Analyse Tweets using Flume, Hadoop and Hive

April 2015

Dr.Thanachart Numnonda
Certified Java Programmer
thanachart@imcinstitute.com

Danairat T.
Certified Java Programmer, TOGAF – Silver
danairat@gmail.com

Lecture: Understanding Flume

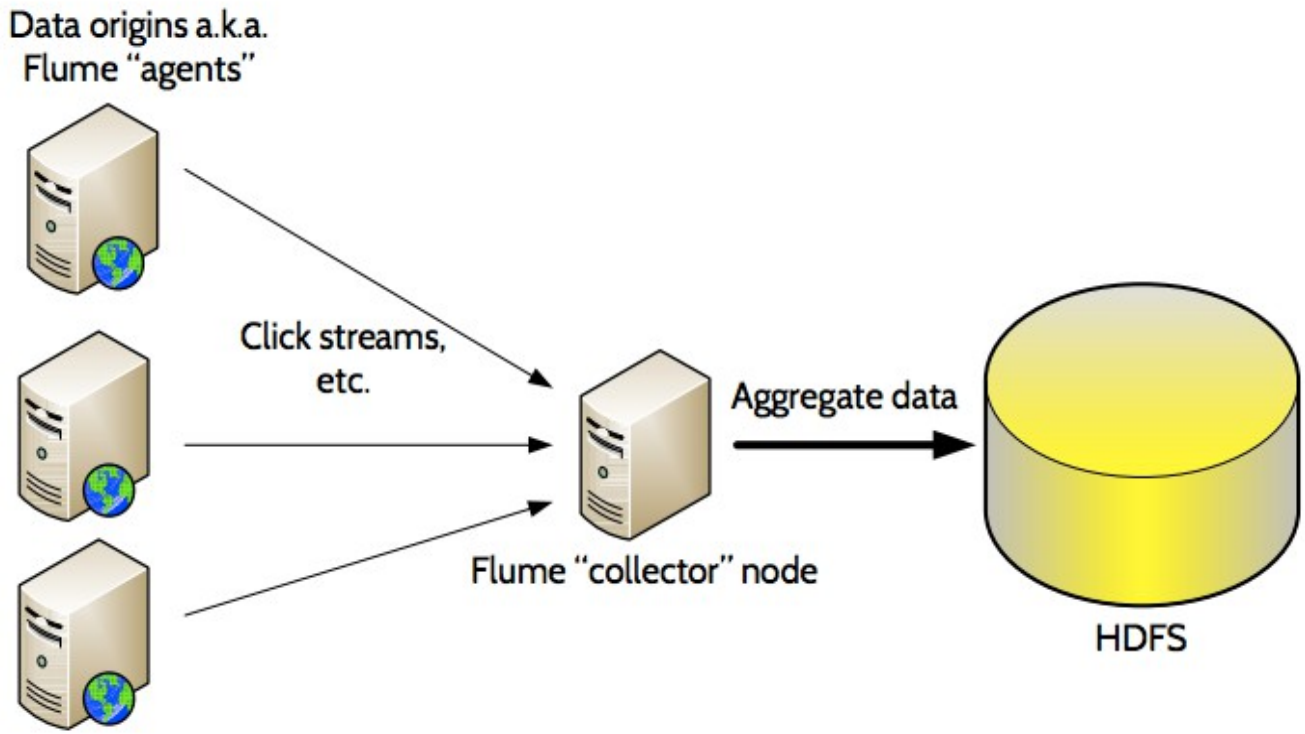
Danairat T., , danairat@gmail.com: Thanachart Numnonda, thanachart@imcinstitute.com

Introduction

Apache Flume is:

- **A distributed data transport and aggregation system for event- or log-structured data**
- **Principally designed for continuous data ingestion into Hadoop… But more flexible than that**

