



## Hue 常见问题及解决方案

---

### 1 Hue 数据库中文编码。

问题现象：

把 hue 的数据库从默认的 sqlite3 切换到 mysql 之后，在 hive editor 中执行带有中文的 sql 会发生如下报错。初始情况下，使用 sqlite3 数据库，可以正常执行。

Your query has the following error(s):

(1366, "Incorrect string value: '\xE9\xBB\x84\xE4\xBA\xA6...' for column 'query' at row 1")

```
1 select * from users where user_full_name like '%黄亦辰%'
```

#### 问题原因：

Mysql 默认的编码格式为 latin1，hue 的数据库迁移工具并不会设置编码格式，迁移之后，在 Mysql 中，hue 的所有表以及表中的每个字段都是 Mysql 默认编码的。因此执行的时候会出现问题。

#### 解决方案：

1 修改 Mysql 的默认编码格式为 utf8（可选）

redhat 系统下，编辑/etc/my.cnf 文件，添加如下配置项。

```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
default-storage-engine=InnoDB
sql_mode=STRICT_ALL_TABLES
user=mysql
default-character-set = utf8
character-set-server = utf8
collation-server = utf8_general_ci
init_connect = 'SET NAMES utf8'
# Disabling symbolic-
links is recommended to prevent assorted security risks
symbolic-links=0

[mysql.server]
default-character-set = utf8

[mysqld_safe]
log-error=/var/log/mysql.log
pid-file=/var/run/mysqld/mysqld.pid
default-character-set = utf8

[client]
default-character-set = utf8
```

重启 mysql。重启完成之后，用 SHOW VARIABLES LIKE 'character\_set\_%';输出如下：

```
| character_set_client | latin1 |
| character_set_connection | latin1 |
| character_set_database | latin1 |
| character_set_results | latin1 |
| character_set_server | latin1 |
| character_set_system | utf8 |
| character_sets_dir | /usr/share/mysql/charsets/ |
```

如果上述任一项中不是 utf8，则需要设置为 utf8。命令为：

```
SET character_set_client = utf8;
SET character_set_connection = utf8;
SET character_set_database = utf8;
SET character_set_results = utf8;
SET character_set_server = utf8;
```

## 2 修改 hue 数据库中所有表格的编码格式为 utf8

由于 hue 中表格比较多，一个一个修改编码格式非常的麻烦，可以采用下面的语句首先生成修改表格编码格式的语句，然后再一起执行。

```
select CONCAT("ALTER TABLE ",TABLE_NAME," DEFAULT
CHARACTER SET utf8 COLLATE utf8_general_ci;") from
information_schema.TABLES where TABLE_SCHEMA="hue";
```

执行完该语句之后，会生成所有 hue 中表格，更改编码格式

的 sql 语句，可以拷贝到 vim 中。将'|'替换为"字符后，一起拷贝至 mysql 中运行。

3 修改所有表格中所有字段的编码格式。（这一步很容易被忽略，往往以为修改了表格的编码格式就可以了），修改的方式与第 2 步类似，同样是首先生成修改表格字段编码格式的语句，然后再执行。

```
select CONCAT("ALTER TABLE `",TABLE_NAME,"` CHANGE `",COLUMN_NAME,"`",COLUMN_NAME,"` ",COLUMN_TYPE, " CHARACTER SET utf8 COLLATE utf8_general_ci;") from information_schema.COLUMNS where CHARACTER_SET_NAME = 'latin1' and TABLE_SCHEMA = 'hue';
```

完成上述步骤后，就应该能够成功执行带有中文的 sql 了。

**问题 2：**kerberos 环境下，如何在 hue workflow 中执行 hive sql。

问题现象：

在启用了 Kerberos 的集群上。使用 hue 的 workflow 功能，创建 shell 脚本的任务。任务 test12：beeline -u

```
'jdbc:hive2://10.203.24.152:10000/default;principal=hive/hadoopcdhd  
evqa02.jqdev.shanghaigm.com@JQDEV.SHANGHAIGM.COM' -e "show  
tables"
```

