver 或者 Metastore 实例的健康状态为 unknown

### 【问题现象】

hiveserver 或者 metastore 实例的健康状态为 unknown

### 【可能原因】

hiveserver 或者 metastore 实例被停止。

## [Hive-40005]Hiveserver 或者 Metastore 实例的健康状态为 Concerning

### 【问题现象】

Hiveserver 或者 Metastore 实例的健康状态为 Concerning

### 【可能原因】

hiveserver 或者 metastore 实例在启动的时候发生异常，无法正常启动。如，当修改 MetaStore GC 参数时，如下将符号 ‘-’ 误删掉，因此遇到此类异常应该检查最近的变更项，以确认是否配置有误：

```
METASTORE_GC_OPTS=Xms1G -Xmx2G -DignoreReplayReqDetect  
-XX\:CMSFullGCsBeforeCompaction\=1 -XX\:+UseConcMarkSweepGC  
-XX\:+CMSParallelRemarkEnabled -XX\:+UseCMSCompactAtFullCollection  
-XX\:+ExplicitGCInvokesConcurrent -server -XX\:MetaspaceSize\=128M  
-XX\:MaxMetaspaceSize\=256M
```

## [Hive-40006]Hive 服务健康状态和 Hive 实例健康状态的区别

Hive 服务的健康状态（也就是在 services 界面看到的健康状态）有 Good, Bad, Partially Healthy, Unknown 四种状态，四种状态除了取决于 Hive 本身服务的可用性（会用简单的 sql 来检测 Hive 服务的可用性），还取决于 Hive 服务所依赖的其他组件的服务状态。

Hive 实例分为 Hiveserver 和 Metastore 两种，健康状态有 Good, Concerning, Unknown 三种状态，这三种状态是通过 jmx 通信来判定，与实例通信正常时为 Good，通信异常时为 Concerning，无法通信时为 Unknown

## [Hive-40007]Hive 中的告警有哪些以及触发的场景

Alarm ID	Alarm Severity	Is Auto Clear	Alarm Name	Alarm Type
----------	----------------	---------------	------------	------------

16000	Minor	TRUE	Percentage of Sessions Connected to the HiveServer to Maximum Number Allowed Exceeds the Threshold	故障告警
16001	Minor	TRUE	Hive Warehouse Space Usage Exceeds the Threshold	故障告警
16002	Minor	TRUE	The Successful Hive SQL Operations Lower than The Threshold	故障告警
16003	Minor	FALSE	HiveServer Failover	事件
16004	Critical	TRUE	Hive Service Unavailable	故障告警

告警触发的场景如下

**16000:** 当连接 HiveServer 的 session 数占允许连接总数的比率超过设定的阈值的时候触发告警。如连接的 session 数为 9，总连接数为 12，设定的阈值为 70%， $9/12 > 70\%$  便触发告警。

**16001:** 当 hive 使用的 hdfs 容量占分配给 hive 的总 hdfs 总容量的比率超过设定的阈值时触发告警。如分配给 hive 的是 500G, hive 已经使用 400G, 设定的阈值时 75%,  $400/500 > 75\%$  便触发告警。

**16002:** 当执行 sql 的成功率低于设定的阈值时变触发告警。如你执行了 4 条失败了 2 条，设定的阈值为 60%，成功率  $2/4 < 60\%$  便触发告警。

**16003:** HiveServer 主备发生切换时触发事件。事件在 event 中查看。

**16004:** Hive 服务的健康状态变为 Bad 时触发告警。

说明：

告警的阈值和告警的级别以及触发告警的时间段可以在 OM 界面中 System 下 Threshold Configuration 中设定。

Hive 运行相关的指标可以在 Hive 监控界面查看。

