Twitter Workshop

Twitter Data Ingestion& Analysis

General Instructions:

Twitter, one of the largest social media websites. It receives millions of tweets (small messages) every day on variety of important issues. This huge amount of raw data can be used for industrial, Social, Economic, Government policies or business purpose by organizing according to our requirement and processing.

Hadoop is one of the best tool options for twitter data analysis as it works for distributed Big data, Streaming data, Time Stamped data, text data etc. This workshop introduces how to use FLUME and HIVE tools for twitter posts analysis. FLUME is used to extract real time twitter data into HDFS. Hive which is SQL like query language is used for some data extraction and analysis. To achieve this, we are going to follow the following steps:

1. Creating Twitter Application.
2. Configuring a Flume agent and getting data from Twitter.
3. Querying collected data using Hive Query Language (HQL)

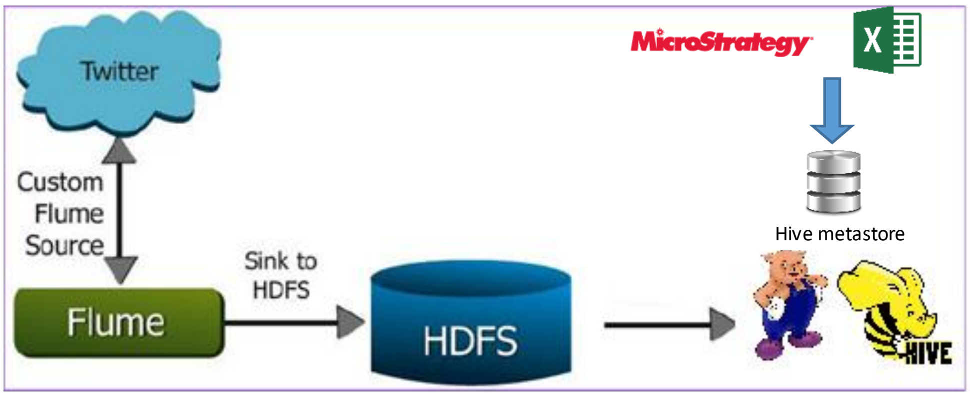
Online resources:
https://dev.twitter.com/overview/api/tweets
https://flume.apache.org/

Prerequisites:

To stream data to HDFS from Twitter you should have the following pre-requisites.

- Approved Twitter developer account
- Hadoop cluster running Flume and Hive (e.g: Cloudera CDH)

Start your Cloudera QuickStart VM to complete the workshop



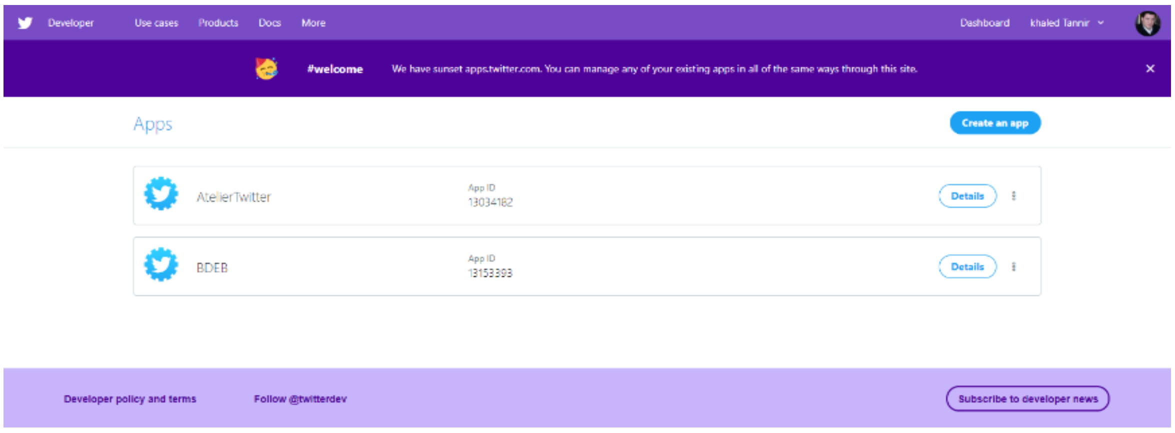
PART I : Collecting Twitter Data

First of all to get Twitter data you need to create a Twitter account and apply to a Twitter developer account. Once it is done you will be able to create your Twitter application. For more information go to: https://apps.twitter.com/

