we want to load the data which is of type semi-structured and unstructured into the  HDFS cluster, or else capture the live streaming data which is generated, from different sources like twitter, weblogs and more into the HDFS cluster, which component of Hadoop ecosystem will be useful to do this kind of job. The solution is **FLUME**.

Learning Flume will help users to collect from and store a large amount of data from different sources into the Hadoop cluster.

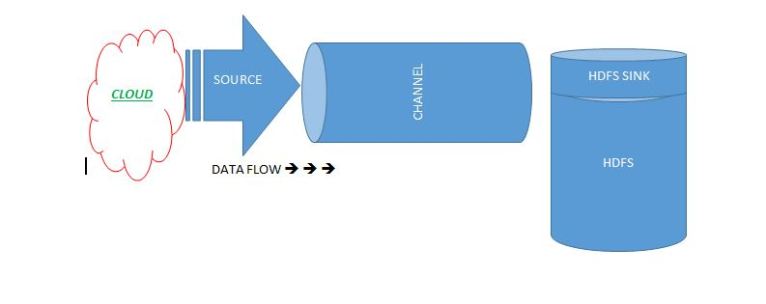
**What is Apache Flume?**

Apache Flume is a Hadoop ecosystem component used to collect, aggregate and moves a large amount of log data from different sources to a centralized data store.

It is an open source component which is designed to locate and store the data in a distributed environment and collects the data as per the specified input key(s).

**Flume Architecture**

Before moving forward to know the working of flume tool, It is mandatory to know the Flume architecture first.



Flume is composed of the following components.

**Flume Event:**It is the main unit of the data that is transported inside the **Flume**(Typically a single log entry). It contains a payload of the byte array that is to be transported from the source path to the destination path which could be accompanied by optional headers.

A Flume event will be in the following structure.

|  |  |
| --- | --- |
| Header | Byte Payload |

**Flume Agent:** Is an independent Java virtual machine daemon process which receives the data (events) from clients and transports to the subsequent destination (sink or agent).

**Source:**Is the component of Flume agent which receives data from the data generators say, twitter, facebook, weblogs from different sites and transfers this data to one or more channels in the form of Flume event.

The external source sends data to Flume in a format that is recognized by the target Flume source. Example, an Avro Flume source can be used to receive Avro data from Avro clients or other Flume agents in the flow that send data from an Avro sink, or the Thrift Flume source will receive data from a Thrift sink, or a Flume Thrift RPC client or Thrift Clients are written in any language generated from the Flume thrift protocol.

**Channel:**Once, the Flume source receives an Event, it stores this data into one or more channel and buffers them till they are consumed by sinks. It acts as a bridge between the source and sinks. These channels are implemented to handle any number of sources and sinks.  
**Sink:**It stores the data into the centralized stores like HDFS and HBase.

**Streaming Twitter Data**

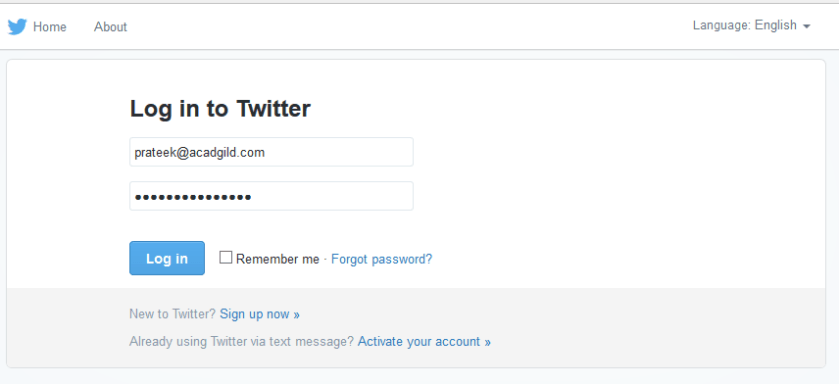
To stream data to our database from twitter we should have the following pre-requisites.

* Twitter account
* Hadoop cluster

If both prerequisites are available we can move to our further step.

