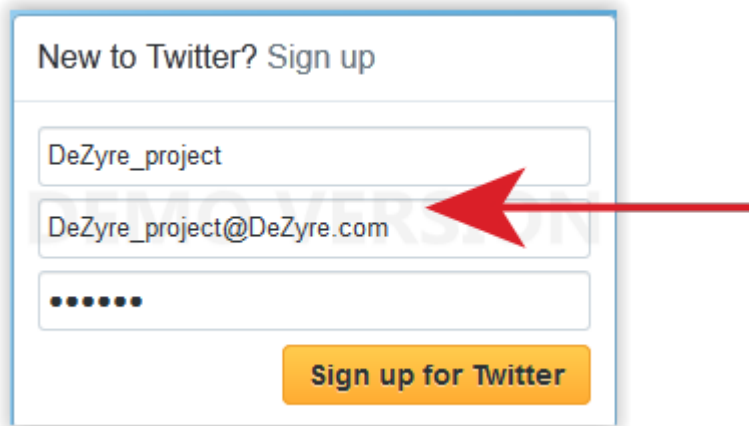


## Analyzing Twitter Data with Apache Hadoop

### Create twitter access token:

1. Create twitter account: Go to twitter.com and create a twitter account as shown below.



New to Twitter? Sign up

DeZyre\_project

DeZyre\_project@DeZyre.com

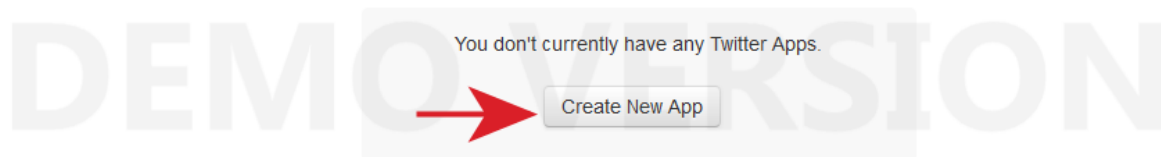
.....

Sign up for Twitter

A red arrow points to the email input field.

2. Verify email for your twitter account.
3. Go to apps.twitter.com and click on "Create App" as shown below in screenshots.

### Twitter Apps



## Create an application

### Application details

**Name \***

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

**Description \***

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

**Website \***

Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens. (If you don't have a URL yet, just put a placeholder here but remember to change it later.)

**Callback URL**

4. Once you have created an application, click on "API Keys" section.

## DeZyre\_Project

Details

Settings

API Keys

Permissions



DeZyre Test Project

<http://www.dezyre.com/Big-Data-and-Hadoop/19>

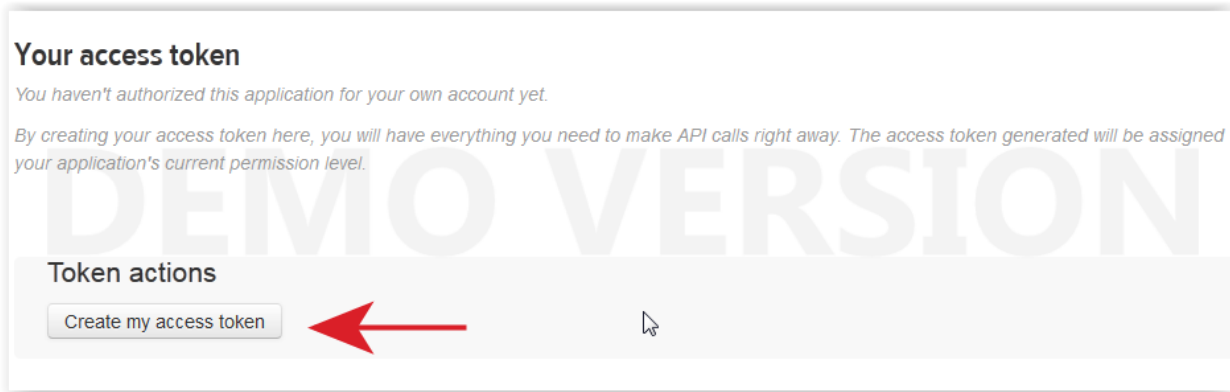
### Organization

Information about the organization or company associated with your application. This information is optional.

