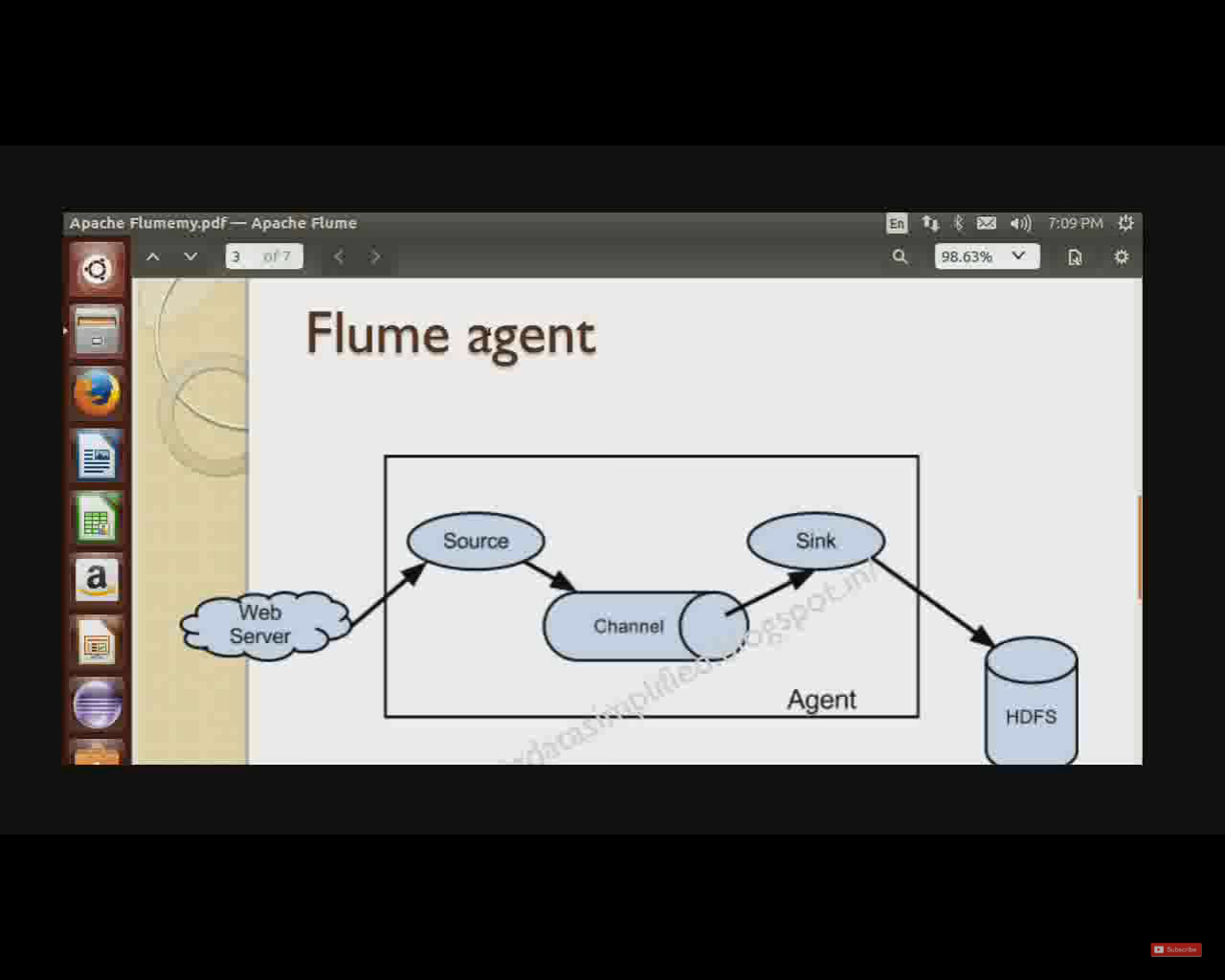
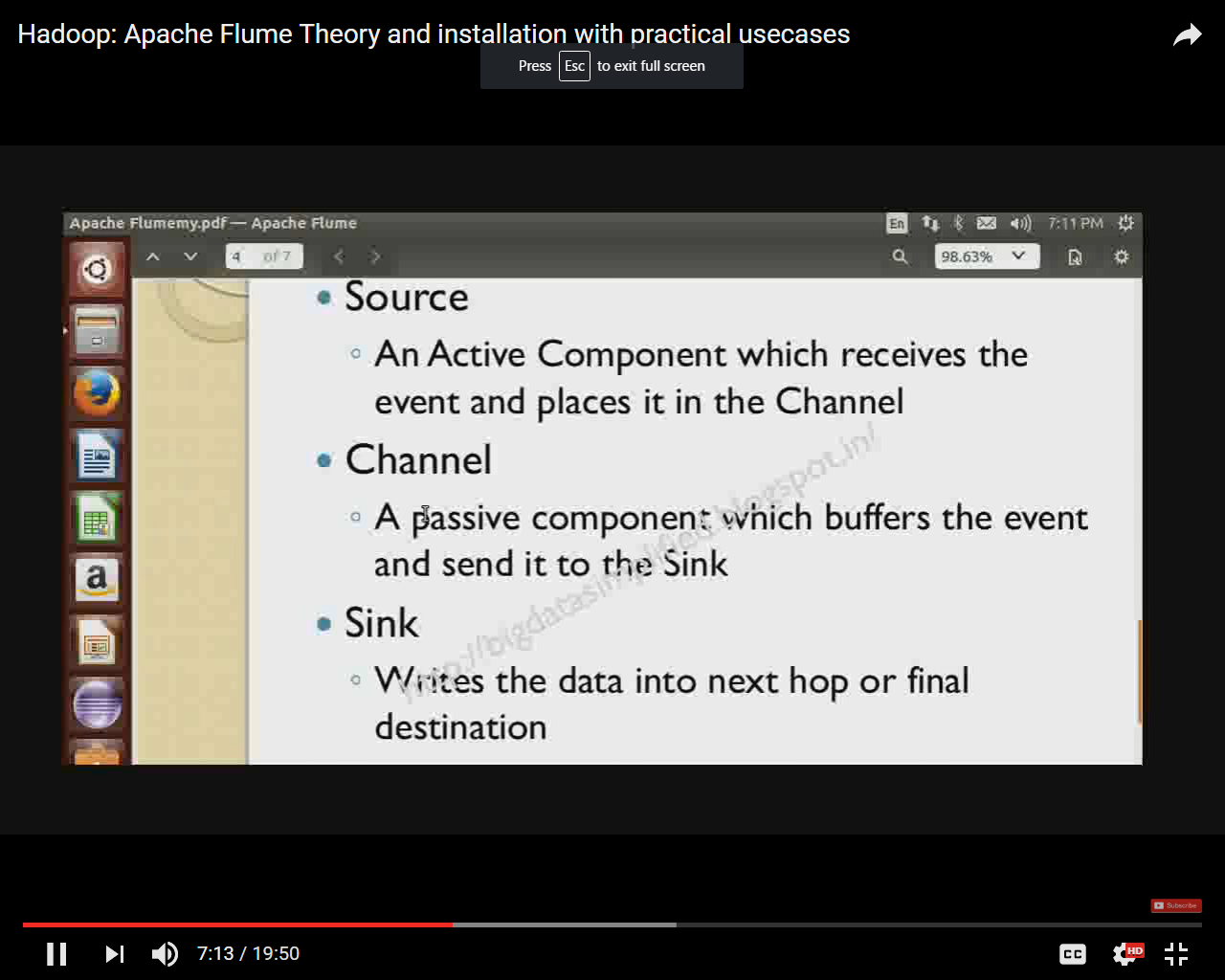
