Práctica - Sqoop and Flume

Ejercicios de Sqoop

1. Realizar el ejercicio 1 del Tutorial para importar datos desde una base de datos relacional a Hive (o Impala)

2. Utilizamos las tablas importadas en el ejercicio anterior. Queremos ver la categoria, producto y precio de los elementos de las categorías 58 y 59. Pero también queremos ver el precio mínimo y promedio de la categoría de cada registro:

```
select
product_category_id,
   product_name,
   product_price,
   min(product_price) OVER (PARTITION BY    product_category_id ORDER BY
product_price ),
   avg(product_price) OVER (PARTITION BY    product_category_id ORDER BY
product_price )
from
   products
where
   product_category_id  in (
      58, 59
   );
```

Ejercicios de Flume

1. Utilizar Flume para realizar un log de eventos. Para ello cree un archivo de configuración **flume.conf** con el siguiente contenido:

```
# Name the components on this agent
aNet2HDFS.sources = netcat-source
aNet2HDFS.channels = memory-channel
aNet2HDFS.sinks = hdfs

# Describe/configure Source
aNet2HDFS.sources.netcat-source.type = netcat
aNet2HDFS.sources.netcat-source.bind = localhost
aNet2HDFS.sources.netcat-source.port = 44445

# Describe the sink
aNet2HDFS.sinks.hdfs.type = hdfs
aNet2HDFS.sinks.hdfs.hdfs.path = hdfs:/results/2017-02/log_data/
aNet2HDFS.sinks.hdfs.hdfs.fileType = DataStream
aNet2HDFS.sinks.hdfs.hdfs.writeFormat = Text
aNet2HDFS.sinks.hdfs.hdfs.batchSize = 10
aNet2HDFS.sinks.hdfs.hdfs.rollSize = 0
aNet2HDFS.sinks.hdfs.hdfs.rollCount = 10
aNet2HDFS.sinks.hdfs.hdfs.filePrefix=data
aNet2HDFS.sinks.hdfs.hdfs.fileSuffix=log

# Use a channel which buffers events in memory
```

```
aNet2HDFS.channels.memory-channel.type = memory
aNet2HDFS.channels.memory-channel.capacity = 1000
aNet2HDFS.channels.memory-channel.transactionCapacity = 100

# Bind the source and sink to the channel
aNet2HDFS.sources.netcat-source.channels = memory-channel
aNet2HDFS.sinks.hdfs.channel = memory-channel
```

Luego desde una consulta lanze flume:

```
flume-ng agent --conf-file / flume.conf  --name aNet2HDFS
-Dflume.root.logger=INFO,console
```

Para poder enviar eventos a flume, desde otra consola ejecute

curl telnet://localhost:44445

Simula varios eventos escribiendolos y enter para enviar cada uno. Compruebe que se generaron los archivos en hdfs.

ctrl-c permite terminar la ejecución en ambas consolas

2. Realizar el ejercicio 4 del tutorial para importar data stream data de logs en una base solr.

3. [Opcional] Conectar a Flume para realizar una búsqueda en twitter es uno de los ejemplos más comunes. Dejo aquí la configuración el inconveniente es que hay que pedir unas credenciales OAuth para poder buscar. También a tener en cuenta es hacer una búsqueda sobre tópicos que no sean trending topics (ni muchos menos) para no abrumar a la máquina y mantener la conexión “polite”. Los pasos son adaptados del siguiente tutorial:
    a. Obtener las credenciales de twitter, como conseguir las creds mediante al camino real es engorroso. Se pueden obtener credenciales temporales en la consola de apigee, utilizando la autenticación OAUTH 1.
    b. Crear un archivo de configuración con el siguiente contenido

```
TwitterAgent.sources = Twitter
TwitterAgent.channels = MemChannel
TwitterAgent.sinks = HDFS

TwitterAgent.sources.Twitter.type = com.cloudera.flume.source.TwitterSource
TwitterAgent.sources.Twitter.channels = MemChannel
TwitterAgent.sources.Twitter.consumerKey =
TwitterAgent.sources.Twitter.consumerSecret =
```

```
TwitterAgent.sources.Twitter.accessToken =
TwitterAgent.sources.Twitter.accessTokenSecret =
TwitterAgent.sources.Twitter.keywords = hadoop, big data, analytics, bigdata

TwitterAgent.sinks.HDFS.channel = MemChannel
TwitterAgent.sinks.HDFS.type = hdfs
TwitterAgent.sinks.HDFS.hdfs.path = hdfs:/user/flume/tweets/%Y/%m/%d/%H/
TwitterAgent.sinks.HDFS.hdfs.fileType = DataStream
TwitterAgent.sinks.HDFS.hdfs.writeFormat = Text
TwitterAgent.sinks.HDFS.hdfs.batchSize = 1000
TwitterAgent.sinks.HDFS.hdfs.rollSize = 0
TwitterAgent.sinks.HDFS.hdfs.rollCount = 10000

TwitterAgent.channels.MemChannel.type = memory
TwitterAgent.channels.MemChannel.capacity = 10000
TwitterAgent.channels.MemChannel.transactionCapacity = 100
```

c. Descargar los siguiente archivos
   i. http://files.cloudera.com/samples/flume-sources-1.0-SNAPSHOT.jar
   ii. http://files.cloudera.com/samples/hive-serdes-1.0-SNAPSHOT.jar

d. copiar el primer archivo a la carpeta de lib de flume:

```
sudo cp $HOME/Downloads/flume-sources-1.0-SNAPSHOT.jar /usr/lib/flume-nglib/
```

e. lanzar la tarea de Flume

```
flume-ng agent --conf-file /home/cloudera/Desktop/twitter-2-hdfs.conf --name
TwitterAgent -Dflume.root.logger=INFO,console
-Dtwitter4j.streamBaseURL=https://stream.twitter.com/1.1/ --classpath
/usr/lib/flume-ng/lib/*
```

f. Comprobar que en HDFS se creen los archivos y ver que el contenido es un listado de tweets.