Organization None

Organization website None

5. Click on “Create My Access Token” button in the “Your access token” section.



6. Your access token will be generated in some time which you would need to use in flume.conf file for flume configuration in next few steps.

### Application settings

*Keep the "API secret" a secret. This key should never be human-readable in your application.*

API key	Yjs2GJWwy4UfDpHabxcUF8lyh
API secret	3Jq0xRKgBkXiQB0ULxcpMNgqOceYGFj5nwICrRtmpAI2EicNuW
Access level	Read-only ( <a href="#">modify app permissions</a> )
Owner	mankum1911
Owner ID	2516488472

### Application actions

Regenerate API keys
Change App Permissions

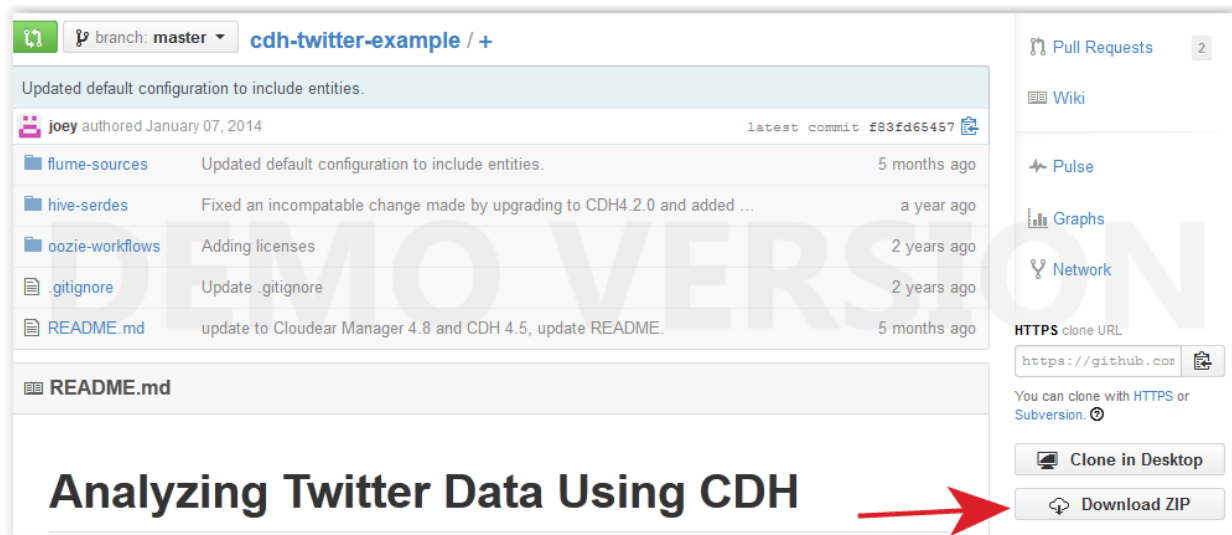
### Your access token

*This access token can be used to make API requests on your own account's behalf. Do not share your access token secret with anyone.*

Access token	2516488472- CVAyV5q1u1lyXgKok0G6iVJxhcqFRGmD1cgcPSC
Access token secret	9Df7vDAsSaeti5vonjaCbZ6Lmq4ozIDdbNsUgUYhNONU0
Access level	Read-only
Owner	mankum1911
Owner ID	2516488472

## Setting up flume agent:

1. Go to this link and download code from [here](https://github.com/cloudera/cdh-twitter-example) (<https://github.com/cloudera/cdh-twitter-example>) **OR** else you can use the jar which is shared under twitter-project.



