

大数据技术-第八章：Hive数据仓库 Hive安装配置



CONTENTS

01. 下载安装的文件

02. 安装MySQL数据库

03. 安装Hive



01

下载安装的文件



» 下载安装的文件

需要3个文件:

- (1). Hive下载: Hive下载可以通过地址<http://archive.apache.org/dist/hive/hive-2.0.0/apache-hive-2.0.0-bin.tar.gz>来获取apache-hive-2.0.0-bin.tar.gz文件, 安装 Hive组件需要与 Hadoop 环境适配。
- (2). 下载MySQL文件: MySQL下载可以通过地址https://cdn.mysql.com//archives/mysql-5.7/mysql-5.7.18-1.el7.x86_64.rpm-bundle.tar来获取mysql-5.7.18-1.el7.x86_64.rpm-bundle.tar文件;
- (3).MySQL的JDBC驱动包:mysql-connector-java-5.1.48.jar。

上述3个文件已经上传到master机器的/opt/software/目录下。

02

安装MySQL数据库



» 安装MySQL数据库

一、卸载MariaDB数据库

我们采用mysql库，卸载掉系统自带的MariaDB数据库。需要使用 root 用户登录，执行如下操作步骤：

(1). 关闭Linux系统防火墙，并将防火墙设定为系统开机并不自动启动。

```
[root@master ~]# systemctl stop firewalld # 关闭防火墙服务
```

```
[root@master ~]# systemctl disable firewalld # 设置防火墙服务开机不启动
```

(2). 卸载Linux系统自带的MariaDB。

首先，查看Linux系统中MariaDB的安装情况。

```
[root@ master ~]# rpm -qa | grep mariadb ## 查询已安装的 mariadb 软件包
```

```
mariadb-libs-5.5.56-2.el7.x86_64
```

以上结果显示Linux系统中已经安装了mariadb-libs-5.5.56-2.el7.x86_64软件包，需要将其卸载。

其次，卸载MariaDB软件包。

```
[root@master ~]# rpm -e --nodeps mariadb-libs-5.5.56-2.el7.x86_64 ## 卸载 mariadb 软件包
```

» 安装MySQL数据库

二、安装MySQL数据库

(1). 按如下顺序依次按照MySQL数据库的mysql-community-common、mysql-community-libs、mysql-community-client、mysql-community-server软件包四个文件。

```
[root@master ~]# mkdir /opt/software/mysql ## 创建目录mysql
```

```
[root@master ~]# tar -xvf /opt/software/mysql-5.7.18-1.el7.x86_64.rpm-bundle.tar -C /opt/software/mysql ## 解压文件mysql-5.7.18到/opt/software/mysql目录
```

```
[root@master ~]# cd /opt/software/mysql/ ## 进入目录/opt/software/mysql/
```

##安装MySQL软件包，按如下顺序依次安装

```
[root@master ~]# rpm -ivh mysql-community-common-5.7.18-1.el7.x86_64.rpm
```

```
[root@master ~]# rpm -ivh mysql-community-libs-5.7.18-1.el7.x86_64.rpm
```

```
[root@master ~]# rpm -ivh mysql-community-client-5.7.18-1.el7.x86_64.rpm
```

在线安装perl支持

```
[root@master ~]# yum install perl
```

##安装MySQL软件包，按如下顺序依次安装

```
[root@master ~]# rpm -ivh mysql-community-server-5.7.18-1.el7.x86_64.rpm
```

安装MySQL数据库

- (2) 修改MySQL数据库配置，在/etc/my.cnf文件中添加如表6-1所示的MySQL数据库配置项。

字段	配置值	字段说明
default-storage-engine	innodb	设置innodb为默认的存储引擎。
innodb_file_per_table	-	设置每个表的数据单独保存，而不是统一保存在innodb系统表空间中。单独保存有方便管理和提升性能两方面优势。
collation-server	utf8_general_ci	设置支持中文编码字符集。
init-connect	'SET NAMES utf8'	设置用户登录到数据库之后，在执行第一次查询之前执行SET NAMES utf8命令，将使用的字符编码设定为utf8。
character-set-server	utf8	将MySQL服务器字符集设定为utf8。