**Step 1:**

Login to  the twitter account



**Step 2:**

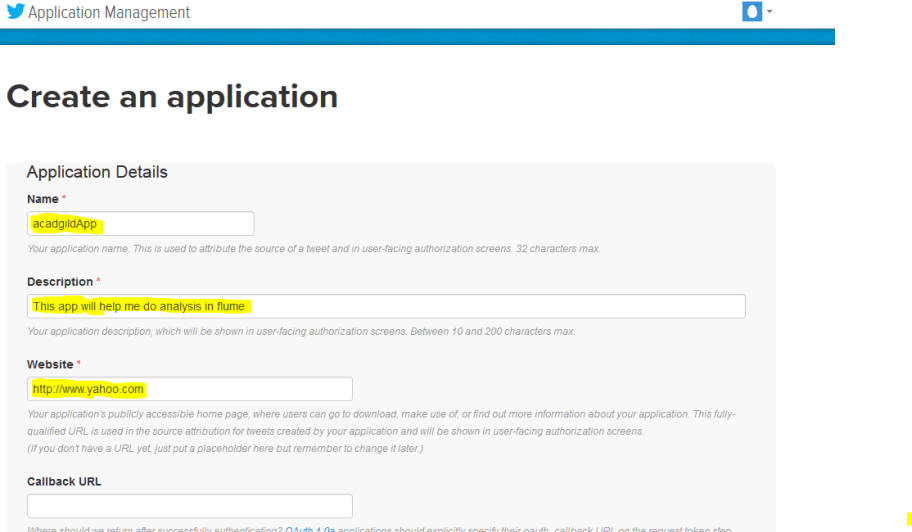
Go to the following link and click the  ‘create new app’ button.

[**https://apps.twitter.com/app**](https://apps.twitter.com/app)



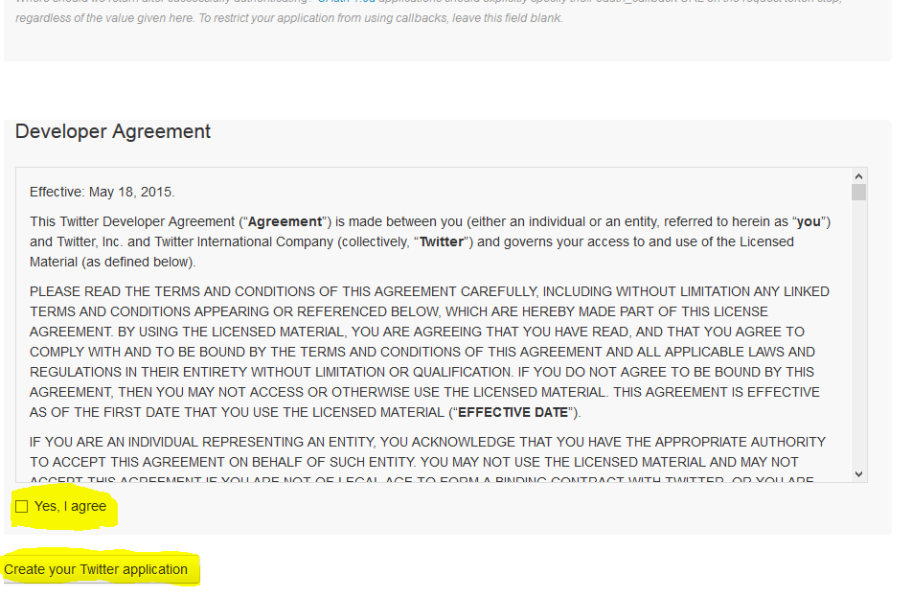
**Step 3:**

Enter the necessary details.