Step 1: Creating Twitter Application

Creating a Twitter application can be done directly from the Twitter Developer website https://developer.twitter.com/

- From the Twitter developer web page, click on the '`Create an App`' button



We have sunset apps.twitter.com. You can manage any of your existing apps in all of the same ways through this site.

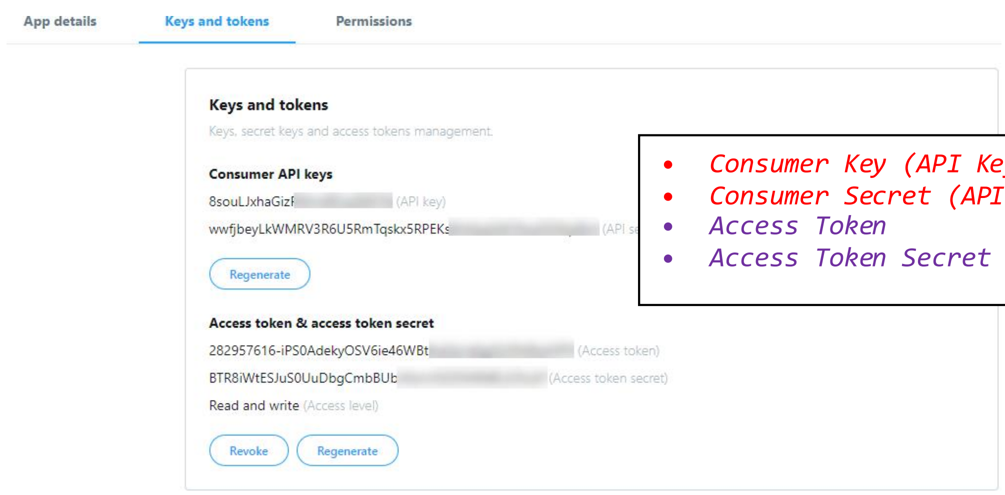
Apps

AtelierTwitter	App ID 13034182	Details
BDEB	App ID 13153393	Details

After creating a new application you need to get/create the **access tokens** so that you don't need to provide your authentication details. Also after creating application you need to get the **consumer keys** to access that application for getting Twitter data.

The following figure shows clearly the application keys that are generated after creating application and in this keys we can see the top two keys are the API key and API secret. Coming to the reaming two keys it is nothing but known as the Access Tokens that you need to generate it by yourselves.

To generate access token, just click on the dedicated button. After clicking you will get the two keys `Access token` and **`Access token`** secret. You need to take these keys **and `token details`** and set in the Flume configuration file. Such that you can get the required data from the Twitter in the form of tweets.



Keys and tokens

Keys, secret keys and access tokens management.

Consumer API keys

8souLJxhaGizF[redacted] (API key)

wwfjbeyLkWMRV3R6U5RmTqskx5RPEKs[redacted] (API se

Access token & access token secret

282957616-iPS0AdekyOSV6ie46WBt[redacted] (Access token)

BTR8iWtESJuS0UuDbgCmbBUb[redacted] (Access token secret)

Read and write (Access level)

- *Consumer Key (API Key)*
- *Consumer Secret (API Secret)*
- *Access Token*
- *Access Token Secret*

From this page you can generate / retrieve the 4 keys you will need. These keys are generated by Twitter for you. They are needed to allow Flume to access your Twitter App and start capturing data.

Step 2: Configuring the Flume Agent

This step consists of configuring the Flume agent to be able to start collecting data from Twitter.