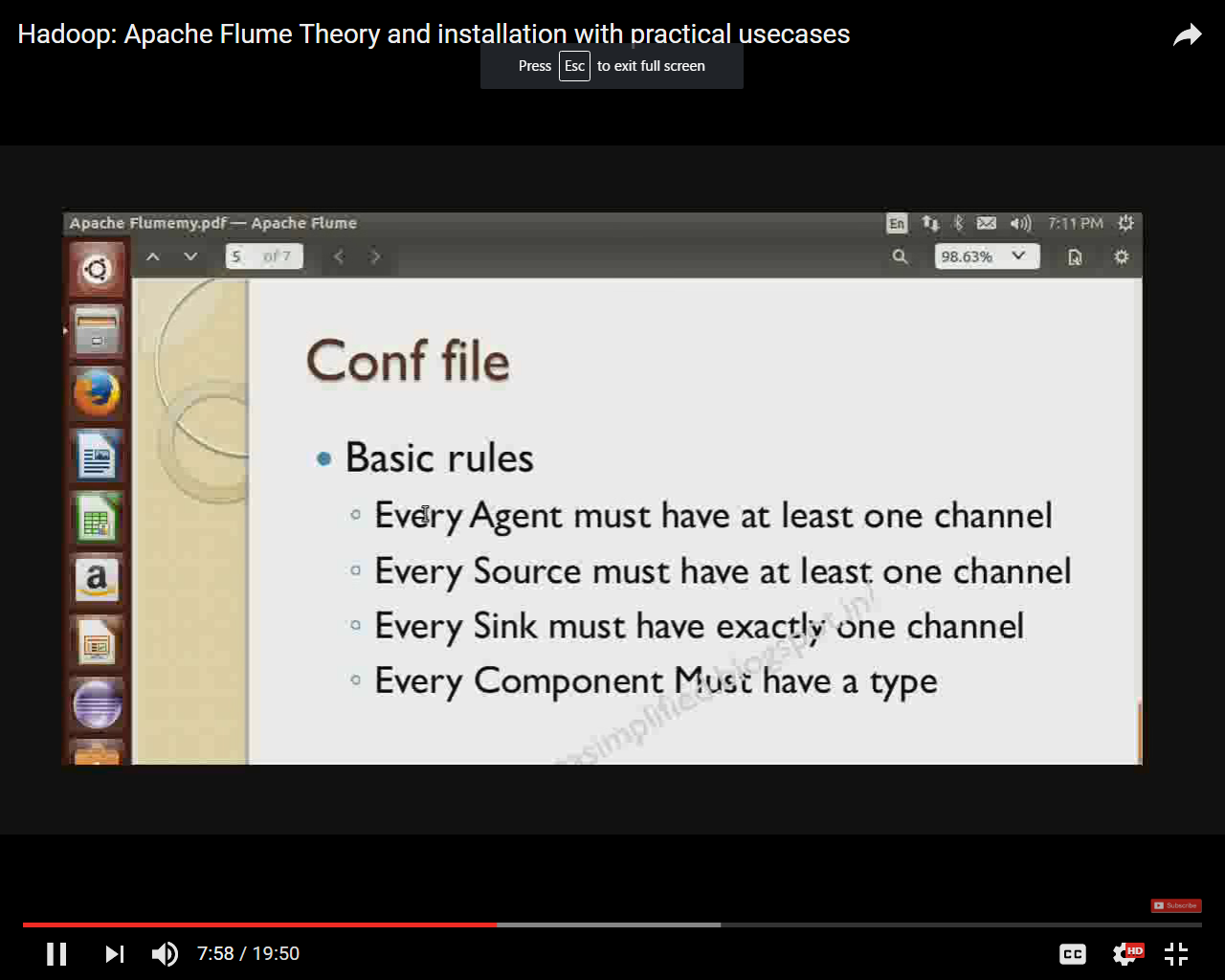
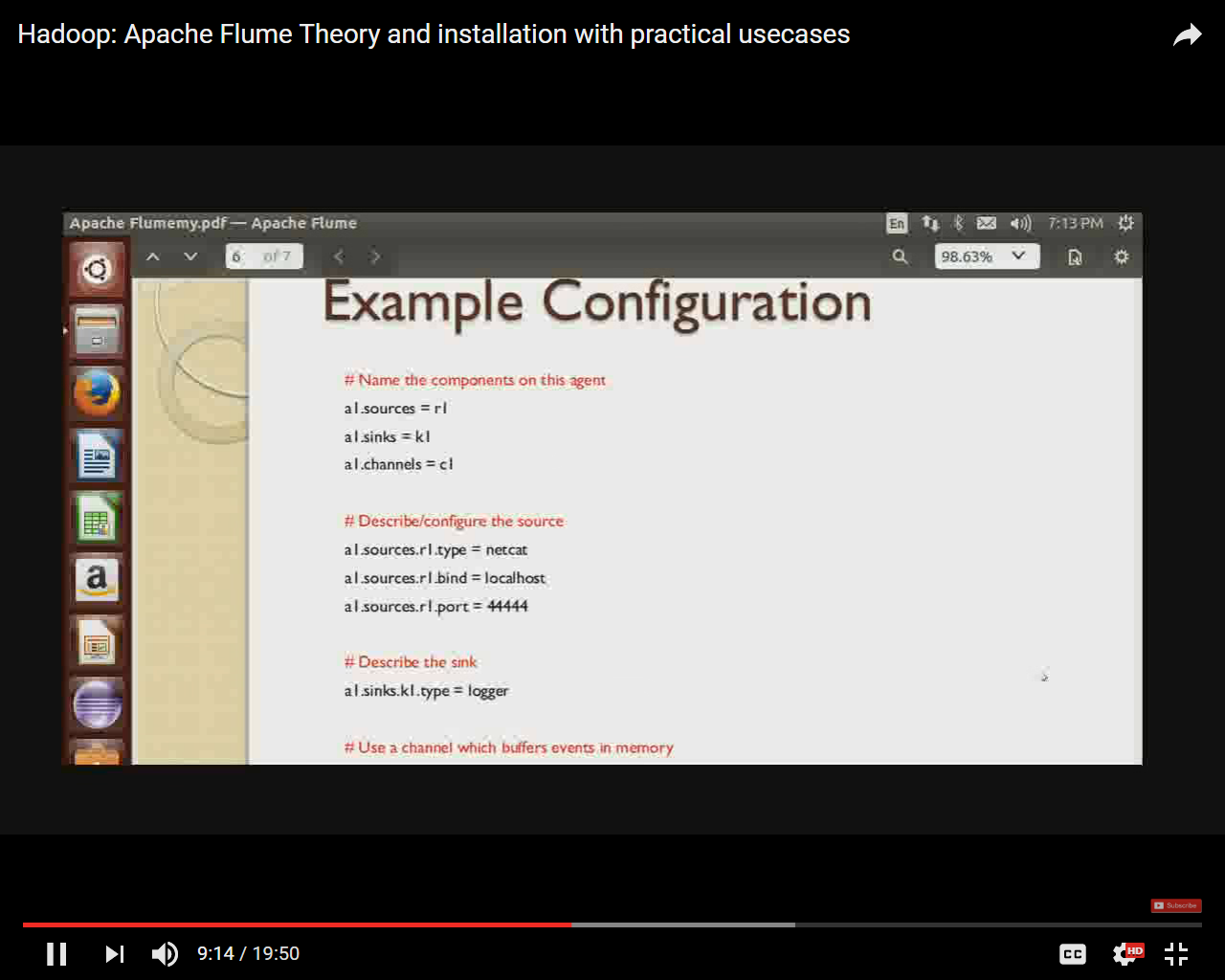
**Flume Architecture**



