Sqoop usage in Hadoop Distributed File System and Observations to Handle Common Errors

K. UmaPavan Kumar, S V N Srinivasu, M N. Nachappa

***Abstract:** The Hadoop framework provides a way of storing and processing the huge amounts of the data. The social media like Facebook, twitter and amazon uses Hadoop eco system tools so as to store the data in Hadoop distributed file system and to process the data Map Reduce (MR). The current work describes the usage of Sqoop in the process of import and export with HDFS. The work involves various possible import/export commands supported by the tool Sqoop in the eco system of Hadoop. The importance of the work is to highlight the common errors while installing Sqoop and working with Sqoop. Many developers and researchers were using Sqoop so as to perform the import/export process and to handle the source data in the relational format. In the current work the connectivity between mysql and sqoop were presented and various commands usage along with the results were presented. The outcome of the work is for each command the possible errors encountered and the corresponding solution is mentioned. The common configuration settings we have to follow so as to handle the Sqoop without any errors is also mentioned.*

***Keywords**: Import, Export, Hadoop, Sqoop, Configuration, Relational Source.*

I. INTRODUCTION

Apache Sqoop is a tool which helps to import/export the data from relational data bases to HDFS and vice-versa. Most of the reporting and data visualization tools built on top of RDBMS. So all the developers need a procedure to transfer the data from HDFS to RDBMS [1]. The expectation here is parallel loads of the data so as to speed up the task. The same way fetching the data from RDBMS to HDFS is required so as to perform the huge data processing with the help of MR and applicability of the analytics to come up with various predictions is possible. The problem with the usage of scripting to perform the above mentioned activities is time consuming and scripts were in efficient to perform import/export. So as to solve the above issues and to perform the import/exports efficiently in the context of parallel mode with the usage of MR and less time than scripting the Sqoop is the best tool. Sqoop efficiently work with MySQL, Oracle and HDFS. Integration of Hive and Pig Latin is also allowed in Sqoop. The best feature of Sqoop is can be integrated with Oozie so as to automate the scheduling process. The organization of the article is as follows in section II Sqoop working and architecture was discussed. In Section III various Sqoop commands to Import/ Export wereexplained. In Section IV along with Sqoop commands implementation and common errors faced in the import/export process were described. In Section V conclusions of the work were mentioned.

II. SQOOP ARCHITECTURE AND WORKING

Sqoop runs on top of Hadoop Cluster, so before installation and running of the Sqoop, the machine should contain the Hadoop Configuration on top of the Hadoop one can install Sqoop by downloading the compatible version of Sqoop with Hadoop version [2]. While performing the import or export process Sqoop uses mapper to slice the incoming data. Each slice is partitioned with the help of Map jobs and transfer the slice of the data to HDFS or RDBMS. One of the important aspect in Sqoop is the tool process the records in the context of type safe manner and uses DB meta data to infer the data types.

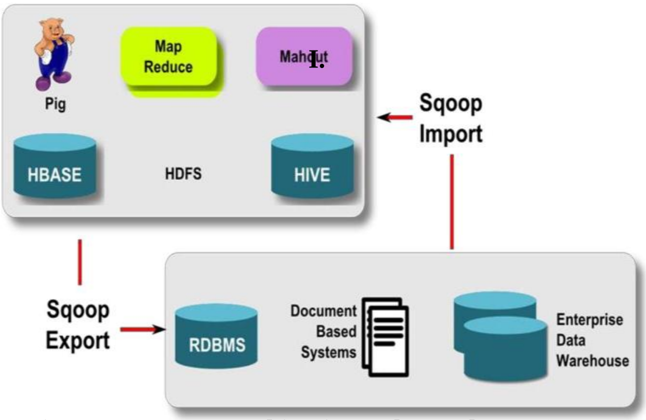


Figure 1: Sqoop Working in Hadoop Cluster (Source Apache).

From figure 1 we can observe that Sqoop helps us to import the data from RDBMS/DWH source to HDFS. The process of Export can be done from HDFS to any RDBMS/DWH with the integration of MR, Hive, Pig, HBase.

you are using *Word*, use either the Microsoft Equation Editor or the *MathType* add-on (http://www.mathtype.com) for equations in your paper (Insert | Object | Create New | Microsoft Equation *or*MathType Equation). “Float over text” should *not* be selected. The tools are part of Hadoop eco system MR is the processing of Hadoop jobs in parallel and distributed manner.Hive is a tool used to perform some storing and processing of the data with Hive Query Language [3].Pig is a tool provides customizations like joins, load, union and is a scripting kind of the language which simplifies the implementation by reducing the lines of the code. HBase is a tool with NOSQL context so as to support column family kind of the data without following indexing, concept of keys or normalizations. With all these mentioned tools Sqoop can be integrated to import/export the data

Revised Manuscript Received on November 30, 2020.