Architecture Overview



odiago

Flume terminology

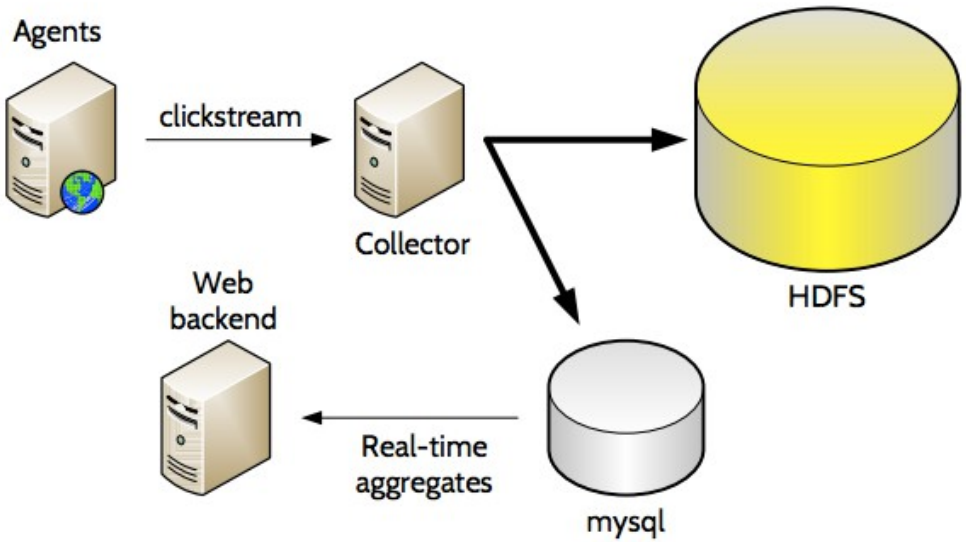
- Every machine in Flume is a node
- Each node has a source and a sink
- Some sinks send data to collector nodes, which aggregate data from many agents before writing to HDFS
- All Flume nodes heartbeat to/receive config from master
- Events enter Flume within seconds of generation

Odiago

Danairat T., , danairat@gmail.com: Thanachart Numnonda, thanachart@imcinstitute.com Apr 2015

Flume isn't an analytic system

- No ability to inspect message bodies
- No notion of aggregates, rolling counters, etc

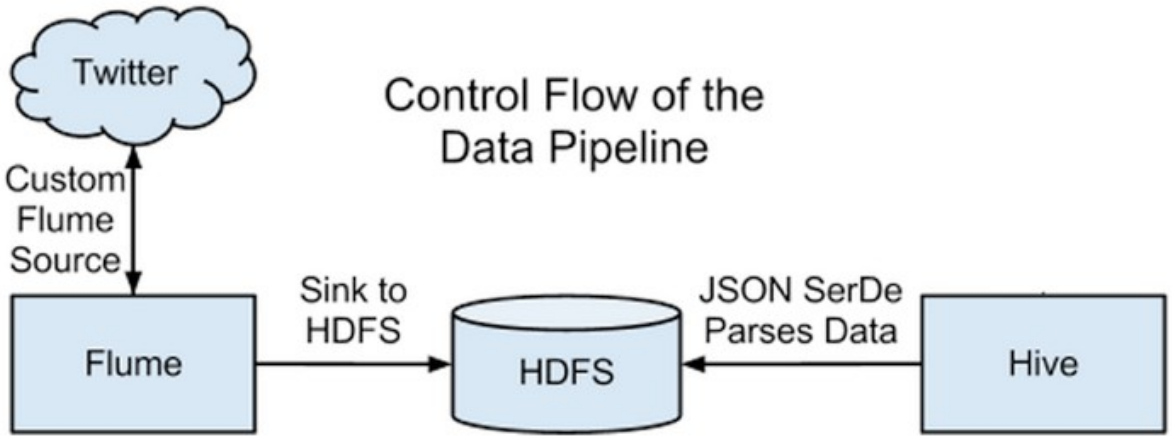


Odiago

Hands-On: Loading Twitter Data to Hadoop HDFS

Danairat T., , danairat@gmail.com: Thanachart Numnonda, thanachart@imcinstitute.com

Exercise Overview



Hive.apache.org

1. Installing Flume

Install Flume binary file

```
$ wget
http://apache.mirrors.hoobly.com/flume/1.4.0/apache
-flume-1.4.0-bin.tar.gz

$ tar -xvzf apache-flume-1.4.0-bin.tar.gz

$ sudo mv apache-flume-1.4.0-bin /usr/local

$ rm apache-flume-1.4.0-bin.tar.gz
```