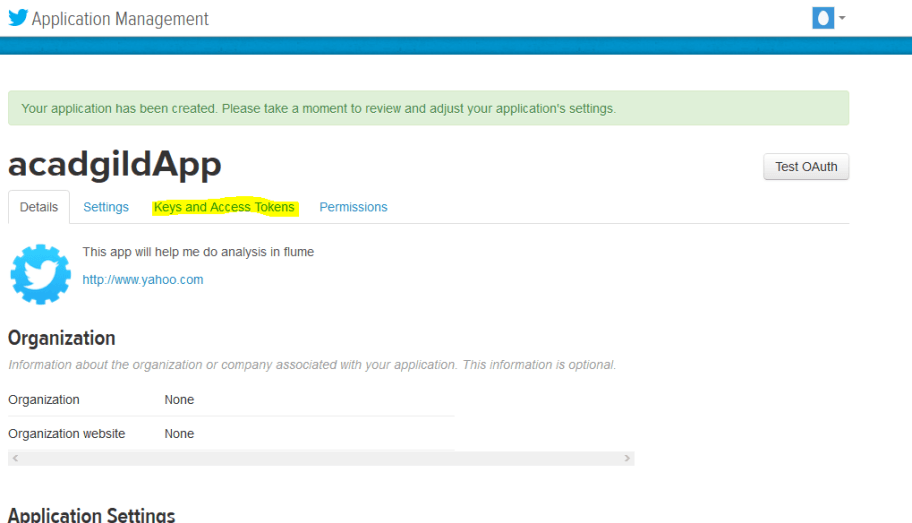
**Step 4:**

Accept the developer agreement and select the ‘create your Twitter application’ button.



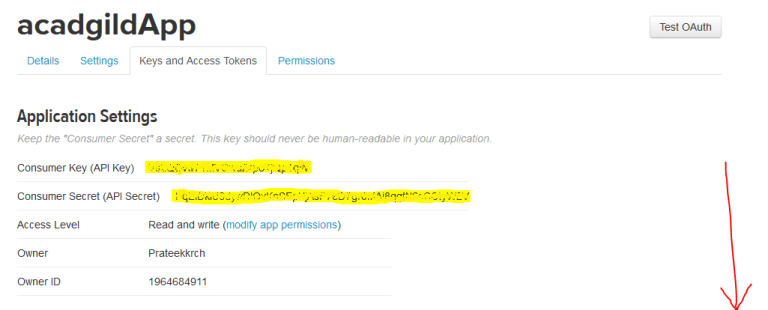
**Step 5:**

Select the ‘Keys and Access Token’ tab.



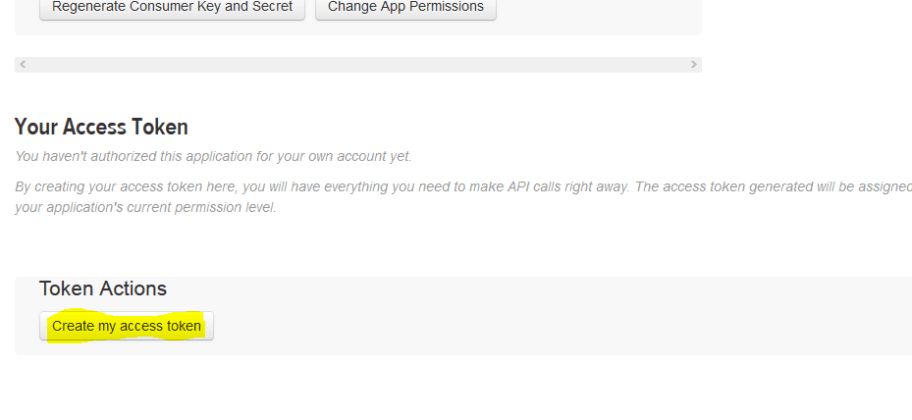
**Step 6:**

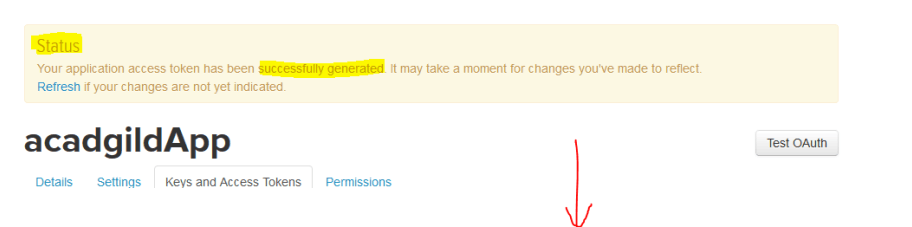
Copy the consumer key and the consumer secret code.