将以下配置信息添加到/etc/my.cnf文件symbolic-links=0配置信息的下方。

default-storage-engine=innodb

innodb_file_per_table

collation-server=utf8_general_ci

init-connect='SET NAMES utf8'

character-set-server=utf8

» 安装MySQL数据库

添加完成后/etc/my.cnf文件内容如下所示。

```
21 socket=/var/lib/mysql/mysql.sock
22
23 # Disabling symbolic-links is recommended to prevent ass
  S
24 symbolic-links=0
25
26 default-storage-engine=innodb
27 innodb_file_per_table
28 collation-server=utf8_general_ci
29 init-connect='SET NAMES utf8'
30 character-set-server=utf8
31
32 log-error=/var/log/mysqld.log
```

» 安装MySQL数据库

(3) 启动MySQL数据库。

```
[root@master ~]# systemctl start mysqld
```

(4) 查询MySQL数据库状态。mysqld进程状态为active (running)，则表示MySQL数据库正常运行。如果mysqld进程状态为failed，则表示MySQL数据库启动异常。此时需要排查/etc/my.cnf文件。

```
[root@master ~]# systemctl status mysqld
```

● mysqld.service - MySQL Server

Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)

Active: active (running) since — 2020-05-11 16:28:27 CST; 1h 26min ago

Docs: man:mysqld(8)

<http://dev.mysql.com/doc/refman/en/using-systemd.html>

Process: 941 ExecStart=/usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid \$MYSQLD_OPTS (code=exited, status=0/SUCCESS)

Process: 911 ExecStartPre=/usr/bin/mysqld_pre_systemd (code=exited, status=0/SUCCESS)

Main PID: 944 (mysqld)

CGroup: /system.slice/mysqld.service

└─944 /usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid

(5) 查询MySQL数据库默认密码。

MySQL数据库安装后的默认密码保存在/var/log/mysqld.log文件中，在该文件中以password关键字搜索默认密码。

```
[root@master ~]# cat /var/log/mysqld.log | grep password
```

```
2021-02-27T14:03:18.104597Z 1 [Note] A temporary password is generated for
```

```
root@localhost: bFpp?boXo5Q>      #默认密码为bFpp?boXo5Q>
```

MySQL数据库是安装后随机生成的，所以每次安装后生成的默认密码不相同。

(6) MySQL数据库初始化。

执行mysql_secure_installation命令初始化MySQL数据库，初始化过程中需要设定数据库root用户登录密码，密码需符合安全规则，包括大小写字符、数字和特殊符号，可设定密码为Password123\$。

在进行MySQL数据库初始化过程中会出现以下交互确认信息：

- 1) Change the password for root ? ((Press y|Y for Yes, any other key for No) 表示是否更改root用户密码，在键盘输入y和回车。
- 2) Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No)表示是否使用设定的密码继续，在键盘输入y和回车。
- 3) Remove anonymous users? (Press y|Y for Yes, any other key for No)表示是否删除匿名用户，在键盘输入y和回车。
- 4) Disallow root login remotely? (Press y|Y for Yes, any other key for No)表示是否拒绝root用户远程登录，在键盘输入n和回车，表示允许root用户远程登录。
- 5) Remove test database and access to it? (Press y|Y for Yes, any other key for No)表示是否删除测试数据库，在键盘输入y和回车。
- 6) Reload privilege tables now? (Press y|Y for Yes, any other key for No) 表示是否重新加载授权表，在键盘输入y和回车。

» 安装MySQL数据库

(7) 添加root用户从本地和远程访问MySQL数据库表单的授权。

```
mysql> grant all privileges on *.* to root@'localhost' identified by 'Password123$';  
Query OK, 0 rows affected, 1 warning (0.01 sec)
```

添加root本地授权

```
mysql> grant all privileges on *.* to root@'%' identified by 'Password123$';  
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

添加root用户远程访问授权

```
mysql> flush privileges;  
Query OK, 0 rows affected (0.00 sec)
```

刷新授权

```
mysql> select user,host from mysql.user where user='root';
```

查询root用户授权情况

```
+-----+-----+  
| user | host      |  
+-----+-----+  
| root | %         |  
| root | localhost |  
+-----+-----+  
2 rows in set (0.00 sec)
```