Danairat T., , danairat@gmail.com: Thanachart Numnonda, thanachart@imcinstitute.com Apr 2015

1. Installing Flume (cont.)

Edit $HOME ./bashrc

```
$ sudo vi $HOME/.bashrc
```

```
export FLUME_HOME=/usr/local/apache-flume-1.4.0-bin
export PATH=$PATH:$HADOOP_PREFIX/bin:$JAVA_HOME/bin:$MAHOUT_HOME/bin:$HIVE_HOME/bin:$FLUME_HOME/bin
```

```
$ exec bash
```


Danairat T., , danairat@gmail.com: Thanachart Numnonda, thanachart@imcinstitute.com Apr 2015

2. Installing a jar file

Copy a jar file and edit conf file

```
$ wget http://files.cloudera.com/samples/flume-sources-1.0-SNAPSHOT.jar
$ sudo mv flume-sources-1.0-SNAPSHOT.jar /usr/local/apache-flume-1.4.0-bin/lib/
$ cd /usr/local/apache-flume-1.4.0-bin/conf/
$ sudo cp flume-env.sh.template flume-env.sh
$ sudo vi flume-env.sh
```

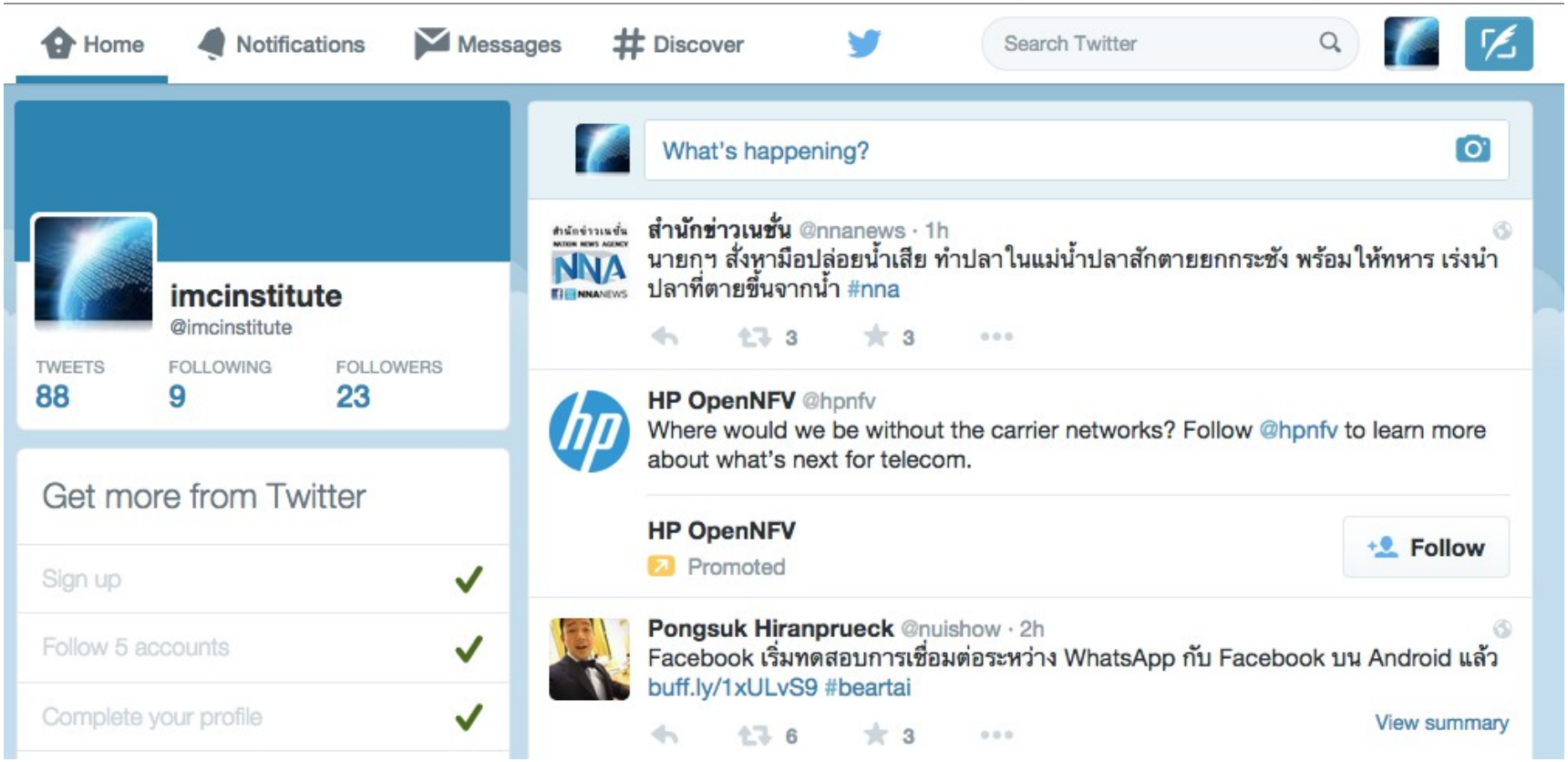
```
JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64

# Give Flume more memory and pre-allocate, enable remote monitoring via JMX
#JAVA_OPTS="-Xms100m -Xmx200m -Dcom.sun.management.jmxremote"

# Note that the Flume conf directory is always included in the classpath.
FLUME_CLASSPATH="/usr/local/apache-flume-1.4.0-bin/lib/flume-sources-1.0-SNAPSHOT.jar"
```