**Step 7:**

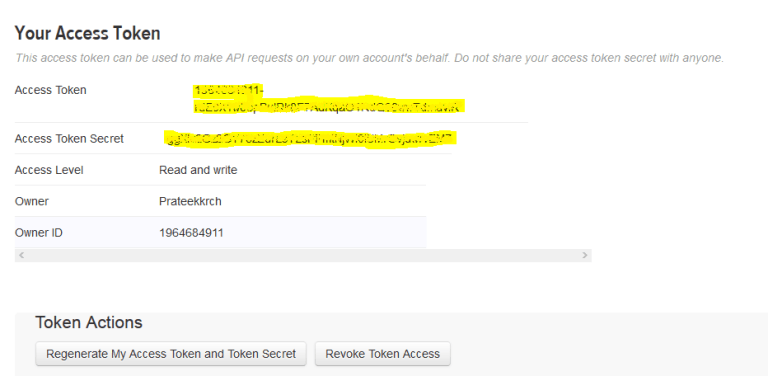
Scroll down further and select the ‘create my access token’ button.

  
Now, you will receive a message stating “that you have successfully generated your application access token”.



**Step 8:**

Copy the Access Token and Access token Secret code.

****

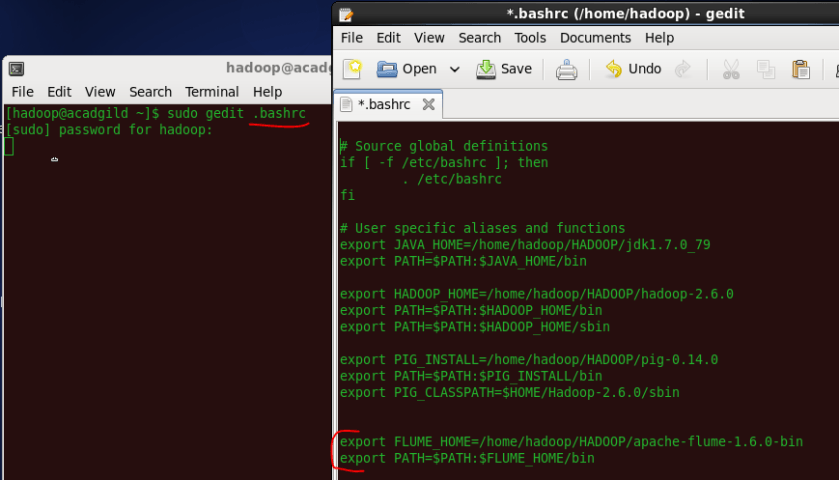
Follow Step 9 and Step 10 to install Apache flume

**Step 9:**Download flume tar file from below link and extract it.

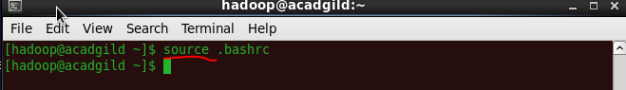
[**https://drive.google.com/drive/u/0/folders/0B1QaXx7tpw3SWkMwVFBkc3djNFk**](https://drive.google.com/drive/u/0/folders/0B1QaXx7tpw3SWkMwVFBkc3djNFk)

Right click on the downloaded flume tar file and select the option as Extract Here to untar the flume directory and update the path of extracted flume directory in the .bashrc file as mentioned in the below image.

**NOTE: keep the path same as where the extracted file exists.**

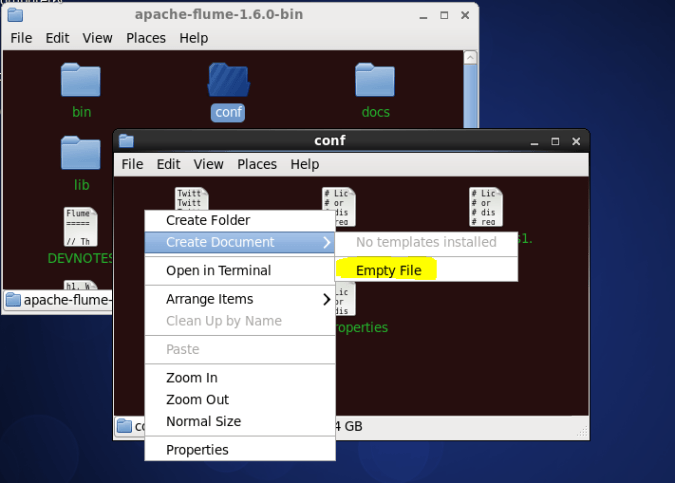


After setting the path of flume directory, save and close the .bashrc file.  And then in the terminal type the below command to update the .bashrc file.