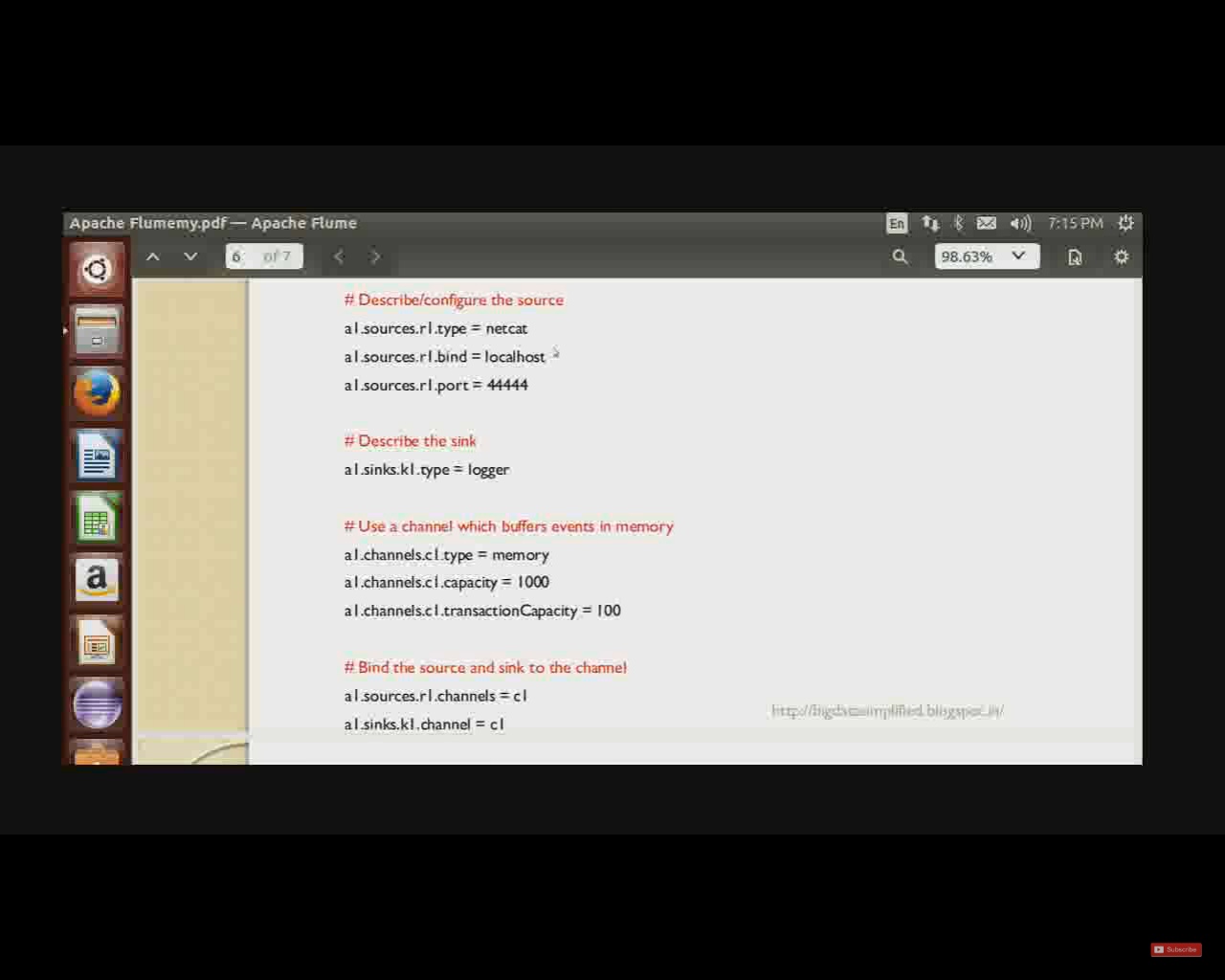






Here a1 is the name of agent, it can be anything

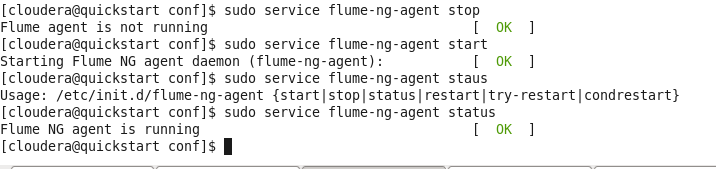
A1.sources = r1 r2 ( if we have two channels just give the name space separted)



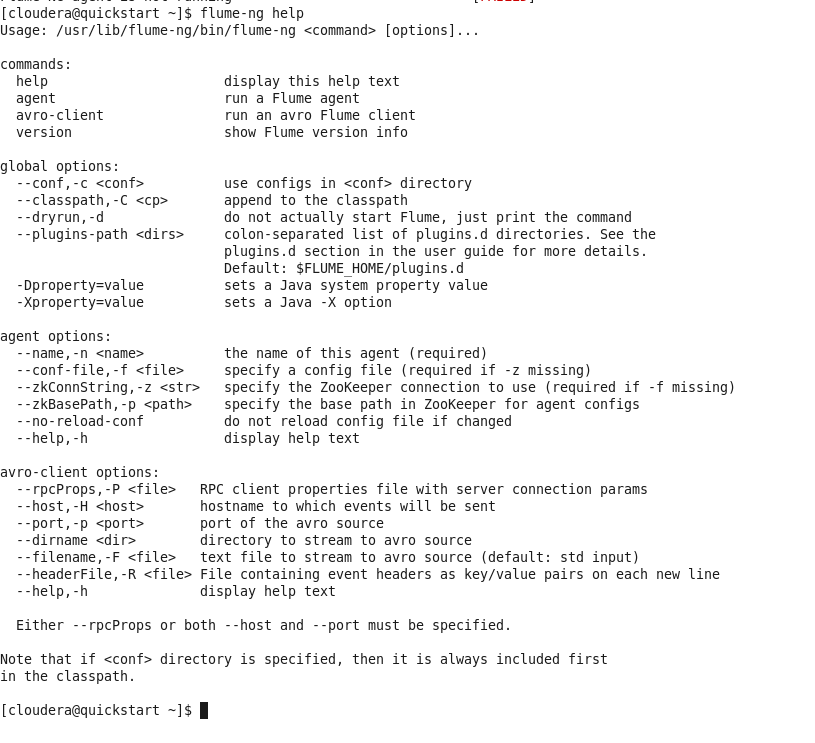
Channels type can be memory or jdbc , capacity are in bytes

*Flume is only a daemon we just need to start Hadoop , no need to start anything for flume*