***** Correspondence Author
Dr K Uma Pavan, Assoicate professor CS IT, Jain Univrsity, Bangalore,
Dr. M N. Nachappa,Professor and Academic Head, School of CS and IT, Jain University,
Dr S.V.N. Srinivasu, Professor CSE, Narasaraopeta Engineering College (Autonomous), Narasaraopet, A.P

III. VARIOUS SQOOP COMMANDS IN THE DATA IMPORT/EXPORT

Sqoop works in the form of Import and Export concept with the RDBMS. Sqoop is having certain commands to work out and will give all the supported commands to exploit the strength of the Sqoop. Sqoop is not providing any separate prompt as that of hive and grunt shell, so directly from terminal of Unix/Linux we have to perform the tasks. But the prerequisite is to observe the Hadoop services running in the machine. The services required to work with Sqoop are Name Node whichprovides the meta data information of the HDFS and is a Master node givesthe instructions to Data Node [4].

Data Node is a slave daemon which is actual storage of the data. Job Tracker is a service which schedules the job and assigns the job to the task tracker. Task Tracker is a slave service that is the actual work horse. The backup of the Name Node is observed in the service as Secondary Name Node.

The following table gives the idea about the supported commands in Sqoop on top of Hadoop cluster. With the command Sqoop help we can get the various commands such as codegen, eval, import, export and import-all-tables etc.,

Sqoop provides a range of commands to integrate with HDFS, HIVE, PIG and while performing export and import of the table it depends on the Mapper tasks to complete the activity.

Table 1: Sqoop Supported Commands to perform various tasks.

Command in Sqoop	Description
Codegen	Code generation to interact with DB
Create-hive-table	Import Table into Hive
Eval	To execute the queries like update and Select and get the result
Export	Export the HDFS directory to RDBMS
Import	Import a table from RDBMS to HDFS
Import-all-tables	Import all the tables from RDBMS to HDFS
List-databases	List out the data bases in the selected RDBMS
List-tables	List out all the tables in the selected database of RDBMS
Version	Display version of the Sqoop running on top of Hadoop

Sqoop Implementation of Import/Export along with Common Errors.

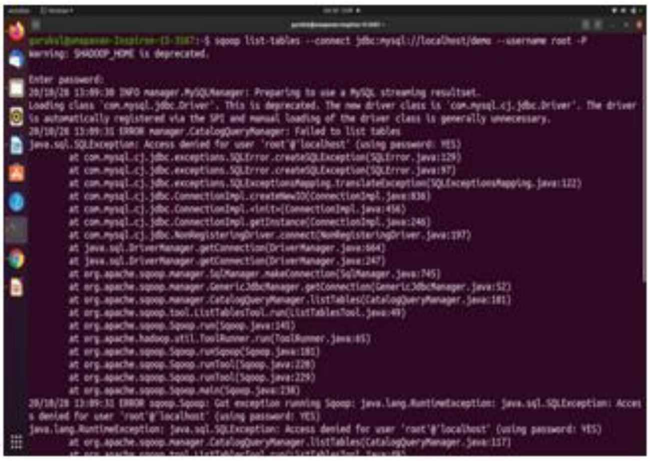
Sqoop as mentioned earlier is a tool to work with various kinds of the environments. The basic point here is establishing a connection with MYSQL and Hadoop is the initial point. The usage of the command involves two aspects one is localhost connectivity provided both Sqoop (in Hadoop Cluster) and MYSQL are running in the same machine, so we can make use of localhost in all the import/export commands [5].

The second context is if MYSQL is accessed via remote location in that case we can make use of ipconfig from the windows machine or ifconfig from the Unix/Linux based machines.

Observation1:

The most common error occurs in the working of localhost or ip/ifconfig is improper specification of the address of the machine so the sqoop-mysql connectivity becomes a major problem. Here the solution is to clearly identify the configuration details and specify the proper configuration value so that we can connect with MYSQL through Sqoop.

