

Big Data – cs522

Lab 4 – Day 12 (Saturday)

- Submit your *own work* on time. No credit will be given if the assignment is submitted after the due date.
 - Note that the completed lab should be submitted in .doc, .docx, .rtf or .pdf format only.
-

This document is divided into two parts.

1. Practice Labs

- a. [For Sqoop](#)
- b. [For Flume](#)

Just try to run through all the steps and see if they work properly for you.
No need to submit this part.

2. Homework Lab

- a. [For Sqoop](#)
- b. [For Flume](#)

You need to submit a simple document (not .zip) wherein I should be able to find all the instructions and commands for both the sqoop and flume HW.

Paste screenshots wherever applicable.

Sqoop Practice Lab

Part A - Importing from MySQL to HDFS

1. Setting up the table in MySQL

- 1) mysql --version
- 2) mysql -uroot -pcloudera
- 3) show databases;
- 4) create database cs522;
- 5) use cs522;
- 6) create table student (id int not null primary key auto_increment, name varchar(20), address varchar(20));
- 7) describe student;
- 8) insert into student values (1, "John", "12th Ave, Iowa"), (2, "Mary", "Boston"), (3, "Bob", "Des Moines"), (4, "Lina", "San Francisco");
- 9) select * from student;
- 10) quit;

2. Run the following sqoop import command.

```
sqoop import --connect jdbc:mysql://quickstart.cloudera/cs522 --username root --password cloudera --table student --target-dir=/user/cloudera/sqoopImportOutput
```

OR

```
sqoop import --connect jdbc:mysql://localhost/cs522 --username root --password cloudera --table student --target-dir=/user/cloudera/sqoopImportOutput -m 2
```

3. Verify the part-m files created in the output folder *sqoopImportOutput* from Cloudera HDFS browser.

Part B - Exporting from HDFS to MySQL

- 1. Delete the "student" data from MySQL DB which is created in Part A above. So now the student table is empty in the cs522 DB.**
- 2. Run the following command to export data from part-m files in HDFS to the MySQL table "student".**

```
sqoop export --export-dir=/user/cloudera/sqoopImport/ --connect jdbc:mysql://localhost/cs522 --username root -P --table student -m 1
```

Flume Practice Lab

1. Under `/home/cloudera/` create a folder named `"cs522"` and inside that create `/log/messages.txt` file which you'll use as a sample log file.
2. Inside folder `/home/cloudera/workspace/training/conf`, create `"myFlumeConf.conf"` file and add the following lines to it.

```
agent1.sources = mySource
agent1.channels = ch1
agent1.sinks = hdfsSink
```

```
agent1.sources.mySource.type = exec
agent1.sources.mySource.channels = ch1
agent1.sources.mySource.command = tail -F
/home/cloudera/cs522/log/messages.txt
```

```
agent1.sinks.hdfsSink.type = hdfs
```

```
agent1.sinks.hdfsSink.hdfs.path = hdfs://localhost/user/cloudera/flumeImport/
agent1.sinks.hdfsSink.hdfs.filePrefix = myFlume
agent1.sinks.hdfsSink.hdfs.fileType = DataStream
agent1.sinks.hdfsSink.hdfs.rollInterval = 3000
agent1.sinks.hdfsSink.hdfs.rollSize = 300
agent1.sinks.hdfsSink.hdfs.rollCount = 0
agent1.sinks.hdfsSink.channel = ch1
```

```
agent1.channels.ch1.type = memory
agent1.channels.ch1.capacity = 200
```

3. From the terminal, start the flume agent with the following command.

```
flume-ng agent -n agent1 -c /home/cloudera/workspace/training/conf/ -f
/home/cloudera/workspace/training/conf/myFlumeConf.conf
```

4. Check the files getting created and updated in HDFS at the path given in the conf file. (user/cloudera/flumeImport)

Sqoop Homework

1. Create a table named "stocks" in MySQL with the following sample schema and data.

id	symbol	quote_date	open_price	high_price	low_price
1	AAPL	2009-01-02	85.88	91.04	85.16
2	AAPL	2008-01-02	199.27	200.26	192.55
3	AAPL	2007-01-03	86.29	86.58	81.9

2. Import this table into HDFS.

NOTE: Only import columns "id", "symbol" and "open_price".

3. Write down all the commands to complete the above requirement.
 4. Paste a screenshot of the HDFS browser to show the imported files.
 5. Paste the contents of the imported file.
-

Flume Homework

Fan Out

It is important to understand this term before solving this HW.

Fan out is the term for delivering events from one source to multiple channels, so they reach multiple sinks. For example, the configuration shown in the following figure delivers events to both an HDFS sink (sink1a via channel1a) and a logger sink (sink1b via channel1b).

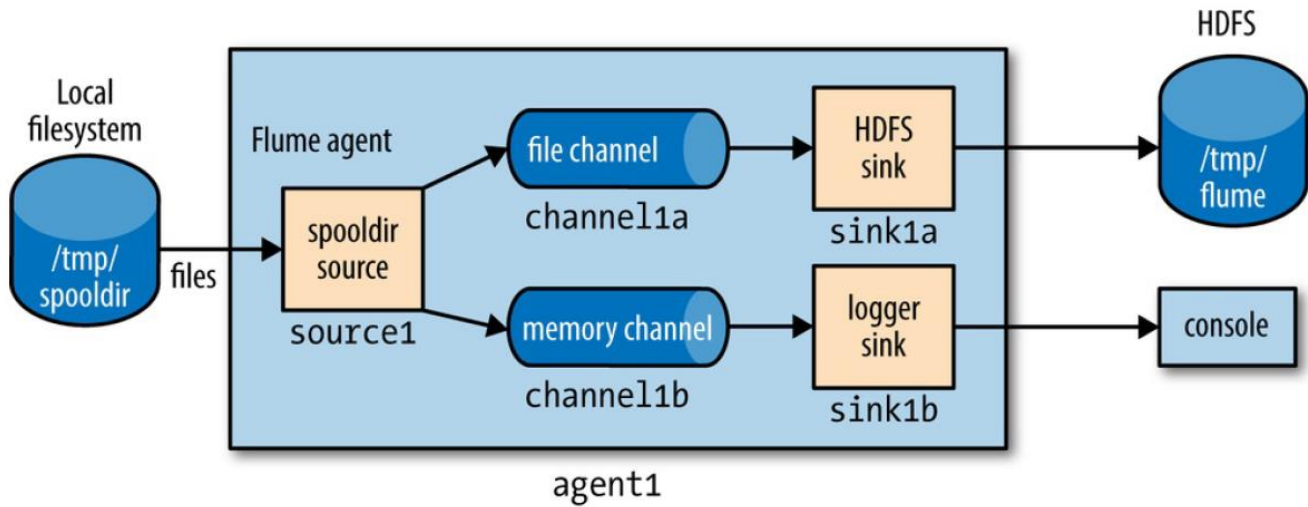


Figure 14-2. Flume agent with a spooling directory source and fanning out to an HDFS sink and a logger sink

Your task is to create a Flume configuration file using a spooling directory source, fanning out to an HDFS sink and a logger sink.
