

Exp: 5**Installation of Hive****Aim:**

To Download and install Hive, Understanding Startup scripts, Configuration files.

Procedure:**Step 1: Download and extract it**

Download the Apache hive and extract it use tar, the commands given below:

```
$wgethttps://downloads.apache.org/hive/hive-3.1.2/apache-hive-3.1.2-bin.tar.gz
```

```
$ tar -xvf apache-hive-3.1.2-bin.tar.gz
```

Step 2: Place different configuration properties in Apache Hive

In this step, we are going to do two things o Placing Hive Home path in bashrc file

```
$nano .bashrc
```

And append the below lines in it

```
export HIVE_HOME="/home/haresh/apache-hive-3.1.2-bin"
export HIVE_CONF_DIR=$HIVE_HOME/conf
export PATH=$PATH:$HIVE_HOME/bin
export HADOOP_USER_CLASSPATH_FIRST=true
```

2. Exporting Hadoop path in Hive-config.sh (To communicate with the Hadoop eco system we are defining Hadoop Home path in hive config field) Open the hiveconfig.sh as shown in below

```
$cd apache-hive-3.1.2-bin/bin
```

```
$cp hive-env.sh.template hive-env.sh
```

```
$nano hive-env.sh
```

Append the below commands on it export

```
HADOOP_HOME=/home/Hadoop/Hadoop export
```

```
HIVE_CONF_DIR=/home/Hadoop/apache-hive-3.1.2/conf
```

```
export HADOOP_HOME=$HADOOP_HOME
export HIVE_CONF_DIR=$HIVE_CONF_DIR
```

Step 3: Install mysql

1. Install mysql in Ubuntu by running this command:

```
$sudo apt update
```

```
$sudo apt install mysql-server
```

2. Alter username and password for MySQL by running below commands:

```
$sudo mysql
```

Opens command line interface for MySQL and run the below SQL queries to change username and set password

```
mysql> SELECT user, host, plugin FROM mysql.user WHERE user = 'root';
```

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH 'mysql_native_password' BY
```

```
'your_new_password'; mysql>
```

FLUSH PRIVILEGES;

```
inflating: DA-301-main/da_lab/exp41/pig_1724404801470.log
inflating: DA-301-main/da_lab/exp41/pig_1724404110819.log
inflating: DA-301-main/da_lab/exp41/pig_1724404339867.log
inflating: DA-301-main/da_lab/exp41/up.py
inflating: DA-301-main/da_lab/hadoop-streaming-3.3.6.jar
creating: DA-301-main/hadoop/
creating: DA-301-main/hadoop/etc/
creating: DA-301-main/hadoop/etc/hadoop/
inflating: DA-301-main/hadoop/etc/hadoop/core-site.xml
inflating: DA-301-main/hadoop/etc/hadoop/hadoop-env.sh
inflating: DA-301-main/hadoop/etc/hadoop/hdfs-site.xml
inflating: DA-301-main/hadoop/etc/hadoop/mapred-site.xml
inflating: DA-301-main/hadoop/etc/hadoop/yarn-site.xml
osboxes@fedora:~/Downloads$ cd DA-301-main/
osboxes@fedora:~/Downloads/DA-301-main$ cp -r apache-hive-3.1.2-bin/ -
osboxes@fedora:~/Downloads/DA-301-main$ cd -
osboxes@fedora:~$ ls
apache-hive-3.1.2-bin  dalah  demo_pig.pig  Desktop  Documents  Downloads  hadoop  Music  Pictures  pig  Public  sample.txt  Templates  udf_example.pig  uppercase_udf.py  Videos
osboxes@fedora:~$ nano .bashrc
osboxes@fedora:~$ source .bashrc
osboxes@fedora:~$ mv apache-hive-3.1.2-bin/ hive
osboxes@fedora:~$ ls
dalah  demo_pig.pig  Desktop  Documents  Downloads  hadoop  hive  Music  Pictures  pig  Public  sample.txt  Templates  udf_example.pig  uppercase_udf.py  Videos
osboxes@fedora:~$ cd hive
osboxes@fedora:~/hive$ chmod +x -R *
osboxes@fedora:~/hive$ sudo mysql
[sudo] password for osboxes:
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/var/lib/mysql/mysql.sock' (2)
osboxes@fedora:~/hive$ mysql -u root -p
Enter password:
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/var/lib/mysql/mysql.sock' (2)
osboxes@fedora:~/hive$ mysql -u root -p
Enter password:
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/var/lib/mysql/mysql.sock' (2)
osboxes@fedora:~/hive$ mysql
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/var/lib/mysql/mysql.sock' (2)
osboxes@fedora:~/hive$ systemctl status mysqld
o mysqld.service - MySQL 8.0 database server
   Loaded: loaded (/usr/lib/systemd/system/mysqld.service; disabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/service.d
           └─10-timeout-abort.conf
   Active: inactive (dead)
osboxes@fedora:~/hive$ systemctl enable mysqld
Created symlink /etc/systemd/system/multi-user.target.wants/mysqld.service → /usr/lib/systemd/system/mysqld.service.
osboxes@fedora:~/hive$ systemctl start mysqld
osboxes@fedora:~/hive$ sudo mysql
```

Step 4: Config hive-site.xml

Config the hive-site.xml by appending this xml code and change the username and password according to your MySQL.

