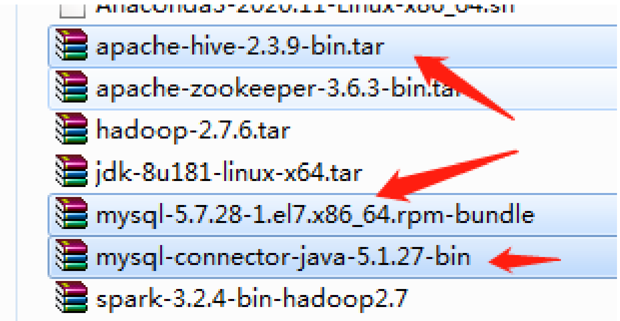
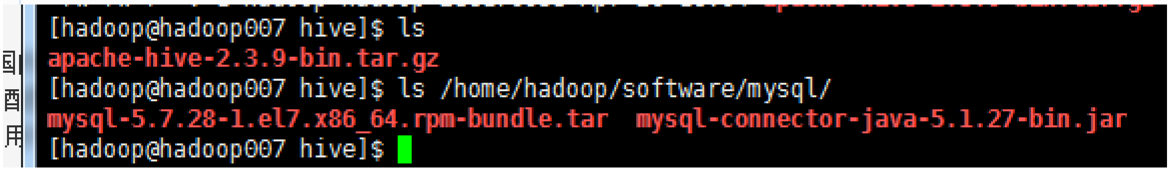
hive安装

hive安装

1、上传三个所需的包到007服务器的software下（hive只需要在主节点安装配置即可）



Anaconda3-2020.11-Linux-x86_64.sh
apache-hive-2.3.9-bin.tar
apache-zookeeper-3.6.3-bin.tar
hadoop-2.7.6.tar
jdk-8u181-linux-x64.tar
mysql-5.7.28-1.el7.x86_64.rpm-bundle
mysql-connector-java-5.1.27-bin
spark-3.2.4-bin-hadoop2.7



```
[hadoop@hadoop007 hive]$ ls
apache-hive-2.3.9-bin.tar.gz
[hadoop@hadoop007 hive]$ ls /home/hadoop/software/mysql/
mysql-5.7.28-1.el7.x86_64.rpm-bundle.tar  mysql-connector-java-5.1.27-bin.jar
[hadoop@hadoop007 hive]$ 
```

2、解压hive安装包。

`tar -zxvf apache-hive-2.3.9-bin.tar.gz -C /home/hadoop/`

3、修改hive名称（文件名太长不好操作）

`mv apache-hive-2.3.9-bin/ hive-2.39`

4、添加环境变量

`vim /etc/profile.d/my_env.sh`

添加内容：

#HIVE_HOME

export HIVE_HOME=/home/hadoop/hive-2.39

export PATH=$PATH:$HIVE_HOME/bin

更新环境变量

source /etc/profile

5、安装hive元数据存储地方MySQL

5.1、检查当前系统是否安装过 MySQL

```
rpm -qa|grep mariadb
```

5.2、如果有卸载掉，

卸载命令：`sudo rpm -e --nodeps mariadb-libs`

5.3、解压MySQL安装包

```
tar -xf mysql-5.7.28-1.el7.x86_64.rpm-bundle.tar
```

5.4、安装需要的rpm包

```
sudo rpm -ivh mysql-community-common-5.7.28-1.el7.x86_64.rpm
sudo rpm -ivh mysql-community-libs-5.7.28-1.el7.x86_64.rpm
sudo rpm -ivh mysql-community-libs-compat-5.7.28-1.el7.x86_64.rpm
sudo rpm -ivh mysql-community-client-5.7.28-1.el7.x86_64.rpm
```