**Step 10:**

Create a new file inside the conf directory inside the Flume-extracted directory.



Note: Make sure you have below jars placed in your $FLUME\_HOME/lib directory:

1. twitter4j-core-X.XX.jar
2. twitter4j-stream-X.X.X.jar
3. twitter4j-media-support-X.X.X.jar

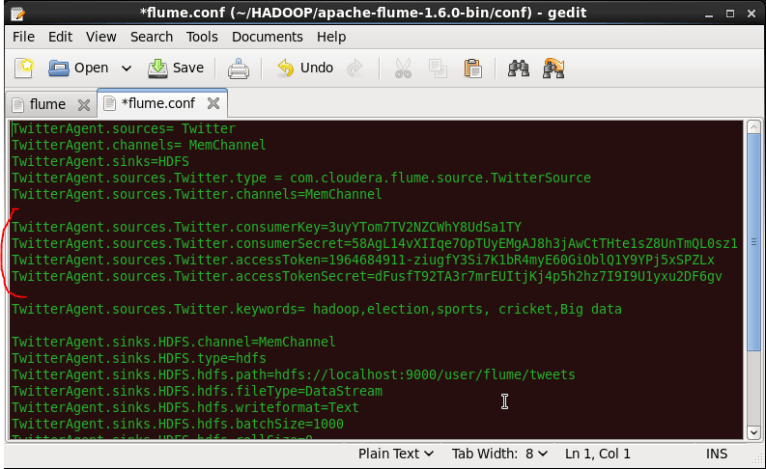
**Step 11:**

Copy theFlumee configuration code from the below link and paste it in the newly created file.

<https://drive.google.com/open?id=0B1QaXx7tpw3Sb3U4LW9SWlNidkk>

**Step 12:**

Change the twitter api keys with the keys generated as shown in the step no 6 and step number 8.



**Step 13:**

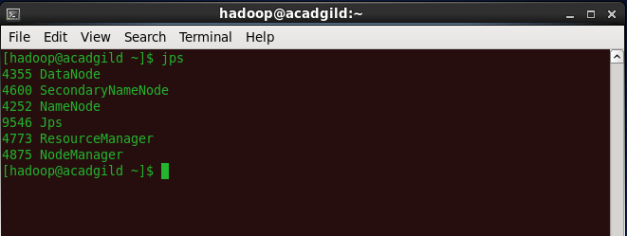
We have to decide which keywords tweet data to be collected from the twitter application. So, you can change the keywords in the TwitterAgent.sources.Twitter.keywords command.

In our example, we are fetching tweet data related to Hadoop, election, sports, cricket and Big data.

**Step 14:**

Open a new terminal and start all the Hadoop daemons, before running the flume command to fetch the twitter data.

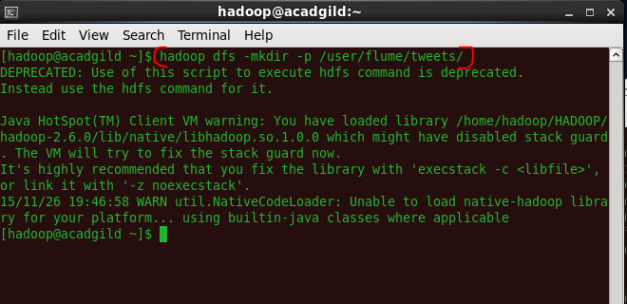
Use the ‘jps’ command to see the running Hadoop daemons.

****

**Step 15:**

Create a new directory inside HDFS path, where the Twitter tweet data should be stored.