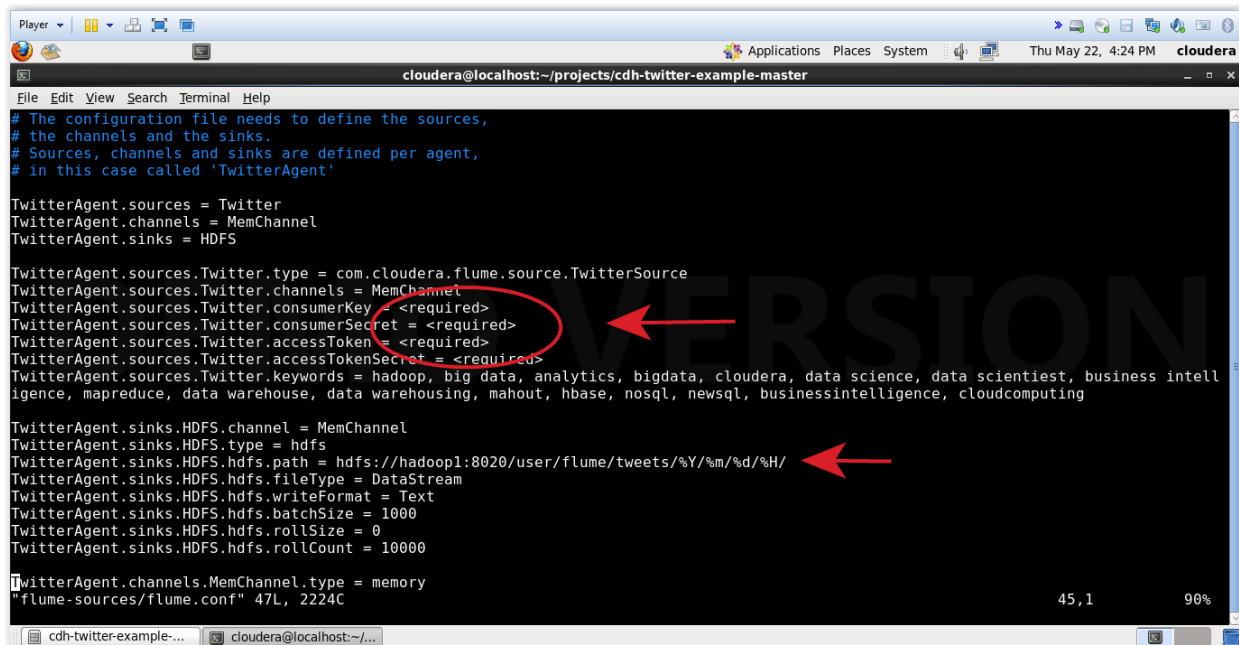
2. Extract the jar and run following command for building flume source and copy to flume library. ***Its not needed if you are using flume-sources-jar shared in twitter-project folder.***

```
$ cd flume-sources/
$ mvn clean install -DskipTests
$ sudo cp /flume-sources-1.0-SNAPSHOT.jar /usr/lib/flume-ng/lib/
$ sudo chmod +r /usr/lib/flume-ng/lib/flume-sources-1.0-SNAPSHOT.jar
```

3. Create a HDFS directory “/user/cloudera/twitter/” where you will get the twitter inputs.

```
$ hadoop dfs -mkdir /user/cloudera/twitter/
```

4. Edit flume conf “flume-sources/flume.conf” to change twitter keys and tokens and hdfs path. Use twitter access tokens from twitter account. Change HDFS path to “/user/cloudera/twitter/%Y/%m/%d/%H/”



```

# The configuration file needs to define the sources,
# the channels and the sinks.
# Sources, channels and sinks are defined per agent,
# in this case called 'TwitterAgent'

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 = <required>
TwitterAgent.sources.Twitter.consumerSecret = <required>
TwitterAgent.sources.Twitter.accessToken = <required>
TwitterAgent.sources.Twitter.accessTokenSecret = <required>
TwitterAgent.sources.Twitter.keywords = hadoop, big data, analytics, bigdata, cloudera, data science, data scientiest, business intell
igence, mapreduce, data warehouse, data warehousing, mahout, hbase, nosql, newsql, businessintelligence, cloudcomputing

TwitterAgent.sinks.HDFS.channel = MemChannel
TwitterAgent.sinks.HDFS.type = hdfs
TwitterAgent.sinks.HDFS.hdfs.path = hdfs://hadoop1:8020/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
"flume-sources/flume.conf" 47L, 2224C
  