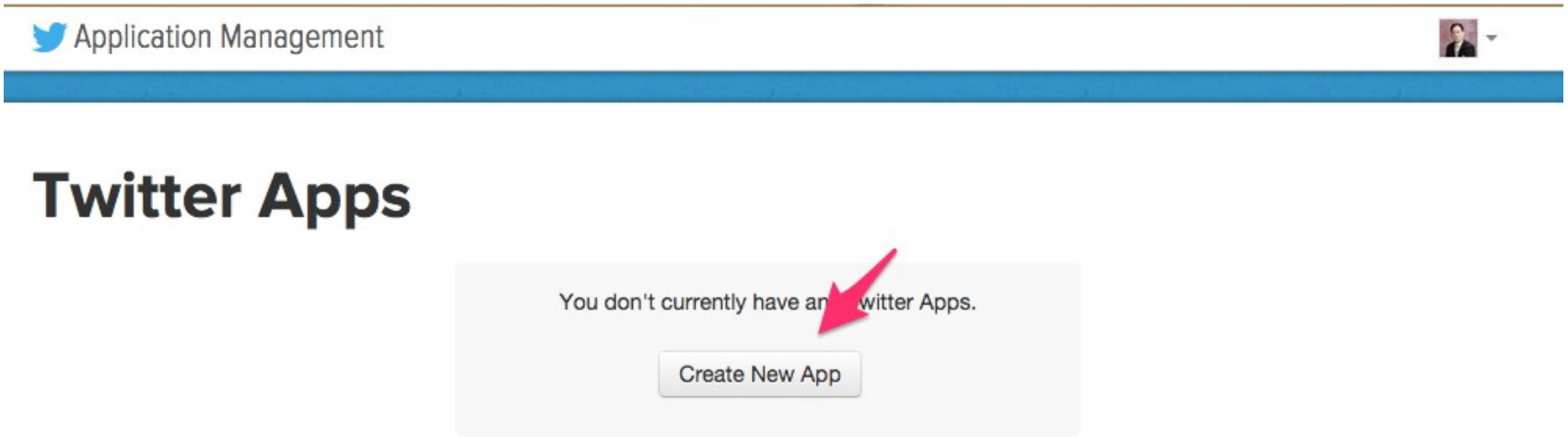
3. Create a new Twitter App

Login to your Twitter @ twitter.com



3. Create a new Twitter App (cont.)

Create a new Twitter App @ apps.twitter.com



3. Create a new Twitter App (cont.)

Enter all the details in the application:

Application Management

Create an application

Application Details

Name *

IMC_Institute_App

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

Description *

IMC Institute Demo App

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

Website *

http://www.imcinstitute.com

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.)