In continuation with this the username and password specification in the running of import/export command from Hadoop terminal. The user name is root (you can use) password here we need to mention the MYSQL installed version password then only we can able connect with the Sqoop. Most of the errors faced by the users in the connection process is username and password specification with MYSQL.



```
[illegible]-Inspiron-13-3167:~$ sqoop list-tables --connect jdbc:mysql://localhost/demo --username root -P
Warning: $HADOOP_HOME is deprecated.

Enter password:
20/10/28 13:09:30 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'. The driver
is automatically registered via the SPI and manual loading of the driver class is generally unnecessary.
20/10/28 13:09:31 ERROR manager.CatalogQueryManager: Failed to list tables
java.sql.SQLException: Access denied for user 'root'@'localhost' (using password: YES)
        at com.mysql.cj.jdbc.exceptions.SQLError.createSQLException(SQLError.java:129)
        at com.mysql.cj.jdbc.exceptions.SQLError.createSQLException(SQLError.java:97)
        at com.mysql.cj.jdbc.exceptions.SQLExceptionsMapping.translateException(SQLExceptionsMapping.java:122)
        at com.mysql.cj.jdbc.ConnectionImpl.createNewIO(ConnectionImpl.java:836)
        at com.mysql.cj.jdbc.ConnectionImpl.<init>(ConnectionImpl.java:456)
        at com.mysql.cj.jdbc.ConnectionImpl.getInstance(ConnectionImpl.java:246)
        at com.mysql.cj.jdbc.NonRegisteringDriver.connect(NonRegisteringDriver.java:197)
        at java.sql.DriverManager.getConnection(DriverManager.java:664)
        at java.sql.DriverManager.getConnection(DriverManager.java:247)
        at org.apache.sqoop.manager.SqlManager.makeConnection(SqlManager.java:745)
        at org.apache.sqoop.manager.GenericJdbcManager.getConnection(GenericJdbcManager.java:52)
        at org.apache.sqoop.manager.CatalogQueryManager.listTables(CatalogQueryManager.java:101)
        at org.apache.sqoop.tool.ListTablesTool.run(ListTablesTool.java:49)
        at org.apache.sqoop.Sqoop.run(Sqoop.java:145)
        at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:65)
        at org.apache.sqoop.Sqoop.runSqoop(Sqoop.java:181)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:220)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:229)
        at org.apache.sqoop.Sqoop.main(Sqoop.java:238)
20/10/28 13:09:31 ERROR sqoop.Sqoop: Got exception running Sqoop: java.lang.RuntimeException: java.sql.SQLException: Acces
s denied for user 'root'@'localhost' (using password: YES)
java.lang.RuntimeException: java.sql.SQLException: Access denied for user 'root'@'localhost' (using password: YES)
        at org.apache.sqoop.manager.CatalogQueryManager.listTables(CatalogQueryManager.java:117)
```

Figure 2: Result set error while connection MYSQL through Sqoop.

Observation 2:

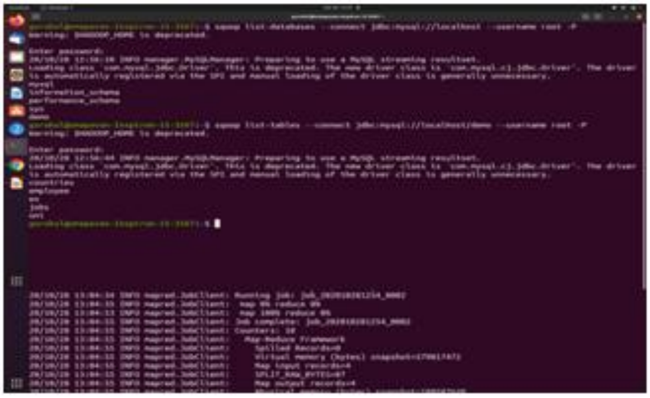
The Sqoop working is integration between HDFS and MYSQL, HIVE and Sqoop and HBase and Sqoop likewise there are distinct aspects were there in the connection management and processing of the queries. To handle the sqoop working in the proper way we need to have a .jar file which mysql-connector in the version compatible of java version of the Hadoop cluster [6]. The same .jar file(mysql-connector) must be there in sqoop library along with java library also.

Observation 3:

The Sqoop user basically need to have the knowledge of the databases available in MYSQL along with the supported tables. While running the commands list-databases can be run normally, where as in list-tables we have to properly mention the data base name in the command itself otherwise Sqoop returns an error.

So before issuing the command the user need to verify the data bases in the MYSQL accordingly he has to use the corresponding DB then he can issue the command list-tables [7].