1. Create a new Flume agent configuration
2. Complete the file with these configuration settings

Flume agent configuration file		
Agent name: TwitterAgent		
Sources	name	Twitter
	type	com.cloudera.flume.source.TwitterSource
	consumerKey	**** your key ****
	consumerSecret	**** your key ****
	accessToken	**** your key ****
	accessTokenSecret	**** your key ****
	keywords	*Hadoop, Hive, HBase, Big Data, Data Science, AI, Spark*
	channels	MemChannel
	Interceptors	ts
	interceptors.ts.type	timestamp
Channels	name	MemChannel
	type	memory
	capacity	1000000
	transactionCapacity	500000
Sinks	name	HDFS
	type	hdfs
	hdfs.path	/tweets/year=%Y/month=%m/day=%d/hour=%H/
	hdfs.writeFormat	Text
	hdfs.fileType	DataStream
	hdfs.batchSize	1000
	hdfs.rollSize	0
	hdfs.rollCount	10000

Step 3: Running the Flume Agent

Prerequisites:

Before running the Flume agent you need to place the provided archive `flume-sources-1.0-SNAPSHOT.jar` into `/usr/lib/flume-ng/lib`

Note: We assume that the archive is located into `/home/cloudera/Downloads/`

1. Use the following command to copy the archive into the target directory

```
$ sudo cp /home/cloudera/Downloads/flume-sources-
  1.0-SNAPSHOT.jar  /usr/lib/flume-ng/lib
```

2. Run the Flume agent

```
$ flume-ng agent -n TwitterAgent -c conf -f
  flumeTwitter.conf
```

Wait a few seconds and check the HDFS directory `/tweets,` it should contains tweets in JSON format collected by the Flume agent.

PART II : Twitter data analysis

In this section, the objective is to analyze the Twitter data collected by the Flume agent. S an example we will use Hive to :

- Query the collected tweets
- Calculate the number of #hashtags.
- To retrieve the three users having most followers
- Get the 10 most popular languages
- Identify the 10 most influential people.

To identify an influential person, we calculate the maximum number of tweets that have been re-tweeted.

Section 1: Reading the tweets using Hive

The data captured by the Flume agent was stored as a Hive partition on HDFS in JSON format. To allow Hive to read this format it is important to:

A. Use an appropriate JSON **SerDe** (Serializer / Deserializer)

To read and write data in JSON format. We could use any Hive JSON Serde. There is one packaged with Hive/HCatalog and available in the Cloudera distribution.

B. Make a schema projection respecting the JSON schema of a Tweet

For more details on the Twitter JSON Objects : https://developer.twitter.com/

Creating the Hive table:

1. From Hive create a new database **twitter** and make it the current database
2. Use the **ADD JAR** command to register the JSON **SerDe** into **Hive** jars list

```
ADD JAR /usr/lib/hive-hcatalog/share/hcatalog/hive-
hcatalog-core-1.1.0-cdh5.13.0.jar;
```

3. Check the JSON SerDe in Hive jars list

```
list jars;
```

4. Create a new External table '**tweets**' based on the following schema:

```
CREATE EXTERNAL TABLE twitter.tweets (
  id string, created_at string, text string,
  source string,

  user STRUCT<
    id_str: string,  name: string, screen_name: string,
    location: string,  description: string,
    followers_count: bigint,  statuses_count: int,
    friends_count: bigint,  verified: BOOLEAN,
    utc_offset: int, time_zone: string,
    geo_enabled: BOOLEAN >,

  favorited BOOLEAN, lang string,
  in_reply_to_status_id string,
  in_reply_to_user_id string,
  in_reply_to_screen_name string,
  geo  string, coordinates string, place string,
  contributors string,

  retweeted_status STRUCT<
    text: string,  retweet_count: INT,
     user:STRUCT<screen_name: string, name: string>,

     in_reply_to_status_id: string,
    in_reply_to_user_id: string,
    in_reply_to_screen_name: string >,

  entities STRUCT<
    hashtags: ARRAY<STRUCT<text: string >>,
    urls: ARRAY< STRUCT<url: string >>,
    user_mentions: ARRAY< STRUCT<screen_name: string,
     name: string, id_str: string >>>,
  timestamp_ms  string)
PARTITIONED BY (year INT, month INT, day INT, hour INT)
ROW FORMAT SERDE
'org.apache.hive.hcatalog.data.JsonSerDe'
LOCATION '/tweets';
```

Data collected by the Flume agent was stored on HDFS as a Hive partitions. To be able to read this data using Hive you need to add these partition(s) to the tweets table.