3. Create a new Twitter App (cont.)

Your application will be created:

https://apps.twitter.com/app/8158163

Application Management

Your application has been created. Please take a moment to review and adjust your application's settings.

IMC_Institute_App

Test OAuth

Details | Settings | Keys and Access Tokens | Permissions

IMC Institute Demo App

http://www.imcinstitute.com

Organization

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

Organization	None
Organization website	None

Application Settings

3. Create a new Twitter App (cont.)

Click on Keys and Access Tokens:

Application Management

IMC_Institute_App

Test OAuth

Details | Settings | Keys and Access Tokens | Permissions

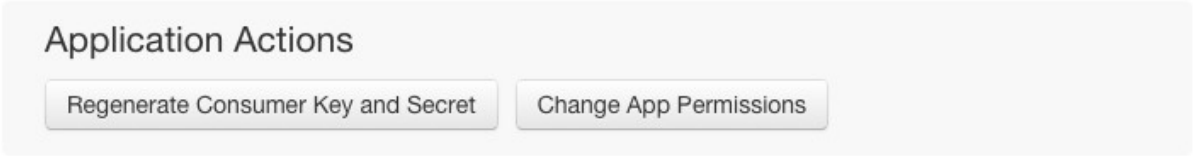
Application Settings

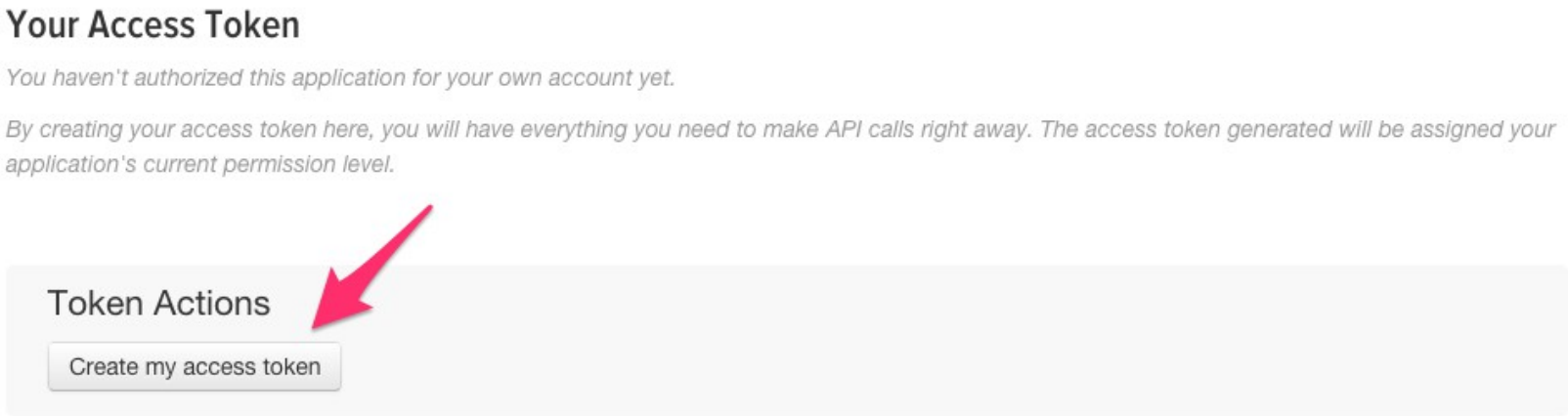
Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.

Consumer Key (API Key)	MjpswndxVj27ylnpOoSBrnfLX
Consumer Secret (API Secret)	QYmuBO1smD5Yc3zE0ZF9ByCgeEQxnxUmhRVCisAvPFudYVjC4a
Access Level	Read and write (modify app permissions)
Owner	imcinstitute
Owner ID	921172807

3. Create a new Twitter App (cont.)

Click on Keys and Access Tokens:





3. Create a new Twitter App (cont.)

Your Access token got created:

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	921172807-EfMXJj6as2dFECDH1vDe5goyTHcxPrF1RlJozqgx
Access Token Secret	HbpZEVip3D5j80GP21a37HxA4y10dH9BHcgEFXUNcA9xy
Access Level	Read and write
Owner	imcinstitute
Owner ID	921172807