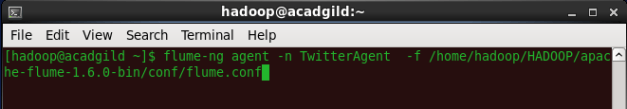
**Hadoop dfs –mkdir –p /user/flume/tweets**

****

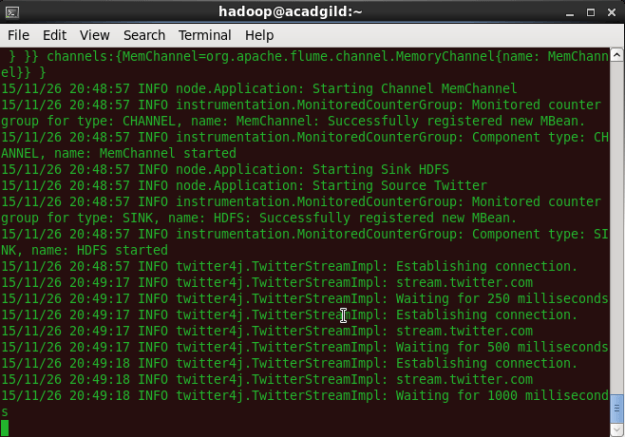
**Step 16:**

For fetching data from Twitter, Use the below command to fetch the twitter tweet data into the HDFS cluster path.

**flume-ng agent -n TwitterAgent -f <location of created/edited conf file>**



The above command will start fetching data from Twitter and steams it into the HDFS given path.

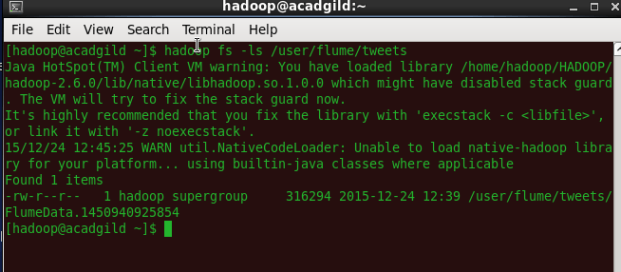


Once, the tweet data started streaming it into the given HDFS path we can use ‘Ctrl+c’ command to stop the streaming process.

**Step 17:**

To check the contents of the tweet data we can use the following command:

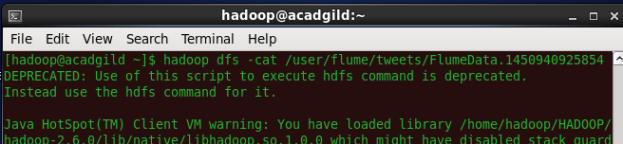
**hadoop dfs –ls /user/flume/tweets**

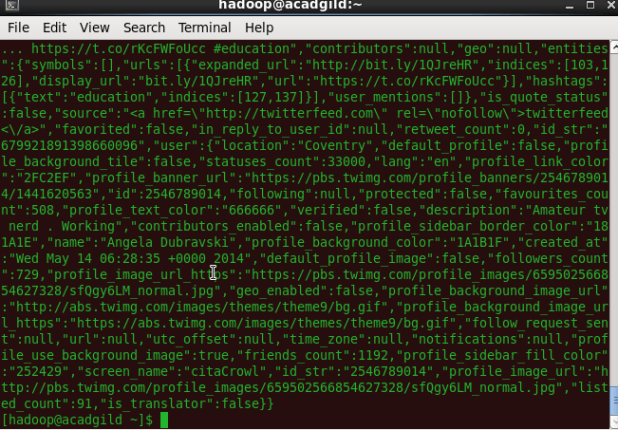


**Step 18:**

We can use the *‘cat’* command to display the tweet  data inside the /user/flume/tweets/FlumeData.145\* path.

**hadoop dfs –cat /us er/flume/tweets/<flumeData file name>**





We can observe from the above image that we have successfully fetched twitter data into our HDFS cluster directory.  Once the tweets have been successfully stored in your database, you can manipulate the tweet data to fit the needs of our future projects. You can follow the above steps for the same.