```

5. Now start flume-agent using below command.

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

6. In some time file will start coming in HDFS. Check using below command.

```
$ hadoop dfs -ls /user/cloudera/twitter
```

## Hive Setup

1. Run following command for building SerDe library **OR** else you can use the jar which is shared under twitter-project.

```
$ cd hive-serdes/  
$ mvn clean install -DskipTests
```

2. Start hive using hive command.
3. Now add jar using add jar command as shown below.

```
Hive> add jar hive-serdes-1.0-SNAPSHOT.jar;
```

4. Create a external table in hive as shown below.

```
Hive> CREATE EXTERNAL TABLE tweets_partioned (  
  id BIGINT,  
  created_at STRING,  
  source STRING,  
  favorited BOOLEAN,  
  retweeted_status STRUCT<  
    text:STRING,  
    user:STRUCT<screen_name:STRING,name:STRING>,  
    retweet_count:INT>,  
  entities STRUCT<  
    urls:ARRAY<STRUCT<expanded_url:STRING>>,  
    user_mentions:ARRAY<STRUCT<screen_name:STRING,name:STRING>>,  
    hashtags:ARRAY<STRUCT<text:STRING>>>,  
  text STRING,  
  user STRUCT<  
    screen_name:STRING,  
    name:STRING,  
    friends_count:INT,  
    followers_count:INT,  
    statuses_count:INT,  
    verified:BOOLEAN,  
    utc_offset:INT,  
    time_zone:STRING>,  
  in_reply_to_screen_name STRING  
)  
PARTITIONED BY (year INT, month INT, dt INT, dthour INT)  
ROW FORMAT SERDE 'com.cloudera.hive.serde.JSONSerDe'  
LOCATION '/user/cloudera/tweets';
```



## Oozie Workflow Setup:

Oozie workflow here is used to create partition based on directories.

1. External JARs are provided to Oozie through a lib directory in the workflow directory. The workflow will need a copy hive-serdes JAR. Run following command :

```
$ mkdir -p oozie-workflows/lib  
$ cp hive-serdes-1.0-SNAPSHOT.jar twitter-project/oozie-workflows/lib/  
$ cp /var/lib/oozie/mysql-connector-java.jar oozie-workflows/lib
```

2. Copy hive-site.xml to the oozie-workflows directory

```
$ cp /etc/hive/conf/hive-site.xml oozie-workflows/  
$ sudo chown oozie:oozie oozie-workflows/hive-site.xml  
$ hadoop dfs -copyFromLocal oozie-workflows /user/cloudera/  
$ oozie job -oozie http://localhost:11000/oozie -config oozie-workflows/job.properties -run
```

## If you are not able to do oozie setup

Using Alter table commands similar to below one add partition to the table.

```
alter table tweets_partitioned add partition (year=2014, month=05, dt=24, dthour=05);  
alter table tweets_partitioned add partition (year=2014, month=05, dt=24, dthour=08);
```

*Check whether has been successfully added and it queryable or not.*

```
hive> select * from tweets_partitioned where year=2014 and month=05 and dt=24 and  
dthour=05 limit 10;
```

To find out influential celebrity or people

Command:

```
select t.retweeted_screen_name, sum(retweets) as total_retweets, count(*) as tweet_count
from
(
select retweeted_status.user.screen_name as retweeted_screen_name,
retweeted_status.text, max(retweeted_status.retweet_count) as retweets from
tweets_partioned group by retweeted_status.user.screen_name,retweeted_status.text
)
t
group by t.retweeted_screen_name
order by total_retweets DESC, tweet_count ASC
limit 10;
```