These are just some commands



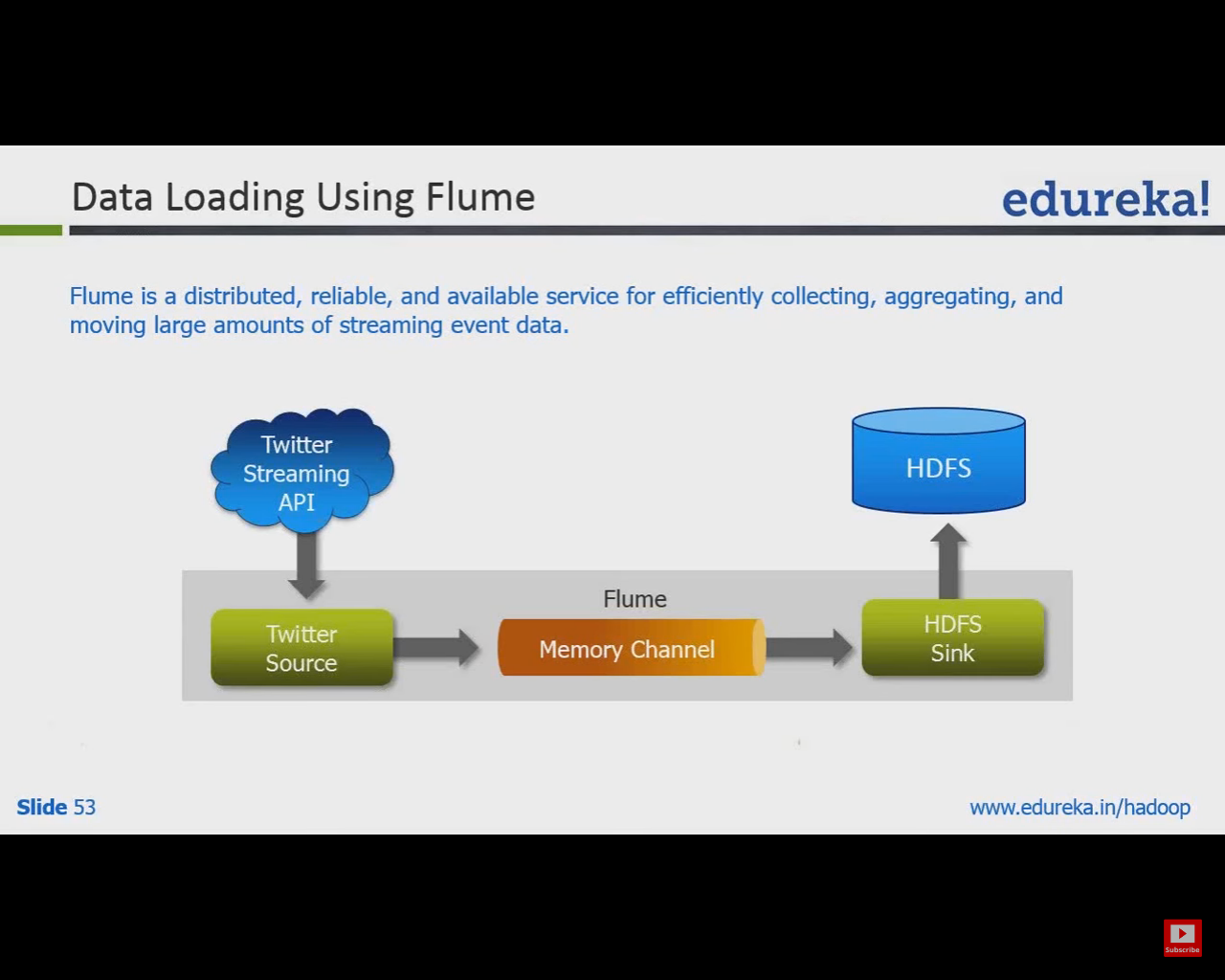
Verifying flume installation



if Flume is installed via an RPM or Debian package, you can use the following commands to start, stop, and restart the Flume agent via init scripts:

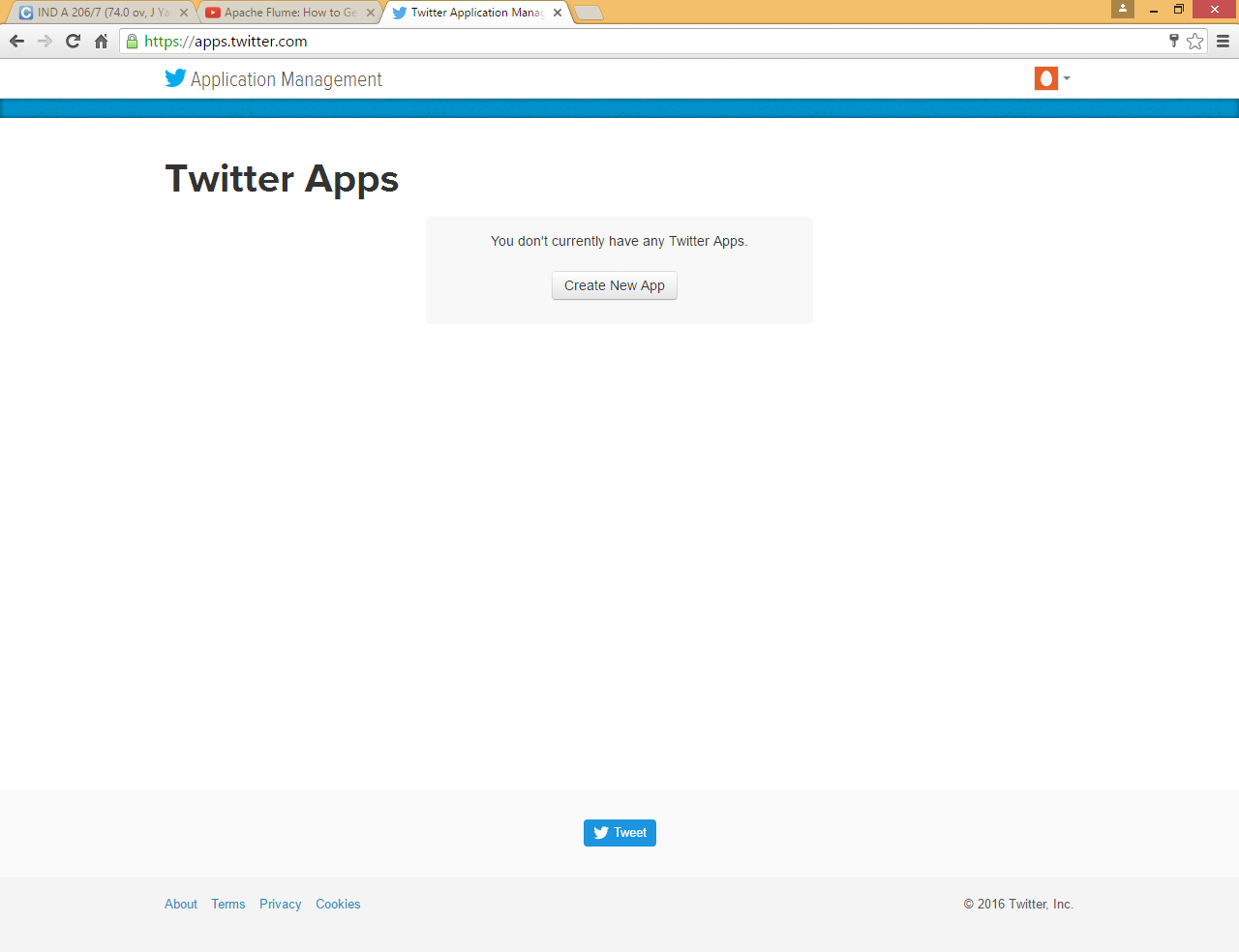
|  |
| --- |
| sudo service flume-ng-agent stop  sudo service flume-ng-agent start  sudo service flume-ng-agent restart  sudo service flume-ng-agent status |