Sqoop import table allows the user so as to import the specified table into HDFS, to import a table into HDFS we can use the command sqoop import the results can be observed in the following figure.



```
[illegible]:~$ sqoop list-databases --connect jdbc:mysql://localhost --username root -P
Warning: $HADOOP_HOME is deprecated.

Enter password:
[illegible] INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'. The driver
is automatically registered via the SPI and manual loading of the driver class is generally unnecessary.
mysql
information_schema
performance_schema
sys
demo
[illegible]:~$ sqoop list-tables --connect jdbc:mysql://localhost/demo --username root -P
Warning: $HADOOP_HOME is deprecated.

Enter password:
[illegible] INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'. The driver
is automatically registered via the SPI and manual loading of the driver class is generally unnecessary.
countries
employees
[illegible]
jobs
[illegible]
[illegible]:~$

[illegible] INFO mapred.JobClient: Running job: job_[illegible]
[illegible] INFO mapred.JobClient:  map 0% reduce 0%
[illegible] INFO mapred.JobClient:  map 100% reduce 0%
[illegible] INFO mapred.JobClient: Job complete: job_[illegible]
[illegible] INFO mapred.JobClient: Counters: 18
[illegible] INFO mapred.JobClient:   Map-Reduce Framework
[illegible] INFO mapred.JobClient:     Spilled Records=0
[illegible] INFO mapred.JobClient:     Virtual memory (bytes) snapshot=[illegible]
[illegible] INFO mapred.JobClient:     Map input records=4
[illegible] INFO mapred.JobClient:     SPLIT_RAW_BYTES=[illegible]
[illegible] INFO mapred.JobClient:     Map output records=4
```

Figure 3: Sqoop Import working to import job table from MYSQL to HDFS.

Observation 4:

The common error we may expect in the processing of the sqoop import is file already existing error. That means while importing any table into HDFS the sqoop performs the map task and create a .jar file along with Job id and a target directory name. The target-directory in HDFS is generally created at the time of import process if the same table would have imported earlier then obviously the directory name in HDFS is same as that of Table in MYSQL which leads to the error. So the user need to verify the directory of HDFS manually then only he can issue the import command so as to avoid the error.

IV. CONCLUSION

The article described the Sqoop working and characteristics of the Sqoop along with the benefits of using the Sqoop between HDFS and MYSQL/Oracle. Sqoop architecture along with import/export process is explained in the work. The work provides the various common errors observed in the connection establishment. Initially username and password related aspects were addressed, there after the problem of connection with localhost/ifconfig is described as many times the users face the issue in the real time scenarios whileconnecting with remote data bases. The third observation is with the list-tables working and the user need to verify the corresponding data base so as to address the issue. The final observation is withimport command wherethere is a possibility of existing the target directory name in the HDFS in the name of the table of MySQL which we are importing to HDFS.

REFERENCES

1. www.apache.org
2. Mr. S. S. Aravinth An Efficient HADOOP Frameworks SQOOP and Ambari for Big Data Processing –International Journal for Innovative Research in Science & Technology| Volume 1 | Issue 10
3. Surajit Paul (surajit.paul@in.ibm.com), Advisory Consultant, IBM Sqoop: Big data conduit between NoSQL and RDBMS, IBM developer works.
4. Rohan Sidhu Implementation of Hadoop and Sqoop for Big Data Organization, e United States Department of Defense
5. “SqoopProjectLicense,” SqoopProjectLicense.[Online]. Available:
6. http://sqoop.apache.org/license.html.
7. T. Hall, "Apache Sqoop," in Sqoop, Hortonworks, 2015. [Online]. Available: http://hortonworks.com/apache/sqoop/.
8. A. Pavlo et al., "A Comparison of Approaches to Large-Scale Data Analysis," 2009. [Online].Available: http://db.csail.mit.edu/pubs/benchmarks-sigmod09.pdf.

AUTHORS PROFILE

Dr K Uma Pavan, Assoicate professor in CS and IT, Jain Univrsity, Bangalore,working in the areas of Machine Learning and Analytics with R and Python. Earned PhD from Pondicherry University. Hadoop and Big data certified.Currently working on ML and DL algorithms with different data sets

Dr S.V.N. Srinivasu, Professor in CSE, Narasaraopeta Engineering College (Autonomous), Narasaraopet, A.P..His area of interest includes Software Engineering, Software Testing, Machine Learning, Deep learning, Mobile Networking, Operating systems, Data Mining, Image Processing and Black chain technology. He Guided 10 Ph.D Scholars and published more than 80 articles in reputed journals which includes Scopus and other indexed bodies. He is a member of IEEE, MIET.

Dr. M N. Nachappa,Professor and Academic Head, School of CS and IT, Jain University,