Token Actions

Regenerate My Access Token and Token Secret | Revoke Token Access

4. Configuring the Flume Agent

```
Copy the flume.conf file from the following url:
https://github.com/cloudera/cdh-twitter-
example/blob/master/flume-sources/flume.conf

$ sudo vi /usr/local/apache-flume-1.4.0-bin/conf/flume.conf
```

flume.conf file

```
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 = MjpswmdxvjZ7ylnpOoSBrnfLX
TwitterAgent.sources.Twitter.consumerSecret = QYmuBO1smD5Yc3zE0ZF9ByCgeEQxnxUmhRVCisAvPFudYVjC4a
TwitterAgent.sources.Twitter.accessToken = 921172807-EfMXJj6as2dFECDH1vDe5goyTHcxPrF1RIJozqgx
TwitterAgent.sources.Twitter.accessTokenSecret = HbpZEVip3D5j80GP21a37HxA4y10dH9BHcgEFXUNcA9xv
TwitterAgent.sources.Twitter.keywords = hadoop, big data, analytics, bigdata, cloudera, data science, data scientiest, business in
telligence, 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://localhost:54310/user/flume/tweets
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
~
```

5. Fetching the data from twitter

```
$ flume-ng agent -n TwitterAgent -c conf -f
/usr/local/apache-flume-1.4.0-bin/conf/flume.conf
```

Wait for 60-90 seconds and let flume stream the data on HDFS, then press Ctrl-c to break the command and stop the streaming. (Ignore the exceptions)

```
15/04/06 15:24:04 INFO instrumentation.MonitoredCounterGroup: Component type: SINK, name: HDFS started
15/04/06 15:24:04 INFO twitter4j.TwitterStreamImpl: Establishing connection.
15/04/06 15:24:07 INFO twitter4j.TwitterStreamImpl: Connection established.
15/04/06 15:24:07 INFO twitter4j.TwitterStreamImpl: Receiving status stream.
15/04/06 15:24:07 INFO hdfs.HDFSDataStream: Serializer = TEXT, UseRawLocalFileSystem = false
15/04/06 15:24:07 INFO hdfs.BucketWriter: Creating hdfs://localhost:54310/user/flume/tweets/FlumeData.1428333847150.tmp
15/04/06 15:24:37 INFO hdfs.BucketWriter: Renaming hdfs://localhost:54310/user/flume/tweets/FlumeData.1428333847150.tmp to hdfs://
localhost:54310/user/flume/tweets/FlumeData.1428333847150
15/04/06 15:24:37 INFO hdfs.BucketWriter: Creating hdfs://localhost:54310/user/flume/tweets/FlumeData.1428333847151.tmp
15/04/06 15:25:07 INFO hdfs.BucketWriter: Renaming hdfs://localhost:54310/user/flume/tweets/FlumeData.1428333847151.tmp to hdfs://
localhost:54310/user/flume/tweets/FlumeData.1428333847151
15/04/06 15:25:08 INFO hdfs.BucketWriter: Creating hdfs://localhost:54310/user/flume/tweets/FlumeData.1428333847152.tmp
```