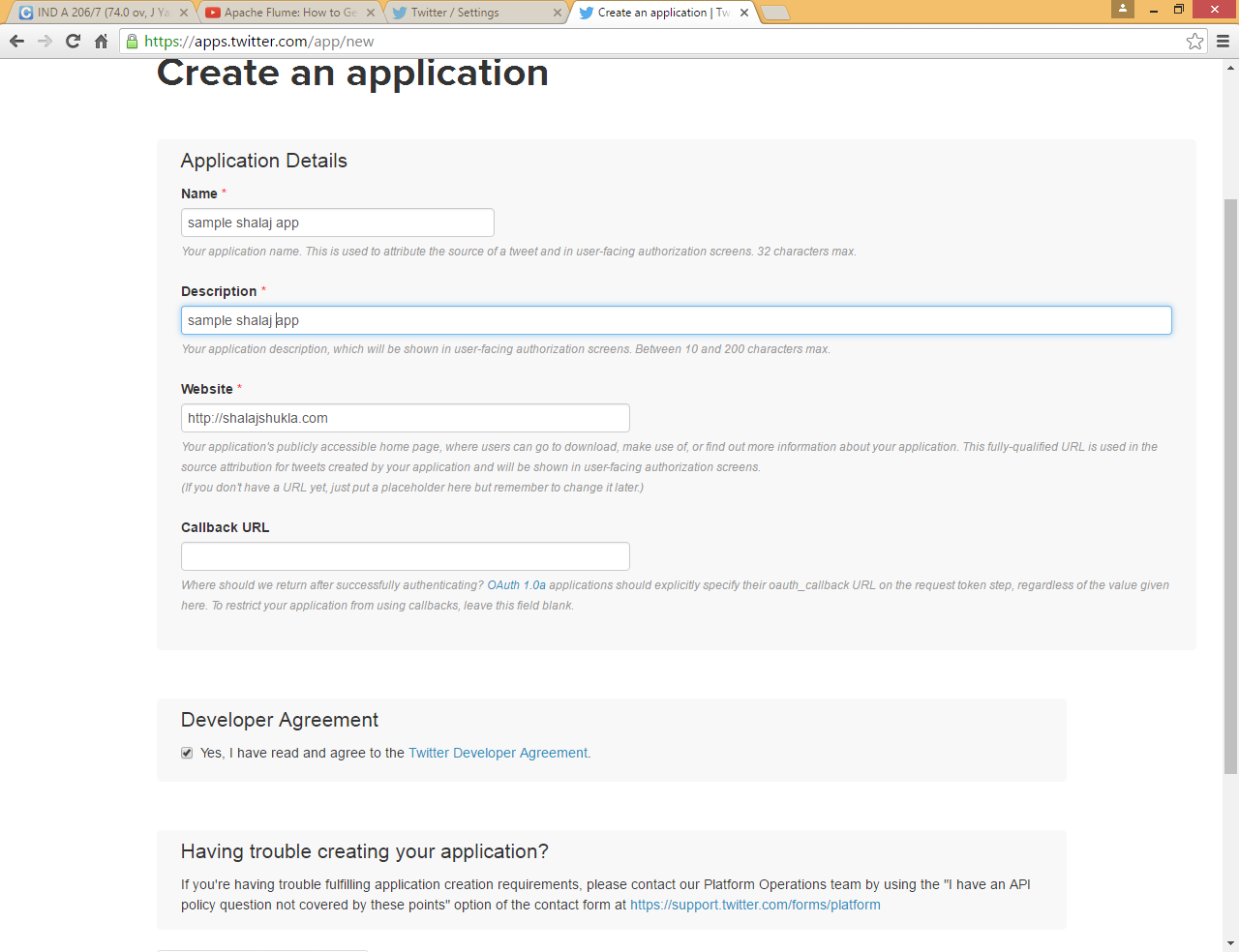
**Here we are trying to extract twitter data and store it in HDFS system**