在命令行中运行这两个脚本都能够运行成功，但是在 hue 中运行会出现

## 报错

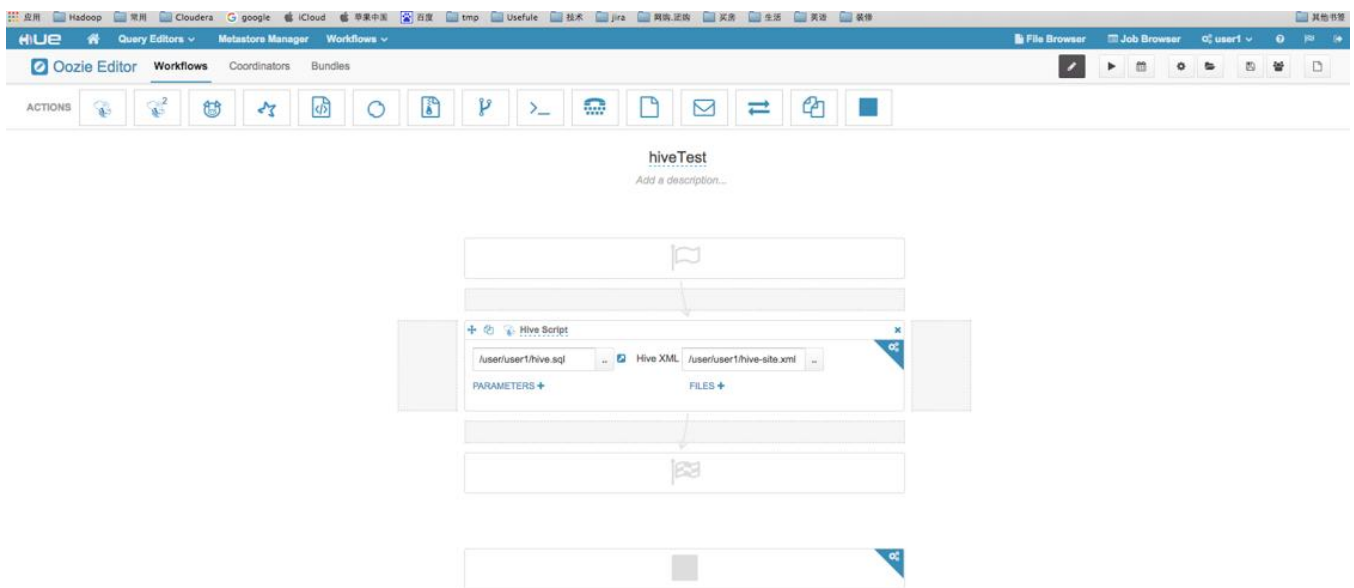
```
-bash-4.1$ beeline -u 'jdbc:hive2://10.203.24.152:10000/default;principal=hive/hadoopcdhdevqa02.jqdev.shanghaigm.com@JQDEV.SHANGHAIGM.COM' -e "show tables"
scan complete in 4ms
Connecting to jdbc:hive2://10.203.24.152:10000/default;principal=hive/hadoopcdhdevqa02.jqdev.shanghaigm.com@JQDEV.SHANGHAIGM.COM
Connected to: Apache Hive (version 1.1.0-cdh5.4.2)
Driver: Hive JDBC (version 1.1.0-cdh5.4.2)
Transaction isolation: TRANSACTION_REPEATABLE_READ
+-----+
| tab_name |
+-----+
| hyperanalytics_log |
| mobileportal_pro |
+-----+
2 rows selected (0.597 seconds)
Beeline version 1.1.0-cdh5.4.2 by Apache Hive
Closing: 0: jdbc:hive2://10.203.24.152:10000/default;principal=hive/hadoopcdhdevqa02.jqdev.shanghaigm.com@JQDEV.SHANGHAIGM.COM
-bash-4.1$
```

### 问题原因：

这个问题是因为在 hue 中运行的时候，没有进行 kinit，因此没有可以使用的 credential，没有提供合法的身份，因此会出现报错。实际上，在命令行中，能够运行成功是因为命令行中有现存的 credential（之前做过 kinit），因此能够执行成功，如果没有 credential 的话，是会报同样的错的。