```
mysql> exit;  
Bye
```

退出MySQL数据库

03

安装Hive



» 安装Hive

在master节点配置。

(1). 解压安装文件。使用root用户完成操作。

将Hive安装包/opt/software/apache-hive-2.0.0-bin.tar.gz路解压到
/usr/local/src路径下

```
[root@master ~]# tar -zxvf /opt/software/apache-hive-2.0.0-bin.tar.gz -C  
/usr/local/src
```

将解压后的apache-hive-2.0.0-bin 文件夹更名为hive;

```
[root@master ~]# mv /usr/local/src/apache-hive-2.0.0-bin /usr/local/src/hive
```

修改 hive 目录归属用户和用户组为 hadoop

```
[root@master ~]# chown -R hadoop:hadoop /usr/local/src/hive
```

» 安装Hive

(2). 设置Hive环境变量并使其生效。

```
[root@master ~]# vi /etc/profile # 在文件末尾追加以下配置内容
```

```
# set hive environment
```

```
export HIVE_HOME=/usr/local/src/hive
```

```
export PATH=$PATH:$HIVE_HOME/bin
```

```
source /etc/profile # 使环境变量配置生效
```

(3) 修改Hive组件配置文件。

切换到hadoop用户执行以下对Hive组件的配置操作。

将/usr/local/src/hive文件夹下hive-default.xml.template文件，更名为hive-site.xml。

```
[root@master ~]# su - hadoop
```

```
[hadoop@master ~]$ cp /usr/local/src/hive/conf/hive-default.xml.template  
/usr/local/src/hive/conf/hive-site.xml
```

(4) 通过vi编辑器修改hive-site.xml文件实现Hive连接MySQL数据库，并设定Hive临时文件存储路径。

```
[hadoop@master ~]$ vi /usr/local/src/hive/conf/hive-site.xml
```

1) 设置MySQL数据库连接。

```
<name>javax.jdo.option.ConnectionURL</name>
```

```
<value>jdbc:mysql://master:3306/hive?createDatabaseIfNotExist=true&useSSL=false</value>
```

```
<description>JDBC connect string for a JDBC metastore</description>
```

2) 配置MySQL数据库root的密码。

```
<property>
```

```
<name>javax.jdo.option.ConnectionPassword</name>
```

```
<value>Password123$</value>
```

```
<description>password to use against metastore database</description>
```

```
</property>
```

3) 配置数据库驱动。

```
<property>  
<name>javax.jdo.option.ConnectionDriverName</name>  
<value>com.mysql.jdbc.Driver</value>  
<description>Driver class name for a JDBC metastore</description>  
</property>
```

4) 配置数据库用户名javax.jdo.option.ConnectionUserName为root。

```
<property>  
<name>javax.jdo.option.ConnectionUserName</name>  
<value>root</value>  
<description>Username to use against metastore database</description>  
</property>
```

» 安装Hive

5) 将以下位置的`${system:java.io.tmpdir}/${system:user.name}`替换为“/usr/local/src/hive/tmp”目录及其子目录。

需要替换以下4处配置内容：

```
<name>hive.querylog.location</name>
```

```
<value>/usr/local/src/hive/tmp</value>
```

```
<description>Location of Hive run time structured log file</description>
```

```
<name>hive.exec.local.scratchdir</name>
```

```
<value>/usr/local/src/hive/tmp</value>
```

```
<name>hive.downloaded.resources.dir</name>
```

```
<value>/usr/local/src/hive/tmp/resources</value>
```

```
<name>hive.server2.logging.operation.log.location</name>
```

```
<value>/usr/local/src/hive/tmp/operation_logs</value>
```

6) 在Hive安装目录中创建临时文件夹tmp。

```
[hadoop@master ~]$ mkdir /usr/local/src/hive/tmp
```

至此，Hive数据仓库的安装和配置完成。

Turing AI 万维图灵 | 大数据系列课程

大数据

BIG
DATA

智 / 能 / 科 / 技 放 / 眼 / 未 / 来