```
$cd apache-hive-3.1.2-bin/bin
```

```
$cp hive-default.xml.template hive-site.xml
```

```
$nano hive-site.xml
```

Append these lines into it

Replace root as your username of MySQL

Replace your_new_password as with your password of MySQL

```
<configuration>
```

```
<property>
```

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

```
<value>jdbc:mysql://localhost/metastore?createDatabaseIfNotExist=true</value>
```

```
</property>
```

```
<property>
```

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

```
<value>com.mysql.cj.jdbc.Driver</value>
```

```
</property>
```

```
<property>
```

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

```
<value>root</value>
```

```
</property>
```

```
<property>
```

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

```
<value>your_new_password</value>
```

```
</property>
```

```
<property>
```

```
<name>datanucleus.autoCreateSchema</name>
```

```

<value>true</value>
</property>
<property>
<name>datanucleus.fixedDatastore</name>
<value>true</value>
</property>
<property>
<name>datanucleus.autoCreateTables</name>
<value>True</value>
</property>
</configuration>

```

Step 5: Setup MySQL java connector:

First, you'll need to download the MySQL Connector/J, which is the JDBC driver for MySQL. You can download it from the below link https://drive.google.com/file/d/1QFhB7Kvcat7a4LzDRe6GcmZva1yAxKz-/view?usp=drive_link

Copy the downloaded MySQL Connector/J JAR file to the Hive library directory. By default, the Hive library directory is usually located at /path/to/apache-hive-3.1.2/lib/ on Ubuntu. Use the following command to copy the JAR file:

```
$sudo cp /path/to/mysql-connector-java-8.0.15.jar /path/to/apache-hive-3.1.2/lib/
```

Replace /path/to/ with the actual path to the JAR file.

Step 6: Initialize the Hive Metastore Schema:

Run the following command to initialize the Hive metastore schema:

```
$$HIVE_HOME/bin/schematool -initSchema -dbTypemysql
```

7: Start hive:

You can test Hive by running the Hive shell: Copy code hive You should be able to run Hive queries, and metadata will be stored in your MySQL database.

\$hive

```
osboxes@fedora:~/hive/conf$ gedit hive-site.xml
osboxes@fedora:~/hive/conf$ rm hive-site.xml
osboxes@fedora:~/hive/conf$ cp ~/Downloads/DA-301-main/apache-hive-3.1.2-bin/conf/hive-site.xml .
osboxes@fedora:~/hive/conf$ gedit hive-site.xml
osboxes@fedora:~/hive/conf$ schematool -initSchema -dbType mysql
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/osboxes/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/osboxes/hadoop/share/hadoop/common/lib/slf4j-reload4j-1.7.36.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.apache.logging.slf4j.Log4jLoggerFactory]
Metastore connection URL:      jdbc:mysql://localhost/metastore?createDatabaseIfNotExist=true&serverTimezone=UTC&useSSL=false&allowPublicKeyRetrieval=true
Metastore Connection Driver :   com.mysql.cj.jdbc.Driver
Metastore connection User:     root
Starting metastore schema initialization to 3.1.0
Initialization script hive-schema-3.1.0.mysql.sql
```

Result:

Thus, the Apache Hive installation is completed successfully.