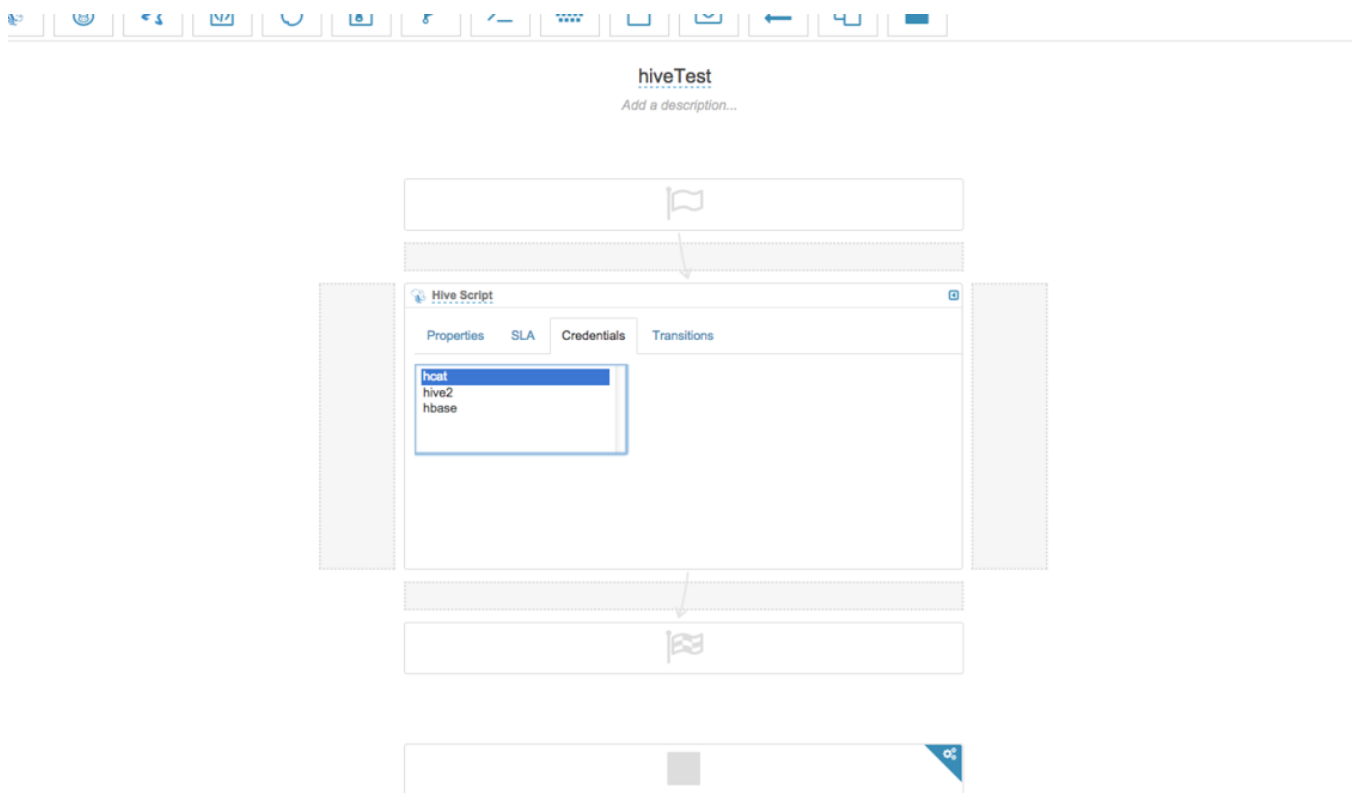
### 解决办法：

在 hue 中解决这个问题，可以通过使用 hivescript，来解决。



hivescript 需要两个文件输入，其中一个是 hive.sql，包含需要运行的 hive 语句，另一个是 hive XML 是 hive 的配置文件，可以从集群的 /etc/hive/conf/hive-site.xml。推荐将这两个文件上传到 hdfs，然后在 hue 中进行配置。

还需要配置一个 credential，如下图所示，选择 hcat。



配置完成，就可以执行了。

### 问题 3：Java 程序中如何访问启用 Kerberos 的 hdfs。

问题现象：

在 java 程序中使用 jaas 方式配置，访问启用 kerberos 的 hdfs，出现如下错误：



```

SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-5.4.2-1.cdh5.4.2.p0.2/jars/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/hdfs/portal/work/lib/slf4j-log4j12-1.5.11.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
Search Subject for Kerberos V5 INIT cred <<DEF>>, sun.security.jgss.krb5.Krb5InitCredential)
Debug is true storeKey false useTicketCache false useKeyTab true doNotPrompt true ticketCache is null isInitiator true keyTab is /tmp/load_test.keytab refreshKrb5Co
nfig is false principal is load_test@LAB.COM tryFirstPass is false useFirstPass is false storePass is false clearPass is false
principal is load_test@LAB.COM
will use keytab
commit succeeded

Exception in thread "main" org.apache.hadoop.ipc.RemoteException(org.apache.hadoop.security.authorize.AuthorizationException): User: load_test@LAB.COM is not allowed
to impersonate load_test
    at org.apache.hadoop.ipc.Client.call(Client.java:1468)
    at org.apache.hadoop.ipc.Client.call(Client.java:1399)
    at org.apache.hadoop.ipc.ProtobufRpcEngine$Invoker.invoke(ProtobufRpcEngine.java:232)
    at com.sun.proxy.$Proxy14.getListing(Unknown Source)
    at org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.getListing(ClientNamenodeProtocolTranslatorPB.java:554)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:606)
    at org.apache.hadoop.io.retry.RetryInvocationHandler.invokeMethod(RetryInvocationHandler.java:187)
    at org.apache.hadoop.io.retry.RetryInvocationHandler.invoke(RetryInvocationHandler.java:102)
    at com.sun.proxy.$Proxy15.getListing(Unknown Source)
    at org.apache.hadoop.hdfs.DFSClient.listPaths(DFSClient.java:1963)
    at org.apache.hadoop.hdfs.DFSClient.listPaths(DFSClient.java:1948)
    at org.apache.hadoop.hdfs.DistributedFileSystem.listStatusInternal(DistributedFileSystem.java:693)
    at org.apache.hadoop.hdfs.DistributedFileSystem.access$600(DistributedFileSystem.java:105)
    at org.apache.hadoop.hdfs.DistributedFileSystem$15.doCall(DistributedFileSystem.java:755)
    at org.apache.hadoop.hdfs.DistributedFileSystem$15.doCall(DistributedFileSystem.java:751)
    at org.apache.hadoop.fs.FileSystemLinkResolver.resolve(FileSystemLinkResolver.java:81)
    at org.apache.hadoop.hdfs.DistributedFileSystem.listStatus(DistributedFileSystem.java:751)
    at com.sgm.portal.hive.dao.JDBCtest2.main(JDBCtest2.java:36)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:606)
    at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
    at org.apache.hadoop.util.RunJar.main(RunJar.java:136)