6. View the straming data

```
$ hadoop fs -ls /user/flume/tweets
```

```
Found 3 items
-rw-r--r--   1 thanachart_imcinstitute_com supergroup      73871 2015-04-06 15:24 /user/flume/tweets/FlumeData.1428333847150
-rw-r--r--   1 thanachart_imcinstitute_com supergroup      87697 2015-04-06 15:24 /user/flume/tweets/FlumeData.1428333847151
-rw-r--r--   1 thanachart_imcinstitute_com supergroup      66087 2015-04-06 15:25 /user/flume/tweets/FlumeData.1428333847152.tmp
```

```
$ hadoop fs -cat /user/flume/tweets/FlumeData.1428333847150
```


```
"filter_level":"low","retweeted":false,"in_reply_to_screen_name":null,"possibly_sensitive":false,"truncated":false,"lang":"en","i
n_reply_to_status_id_str":null,"id":585096165642805248,"in_reply_to_user_id_str":null,"timestamp_ms":"1428332771107","in_reply_to_
status_id":null,"created_at":"Mon Apr 06 15:06:11 +0000 2015","favorite_count":0,"place":null,"coordinates":null,"text":"5 Common
HR Goals and How Big Data Can Help You Achieve Them #BigDataHR http://t.co/tiBwCsuFre","contributors":null,"geo":null,"entities":{
"trends":[],"symbols":[],"urls":[{"expanded_url":"http://rightrelevance.com/tw/bigdatarr/28e75c7a474b3071d614e8acc7f6b9cfe62ca2d6/
big%20data/big%20data","indices":[72,94],"display_url":"rightrelevance.com/tw/bigdatarr/2\u2026","url":"http://t.co/tiBwCsuFre"}],
"hashtags":[{"text":"BigDataHR","indices":[61,71]}],"user_mentions":[]},"source":"<a href=\"http://www.rightrelevance.com\" rel=\"
nofollow\">BigDataRR<\/a>","favorited":false,"in_reply_to_user_id":null,"retweet_count":0,"id_str":"585096165642805248","user":{"l
ocation":"","default_profile":true,"profile_background_tile":false,"statuses_count":3364,"lang":"en","profile_link_color":"0084B4"
```


```
$ hadoop fs -rm /user/flume/tweets/*.tmp
```

7. Analyse data using Hive

Get a Serde Jar File for parsing JSON file

```
$ wget
http://files.cloudera.com/samples/hive-serdes-1.0-SNAPSHOT.jar

$ mv hive-serdes-1.0-SNAPSHOT.jar  /usr/local/apache-hive-
1.1.0-bin/lib/

$ hive
```

Register the Jar file.

```
hive> ADD JAR /usr/local/apache-hive-1.1.0-bin/lib/hive-
serdes-1.0-SNAPSHOT.jar;
```


Danairat T., , danairat@gmail.com: Thanachart Numnonda, thanachart@imcinstitute.com Apr 2015

7. Analyse data using Hive (cont.)

Running the following hive command

```
CREATE EXTERNAL TABLE tweets (
   id BIGINT,
   created_at STRING,
   source STRING,
   favorited BOOLEAN,
   retweet_count INT,
   retweeted_status STRUCT<
      text:STRING,
      user:STRUCT<screen_name:STRING,name:STRING>>,
   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
)
ROW FORMAT SERDE 'com.cloudera.hive.serde.JSONSerDe'
LOCATION '/user/flume/tweets';
```

http://www.thecloudavenue.com/2013/03/analyse-tweets-using-flume-hadoop-and.html

7. Analyse data using Hive (cont)

Finding user who has the most number of followers

```
hive> elect user.screen_name, user.followers_count c from
tweets order by c desc;
```

```
Starting Job = job_201504051617_0010, Tracking URL = http://localhost:50030/jobdetails.jsp?jobid=job_201504051617_0010
Kill Command = /usr/local/hadoop/libexec/../bin/hadoop job  -kill job_201504051617_0010
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2015-04-06 15:37:27,782 Stage-1 map = 0%,  reduce = 0%
2015-04-06 15:37:31,837 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 1.27 sec
2015-04-06 15:37:39,899 Stage-1 map = 100%,  reduce = 33%, Cumulative CPU 1.27 sec
2015-04-06 15:37:40,908 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 2.42 sec
MapReduce Total cumulative CPU time: 2 seconds 420 msec
Ended Job = job_201504051617_0010
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 2.42 sec   HDFS Read: 170686 HDFS Write: 687 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 420 msec
OK
vinnaum 11523
navchatterji    5485
HCITExpert      4751
NWDCScoop       4097
7wdata  3005
MotivasiMariaP  2007
WesleyBackelant 1977
IFTTTMarketing  1307
jonathangibs    968
ephraimcohen    914
feshob  716
DKajouri        713
```


Danairat T., , danairat@gmail.com: Thanachart Numnonda, thanachart@imcinstitute.com Apr 2015

Thank you

www.imcinstitute.com
www.facebook.com/imcinstitute

Danairat T., , danairat@gmail.com: Thanachart Numnonda, thanachart@imcinstitute.com