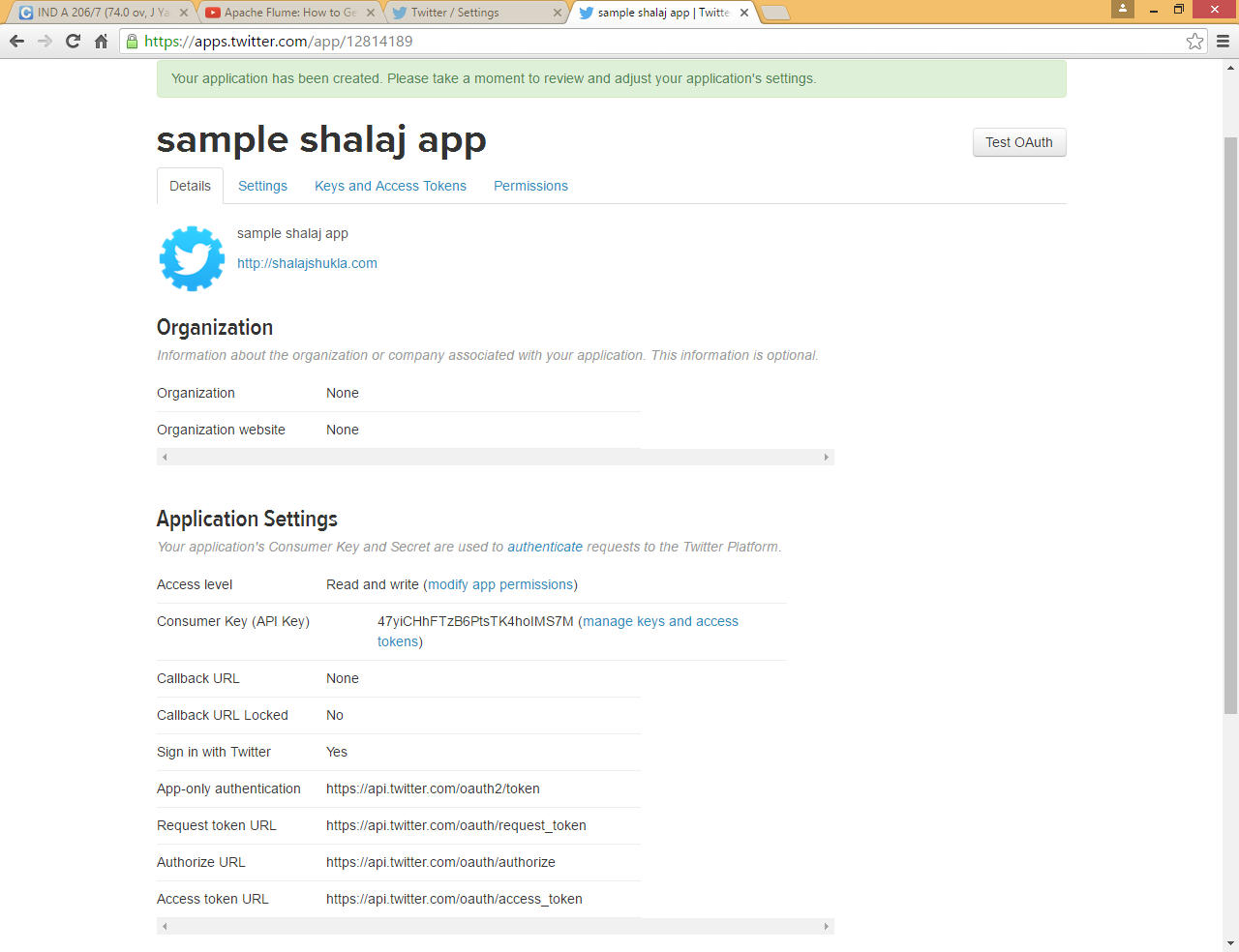


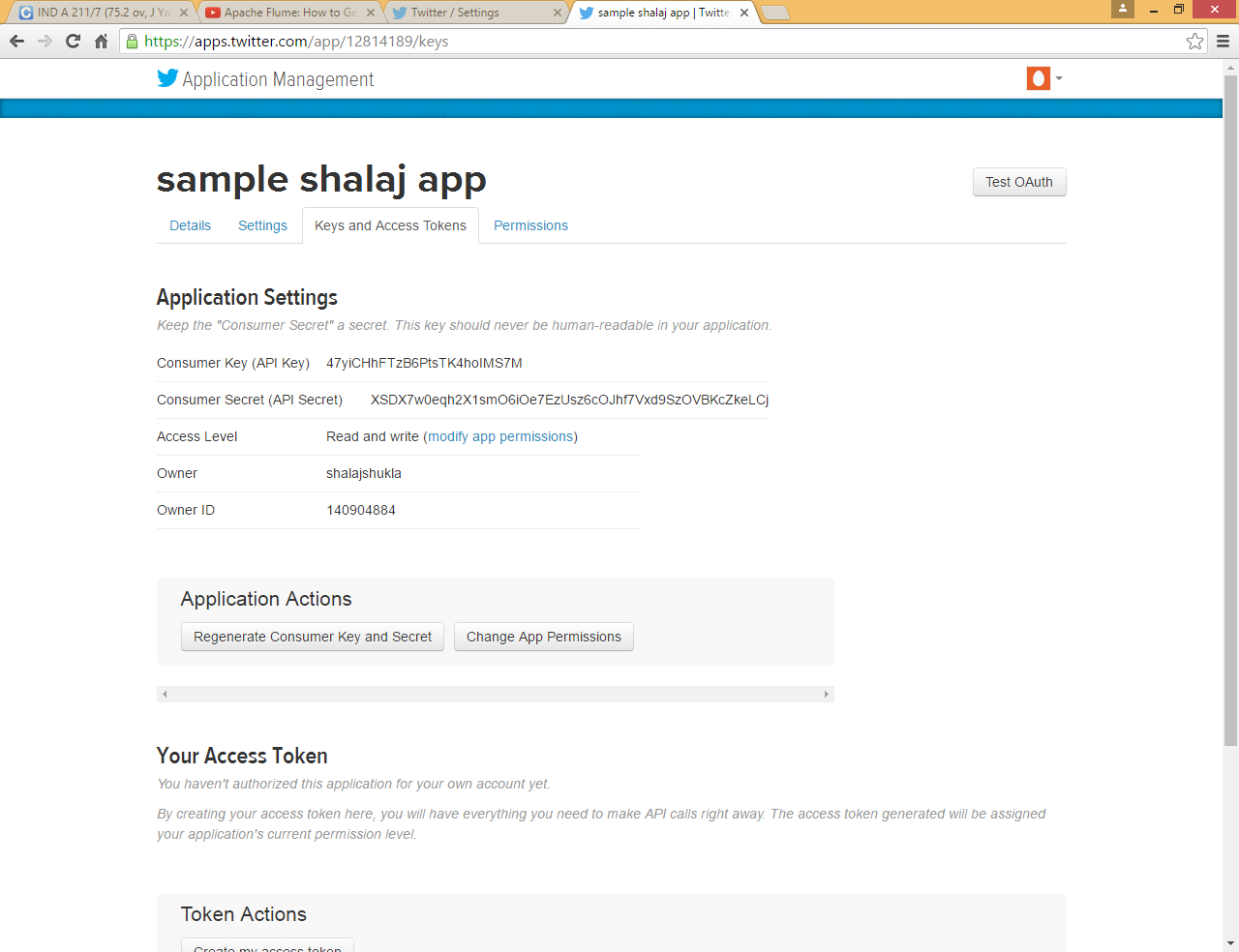
Go to <https://apps.twitter.com/> and Create Application

Prerequisite – you need to have twitter account and mobile number should be registered