```

## Java 代码如下：

```

public static void main(String[] args) throws Exception {
    System.setProperty("java.security.krb5.conf", "/etc/krb5.conf");
    System.setProperty("java.security.auth.login.config", "/tmp/jaas.conf");
    System.setProperty("javax.security.auth.useSubjectCredsOnly", "false");
    System.setProperty("sun.security.jgss.debug", "true");
    LoginContext loginContext = null;
    try {
        loginContext = new LoginContext("com.sun.security.jgss.initiate"); // 和jaas中的配置一致
        loginContext.login();
        FileSystem fs = FileSystem.get(new Configuration());
        Path path = new Path("/home/hdfs/portal");
        FileStatus[] files = fs.listStatus(path);
        for (FileStatus fileStatus : files) {
            System.out.println(fileStatus);
        }
    } catch (LoginException e) {
        e.printStackTrace();
    }
}

```

## 问题原因：

之前常用的 Jaas 方式访问 kerberos 集群，在 jdbc 等场景下没有任何问题，但是可能不适用于访问 hdfs。事实上，hadoop 提供了专业的 UserGroupInformation 类，可以代替 Jaas。

解决方案：

使用 hadoop 自带 UserGroupInformation

```
try {
    Configuration conf = new Configuration();
    conf.set("hadoop.security.authentication", "Kerberos");
    UserGroupInformation.setConfiguration(conf);

    UserGroupInformation.loginUserFromKeytab("user1@HADOOP.COM", "./user1.keytab")
    conf.addResource("./hdfs-site.xml");
    conf.addResource("./core-site.xml");
    FileSystem fs = FileSystem.get(conf);
    Path p = new Path("/user/user1");
    FileStatus[] files = fs.listStatus(p);
    for(FileStatus fileStatus: files){
        System.out.println(fileStatus);
    }

} catch (IOException e1) {
    // TODO Auto-generated catch block
    e1.printStackTrace();
}

Connection con = null;
String sql = "show databases";
try {
    Class.forName(driver_name);
    con = DriverManager.getConnection(url);
    Statement stmt = con.createStatement();
```

该接口同样适用于 hive jdbc 场景。

问题 4：如何在 hue 中添加 Role？

问题现象：

使用 hive 用户登陆 hue 之后，无法在 hue 中 sentry table 页面中找到创

建 role 的页面。但是 hive 用户已经在后台通过 beeline 设置了管理员的权限。

#### 问题原因：

在 Sentry 中有配置 `sentry.service.admin.group`，CDH 中默认配置是 `hive`，`impala`，`hue`。只有属于这几个 Group 的用户才具备 Sentry Server 的管理员权限。在 hue 中获得 Sentry 的管理员权限，需要用户同时满足在 hue 中属于这几个 Group，以及在 Sentry 中属于这几个 Group。即 hive 用户需要在 Sentry 中具备管理员的角色，同时需要在 hue 中属于 hue 中的 group `hive`。

#### 解决方案：

1 在 hive 中创建 Group Hive

**HUE** **Query Editors** **Metastore Manager** **Workflows** **Security**

**User Admin** **Users** **Groups** **Permissions**

### Hue Groups - Create group

Name

members ☐ Select all

**A**

☐ admin

**H**

☒ hive

**S**

☐ sample

**U**

☐ user1

permissions ☒ Select all

**A**

☒ about.access:Launch this application(2)

**B**

☒ beeswax.access:Launch this application(3)

**F**

☒ filebrowser.access:Launch this application(4)

**H**

☒ hbase.write:Allow writing in the HBase app.(5)

2 完成之后重新使用 hive 用户登陆。可以在下图右侧+号中创建新的

# Role.