如果Linux是最小化安装，需要下载依赖在执行下面这个包的安装！

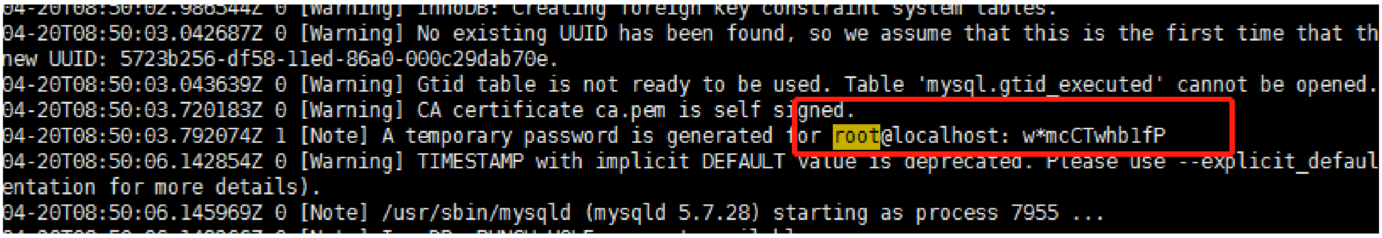
sudo yum install -y libaio
sudo yum install -y net-tools
sudo rpm -ivh mysql-community-server-5.7.28-1.el7.x86_64.rpm

5.5、初始化数据库

```
sudo mysqld --initialize --user=mysql
```

5.6、查看初始化生成的root密码（初始化后会自动生成一个随机的root密码在/var/log/mysqld.log文件中）

```
cat /var/log/mysqld.log
```



```
04-20T08:50:02.988544Z 0 [Warning] InnoDB: Creating foreign key constraint system tables.
04-20T08:50:03.042687Z 0 [Warning] No existing UUID has been found, so we assume that this is the first time that th
new UUID: 5723b256-df58-11ed-86a0-000c29dab70e.
04-20T08:50:03.043639Z 0 [Warning] Gtid table is not ready to be used. Table 'mysql.gtid_executed' cannot be opened.
04-20T08:50:03.720183Z 0 [Warning] CA certificate ca.pem is self signed.
04-20T08:50:03.792074Z 1 [Note] A temporary password is generated for root@localhost: w*mcCTwhb1fP
04-20T08:50:06.142854Z 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaul
entation for more details).
04-20T08:50:06.145969Z 0 [Note] /usr/sbin/mysqld (mysqld 5.7.28) starting as process 7955 ...
```

5.7、启动MySQL服务

```
sudo systemctl start mysqld
```

5.8、登陆数据库（输入在/var/log/mysqld.log中的密码）

```
mysql -uroot -p
```

5.9、修改root密码（一定要先修改密码，不然操作MySQL会出现问题）

```
set password = password("123456");
```

5.9.1、修改密码校验强度。(如果上一步修改密码不成功，提示密码过于简单的话再执行此步骤)。

```
set global validate_password_policy=0;
set global validate_password_length=3;
```

5.10、修改mysql库下的user表中的root用户允许任意ip连接，并刷新

```
update mysql.user set host='%' where user='root';
flush privileges;
```

6、配置hive参数

6.1、拷贝MySQL驱动到hive的lib下

`cp mysql-connector-java-5.1.27-bin.jar /home/hadoop/hive-2.39/lib/`

添加hive-site.xml文件

`vim /home/hadoop/hive-2.39/conf/hive-site.xml`

添加内容：

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
<!-- jdbc 连接的 URL -->
<property>
<name>javax.jdo.option.ConnectionURL</name>
<value>jdbc:mysql://hadoop007:3306/metastore?useSSL=false</value>
</property>

<!-- jdbc 连接的 Driver-->
<property>
<name>javax.jdo.option.ConnectionDriverName</name>
<value>com.mysql.jdbc.Driver</value>
</property>

<!-- jdbc 连接的 username-->
<property>
<name>javax.jdo.option.ConnectionUserName</name>
<value>root</value>
</property>

<!-- jdbc 连接的 password -->
<property>
<name>javax.jdo.option.ConnectionPassword</name>
<value>123456</value>
</property>

<!-- Hive 元数据存储版本的验证 -->
```

```
<property>
<name>hive.metastore.schema.verification</name>
<value>false</value>
</property>

<!--元数据存储授权-->
<property>
<name>hive.metastore.event.db.notification.api.auth</name>
<value>false</value>
</property>