5. Add a Hive partition to the tweets table Hive to read collected data

```
MSCK REPAIR TABLE tweets
```

Tweets Analysis:

1. Display the first 20 tweets

```
select * from tweets limit 20;
```

2. Write a HiveQL to retrieve the three users having most followers
3. What are 10 most popular languages?
4. Identify the 10 most influential people.

PART III: Twitter data visualization

In this section, we will use a visualization software to connect to the Hive metastore and create a graph based on the data stored on HDFS.
For instance we will use MicroStrategy Desktop but you can use any other visualization software such as Tableau Software, Altryx, etc…

Step 1: Data preparation

We will create a new Hive view having the same structure and containing a subset of the tweets table rows. The main reason for having a view of a table is to simplify some of the complexities of a larger table into a more flat structure.
Unlike some RDBMS, once the Hive view is created, its schema is frozen immediately. Subsequent changes in the underlying table will not be reflected in the View; and if the table is dropped, the view will fail.

1. Create a Hive view tweets_vis table having the same structure as the tweets tabe.

2. Check the view and display the first 10 rows from the view

Step 2: Data visualization

This step is to configure MicroStrategy Desktop to access the Hive server and extract the data from the tweets_vis view. From MicroStrategy software you will:

- Add a new data source
- Write a HiveQL query to extract the data from the view
- Create a new bar graph

1. Open MicroStrategy
2. Create a new dashboard
3. Add a new data connection -> **Hadoop** -> **Cloudera Hive**
4. From the window '**Select importation options**' -> **Create a new query**
5. Configure the data source

a.	Version	`Hive 0.14.x`
b.	Host	*`*** Cloudera VM IP address ***`*
c.	Port	`10000`
d.	Database	`default`
e.	User	`cloudera`
f.	Password	`cloudera`

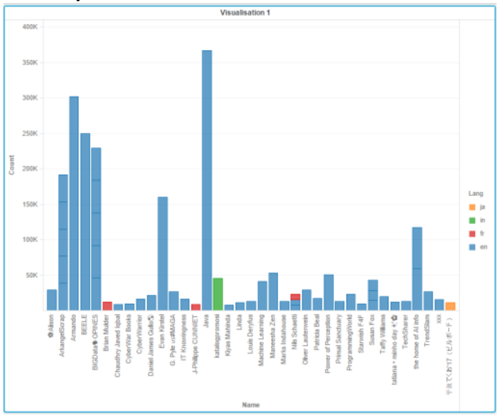
6. From the **`'Import from tables'`** window, add the following HiveQL query

```
select user.name as name, user.followers_count as
count, lang from  tweets_vis order by count desc
limit 50
```

(*this query will return the top 50 users having the most followers*)

7. From **Type Query -> Modify** the query dataset. Add the **Count** column as **Measure**

8. Select Bars Chart
   a. Field **Name** -> Horizontal
   b. Field **Count** -> Vertical
   c. Field **Lang** -> By Colour



PART IV: Twitter data export

In this section you will use Sqoop to export the result of a Hive query into MySQL table Tweets_followers.

1. From Hive create a new table tweets_followers and use the query from the previous section to populate it.
2. Open a new MySQL session
3. Create a new database Export and make it as the current
4. Create a new table **tweets_followers_export**

```
CREATE TABLE Tweets_followers_export (
    name varchar(255),
    count int(11),
    lang varchar(5)
);
```

5. Use Sqoop to export **tweets_followers** table to **tweets_followers_export** MySQL table

6. Check data has been imported into tweets_followers_export table