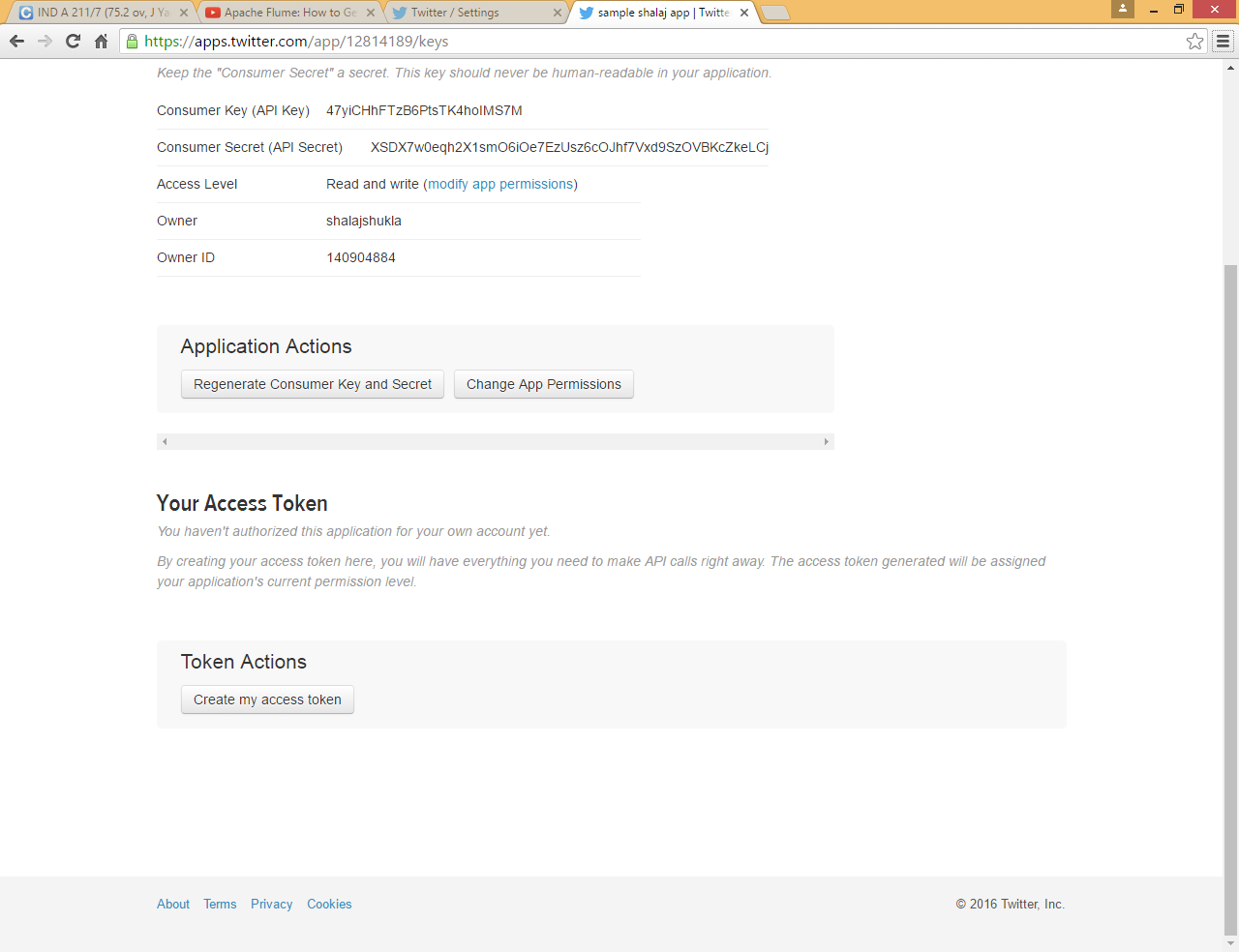




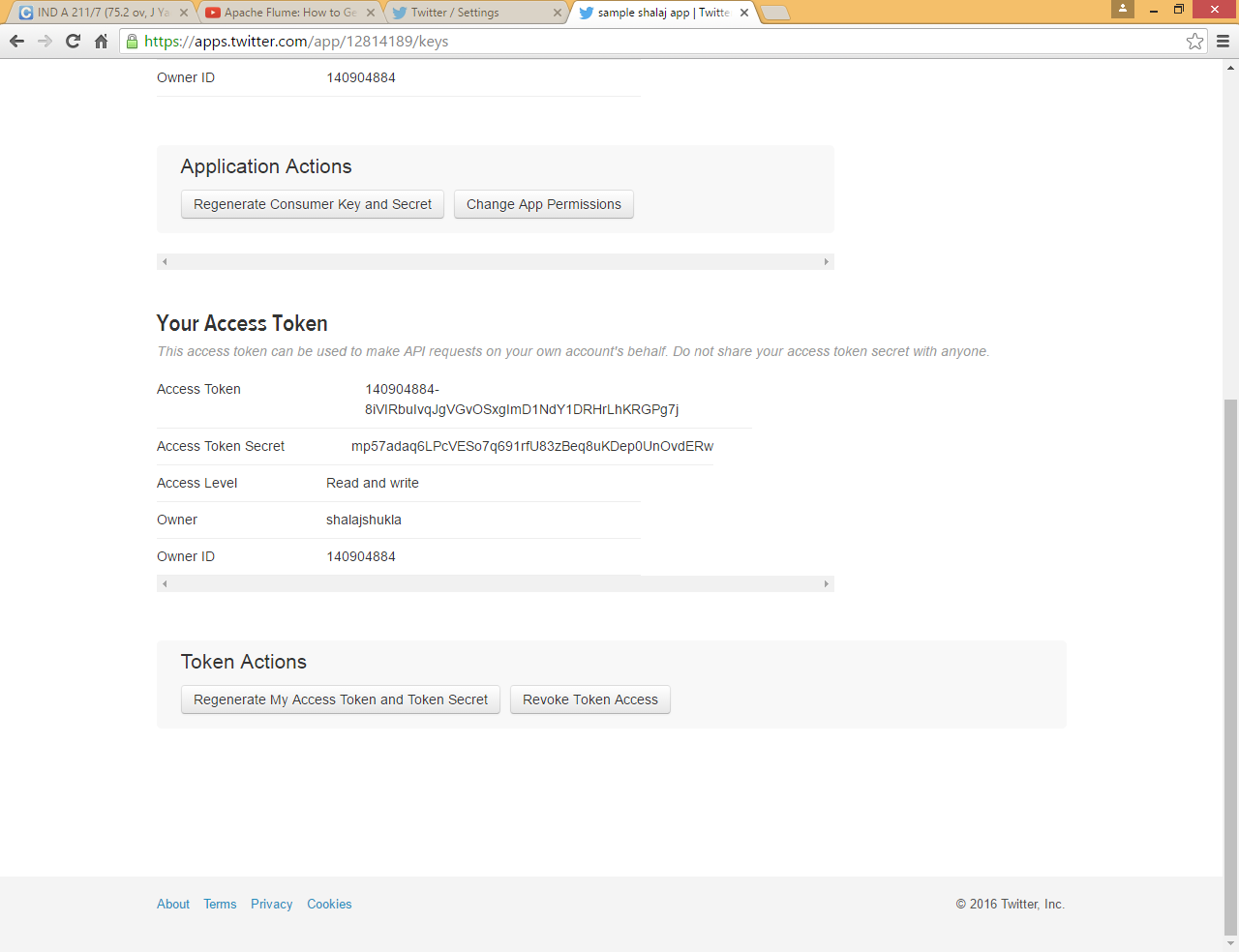


Consumer Key (API Key) 47yiCHhFTzB6PtsTK4hoIMS7M

Consumer Secret (API Secret) XSDX7w0eqh2X1smO6iOe7EzUsz6cOJhf7Vxd9SzOVBKcZkeLCj



Click create my access token



Access Token 140904884-8iVIRbuIvqJgVGvOSxgImD1NdY1DRHrLhKRGPg7j

Access Token Secret mp57adaq6LPcVESo7q691rfU83zBeq8uKDep0UnOvdERw

Create flume-twitter.conf file

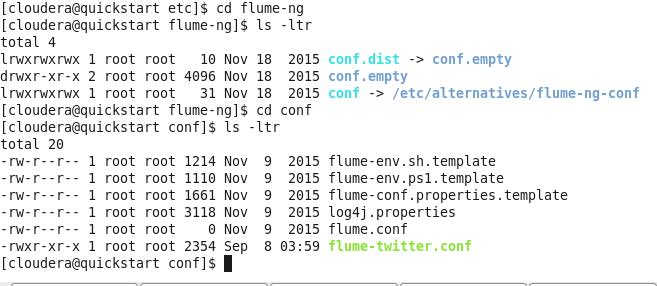
|  |
| --- |
| # Licensed to the Apache Software Foundation (ASF) under one  # or more contributor license agreements. See the NOTICE file  # distributed with this work for additional information  # regarding copyright ownership. The ASF licenses this file  # to you under the Apache License, Version 2.0 (the  # "License"); you may not use this file except in compliance  # with the License. You may obtain a copy of the License at  #  # http://www.apache.org/licenses/LICENSE-2.0  #  # Unless required by applicable law or agreed to in writing,  # software distributed under the License is distributed on an  # "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY  # KIND, either express or implied. See the License for the  # specific language governing permissions and limitations  # under the License.  # 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 = 47yiCHhFTzB6PtsTK4hoIMS7M  TwitterAgent.sources.Twitter.consumerSecret = XSDX7w0eqh2X1smO6iOe7EzUsz6cOJhf7Vxd9SzOVBKcZkeLCj  TwitterAgent.sources.Twitter.accessToken = 140904884-8iVIRbuIvqJgVGvOSxgImD1NdY1DRHrLhKRGPg7j  TwitterAgent.sources.Twitter.accessTokenSecret = mp57adaq6LPcVESo7q691rfU83zBeq8uKDep0UnOvdERw  TwitterAgent.sources.Twitter.keywords = hadoop, big data  TwitterAgent.sinks.HDFS.channel = MemChannel  TwitterAgent.sinks.HDFS.type = hdfs  TwitterAgent.sinks.HDFS.hdfs.path = hdfs://quickstart.cloudera:8020//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 |

Find hdfs hostname and port from /etc/Hadoop/core-site.xml

|  |
| --- |
| <property>  <name>fs.defaultFS</name>  <value>hdfs://quickstart.cloudera:8020</value>  </property> |

Copy this file under /etc/flume-ng/conf

|  |
| --- |
| *[cloudera@quickstart conf.pseudo]$ sudo su -*  *[root@quickstart ~]# cp /home/cloudera/user/training/flume-twitter.conf /etc/flume-ng/conf*  *[root@quickstart ~]#* |



[cloudera@quickstart conf]$ sudo cp flume-env.sh.template flume-env.sh

[cloudera@quickstart conf]$ sudo vi flume-env.sh

|  |
| --- |
| # Licensed to the Apache Software Foundation (ASF) under one  # or more contributor license agreements. See the NOTICE file  # distributed with this work for additional information  # regarding copyright ownership. The ASF licenses this file  # to you under the Apache License, Version 2.0 (the  # "License"); you may not use this file except in compliance  # with the License. You may obtain a copy of the License at  #  # http://www.apache.org/licenses/LICENSE-2.0  #  # Unless required by applicable law or agreed to in writing, software  # distributed under the License is distributed on an "AS IS" BASIS,  # WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  # See the License for the specific language governing permissions and  # limitations under the License.  # If this file is placed at FLUME\_CONF\_DIR/flume-env.sh, it will be sourced  # during Flume startup.  # Enviroment variables can be set here.  # export JAVA\_HOME=/usr/lib/jvm/java-6-sun  # Give Flume more memory and pre-allocate, enable remote monitoring via JMX  # export JAVA\_OPTS="-Xms100m -Xmx2000m -Dcom.sun.management.jmxremote"  # Note that the Flume conf directory is always included in the classpath.  FLUME\_CLASSPATH="/home/cloudera/user/training/flume-sources-1.0-SNAPSHOT.jar" |

Go to /usr/lib/flume-ng/lib and rename twitter\*.jar to twitter\*.org

|  |
| --- |
| [cloudera@quickstart lib]$ ls twi\*  twitter4j-core-3.0.3.jar twitter4j-media-support-3.0.3.jar twitter4j-stream-3.0.3.jar  [cloudera@quickstart lib]$ sudo mv twitter4j-core-3.0.3.jar twitter4j-core-3.0.3.org  [cloudera@quickstart lib]$ sudo mv twitter4j-media-support-3.0.3.jar twitter4j-media-support-3.0.3.org  [cloudera@quickstart lib]$ sudo mv twitter4j-stream-3.0.3.jar twitter4j-stream-3.0.3.org  [cloudera@quickstart lib]$ ls twi\*  twitter4j-core-3.0.3.org twitter4j-media-support-3.0.3.org twitter4j-stream-3.0.3.org  [cloudera@quickstart lib]$ |

Place flume jar , you could download it from [here](http://files.cloudera.com/samples/flume-sources-1.0-SNAPSHOT.jar).

|  |
| --- |
| [cloudera@quickstart conf.pseudo]$ sudo cp /home/cloudera/user/training/flume-sources-1.0-SNAPSHOT.jar /usr/lib/flume-ng/lib/  [cloudera@quickstart conf.pseudo]$ |

You can also run the agent in the foreground directly by using the flume-ng agent command:

$ /usr/bin/flume-ng agent -c <config-dir> -f <config-file> -n <agent-name>

For example:

$ /usr/bin/flume-ng agent -c /etc/flume-ng/conf -f /etc/flume-ng/conf/flume.conf -n agent

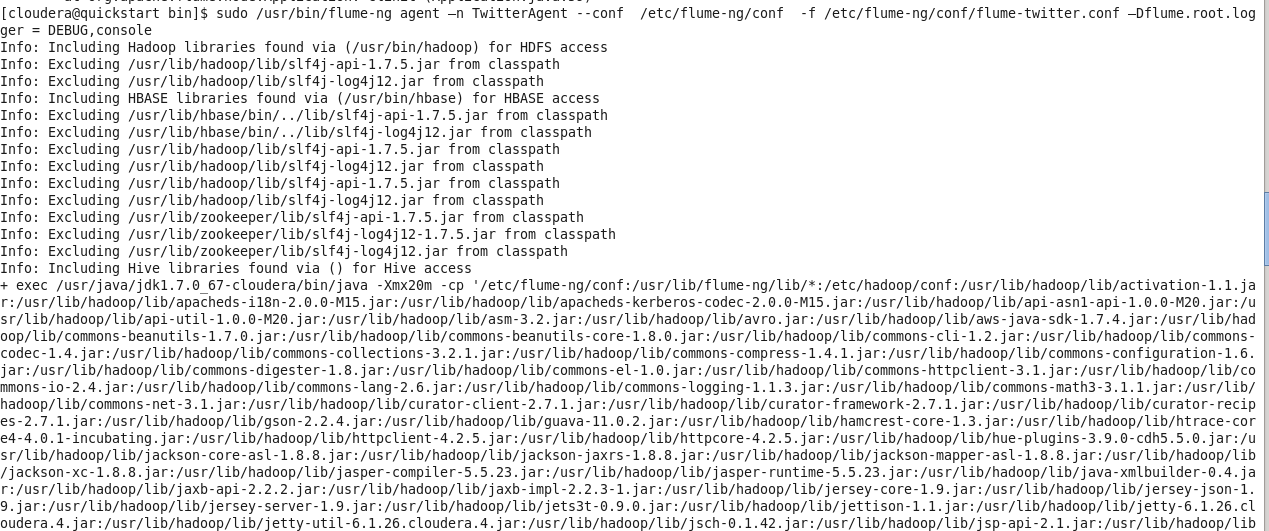
Now Run below command

|  |
| --- |
| [cloudera@quickstart bin]$ sudo /usr/bin/flume-ng agent -n TwitterAgent --conf /etc/flume-ng/conf -f /etc/flume-ng/conf/flume-twitter.conf |

OR

If you want to see log on same console use below command

|  |
| --- |
| [cloudera@quickstart bin]$ sudo /usr/bin/flume-ng agent -n TwitterAgent --conf /etc/flume-ng/conf -f /etc/flume-ng/conf/flume-twitter.conf -Dflume.root.logger=DEBUG,console |



You can find log under **/var/log/flume-ng/flume.log**

**Getiing below error**

|  |
| --- |
| 08 Sep 2016 20:58:27,093 INFO [Twitter Stream consumer-1[Waiting for 16000 milliseconds]] (twitter4j.internal.logging.SLF4JLogger.info:83) - Establishing connection.  08 Sep 2016 20:58:27,797 INFO [Twitter Stream consumer-1[Establishing connection]] (twitter4j.internal.logging.SLF4JLogger.info:83) - sun.security.validator.ValidatorException: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target  08 Sep 2016 20:58:27,798 INFO [Twitter Stream consumer-1[Establishing connection]] (twitter4j.internal.logging.SLF4JLogger.info:83) - Waiting for 16000 milliseconds |

Check site

<http://evanthika.blogspot.in/2014/01/how-to-solve-pkix-path-building-failed.html>

|  |
| --- |
| 1. Go to https://twitter.com/, click on the lock icon at the address bar, click on the 'Connection' tab, then click on the link 'Certificate Information'. From the 'Certificate Viewer', select the tab 'Details' and click on the 'Export' button and download the certificate (twitter.com) to a preferred location.  2. Once downloaded, issue the below command to import the public certificate of Twitter to the client-truststore.jks.  **$ keytool -importcert -file $somepath/twitter.com -keystore $ESB\_HOME/repository/resources/security/client-truststore.jks -alias "Twitter"**  3. Restart the WSO2 ESB server and invoke the API again and you will get the expected result. |

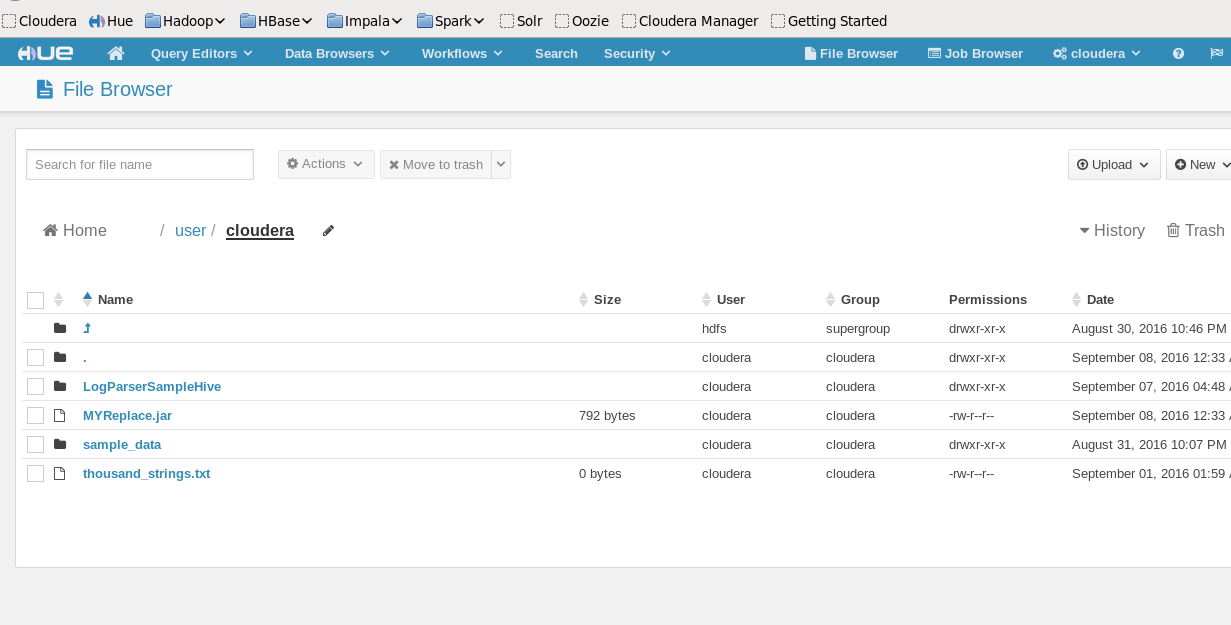
Check below link to add certificate in java keystore

<http://www.grim.se/guide/jre-cert>

|  |
| --- |
| keytool -import -alias alias -keystore path-to-jre/lib/security/cacerts -file path-to-certificate-file  Example:  keytool -import -alias sunas -keystore /opt/jdk1.6/jre/lib/security/cacerts -file /home/gugrim/tmp/sunas.der  You will be prompted for the keystore password, which is by default **changeit**.  Tried below but didn’t work  sudo keytool -import -alias twitter -keystore cacerts -file /home/cloudera/user/training/TwitterBase64.cer  sudo keytool -import -alias twitter -keystore cacerts -file /home/cloudera/user/training/TwitterDER.cer |

If it work fine you will get the data in hdfs file system --- under flume/twitter

You can see these files in file explorer under hue or terminal



We will get files with json contents, later we can store these json data in hive using JSONSerDe and do our data analysis