<!-- Hive 默认在 HDFS 的工作目录 -->
<property>
<name>hive.metastore.warehouse.dir</name>
<value>/user/hive/warehouse</value>
</property>
</configuration>
```

6.2、登陆MySQL数据库创建Metastore（其实不创建hive在初始化的时候也会自动创建）

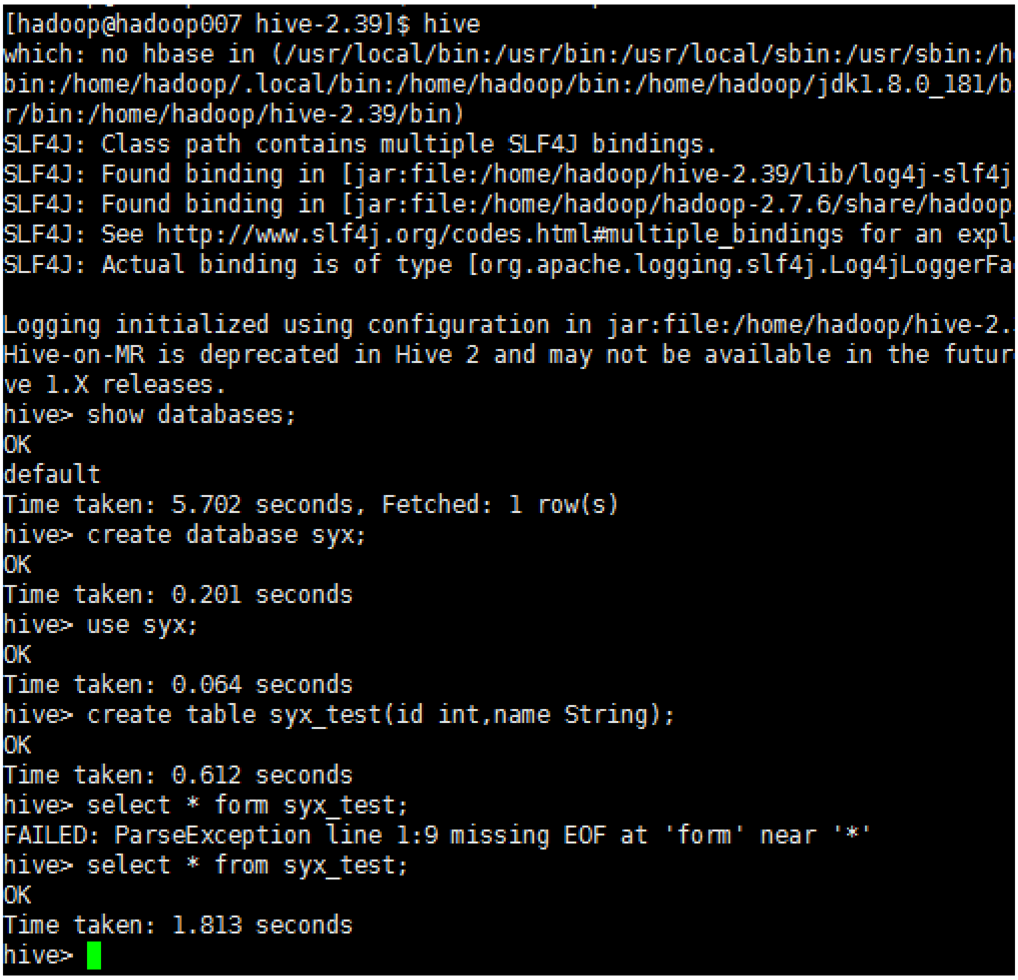
`mysql -uroot -p123456`

`create database metastore;`

6.3、初始化Hive元数据库

`/home/hadoop/hive-2.39/bin/schematool -initSchema -dbType mysql - verbose`

6.4、启动hive测试，因为已经配置了hive的环境变量，直接敲hive即可进入hive的客户端进行hiveSql操作。



```
[hadoop@hadoop007 hive-2.39]$ hive
which: no hbase in (/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/h
bin:/home/hadoop/.local/bin:/home/hadoop/bin:/home/hadoop/jdk1.8.0_181/b
r/bin:/home/hadoop/hive-2.39/bin)
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/hadoop/hive-2.39/lib/log4j-slf4j
SLF4J: Found binding in [jar:file:/home/hadoop/hadoop-2.7.6/share/hadoop
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an expl
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFa

Logging initialized using configuration in jar:file:/home/hadoop/hive-2.
Hive-on-MR is deprecated in Hive 2 and may not be available in the futur
ve 1.X releases.
hive> show databases;
OK
default
Time taken: 5.702 seconds, Fetched: 1 row(s)
hive> create database syx;
OK
Time taken: 0.201 seconds
hive> use syx;
OK
Time taken: 0.064 seconds
hive> create table syx_test(id int,name String);
OK
Time taken: 0.612 seconds
hive> select * form syx_test;
FAILED: ParseException line 1:9 missing EOF at 'form' near '*'
hive> select * from syx_test;
OK
Time taken: 1.813 seconds
hive> 
```