|  |  |
| --- | --- |
| Alumno | Edwin Giovanni Gonzalez Mata |
|  | Ejercicio Final Individual - Extractores de Datos |

# Ejercicio 1 (Flume)

En este ejercicio vamos a utilizar Flume para consumir eventos syslog y almacenarlos en HDFS:



Parte 1

Ruta y nombre del archivo : /etc/flume/conf/Flume-Ejercicio1.properties

agent1.sources = source1

agent1.sinks = sink1

agent1.channels = channel1

agent1.sources.source1.channels = channel1

agent1.sinks.sink1.channel = channel1

agent1.sources.source1.type = syslogudp

agent1.sources.source1.port = 5140

agent1.sources.source1.host = localhost

# Configurando Sink

agent1.sinks.sink1.type = hdfs

#especificamos la ruta destino en hdfs:

agent1.sinks.sink1.hdfs.path = /tmp/flume/syslog

#prefijo para los ficheros que cree flume

agent1.sinks.sink1.hdfs.filePrefix = logs

#prefijo para los ficheros que estan siendo escritos:

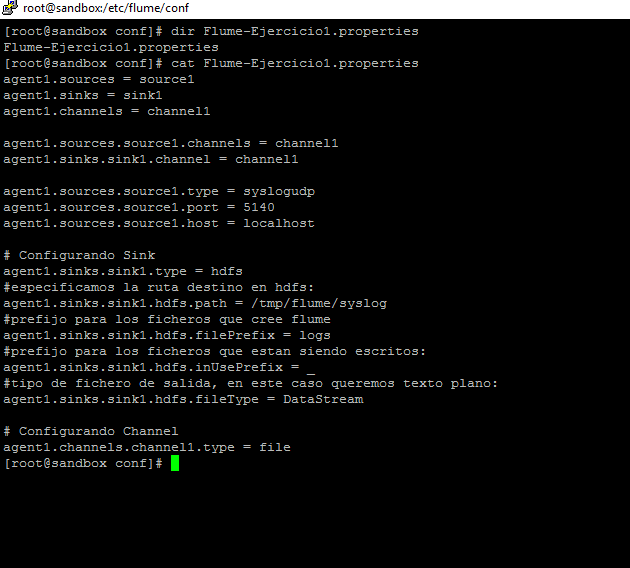
agent1.sinks.sink1.hdfs.inUsePrefix = \_

#tipo de fichero de salida, en este caso queremos texto plano:

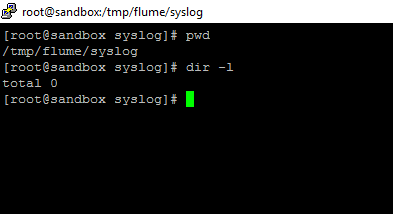
agent1.sinks.sink1.hdfs.fileType = DataStream

# Configurando Channel

agent1.channels.channel1.type = file

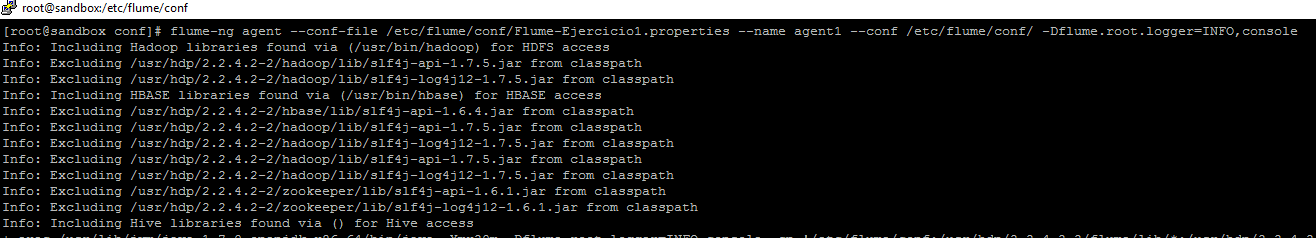


Creación de directorio /tmp/flume/syslog y contenido del mismo



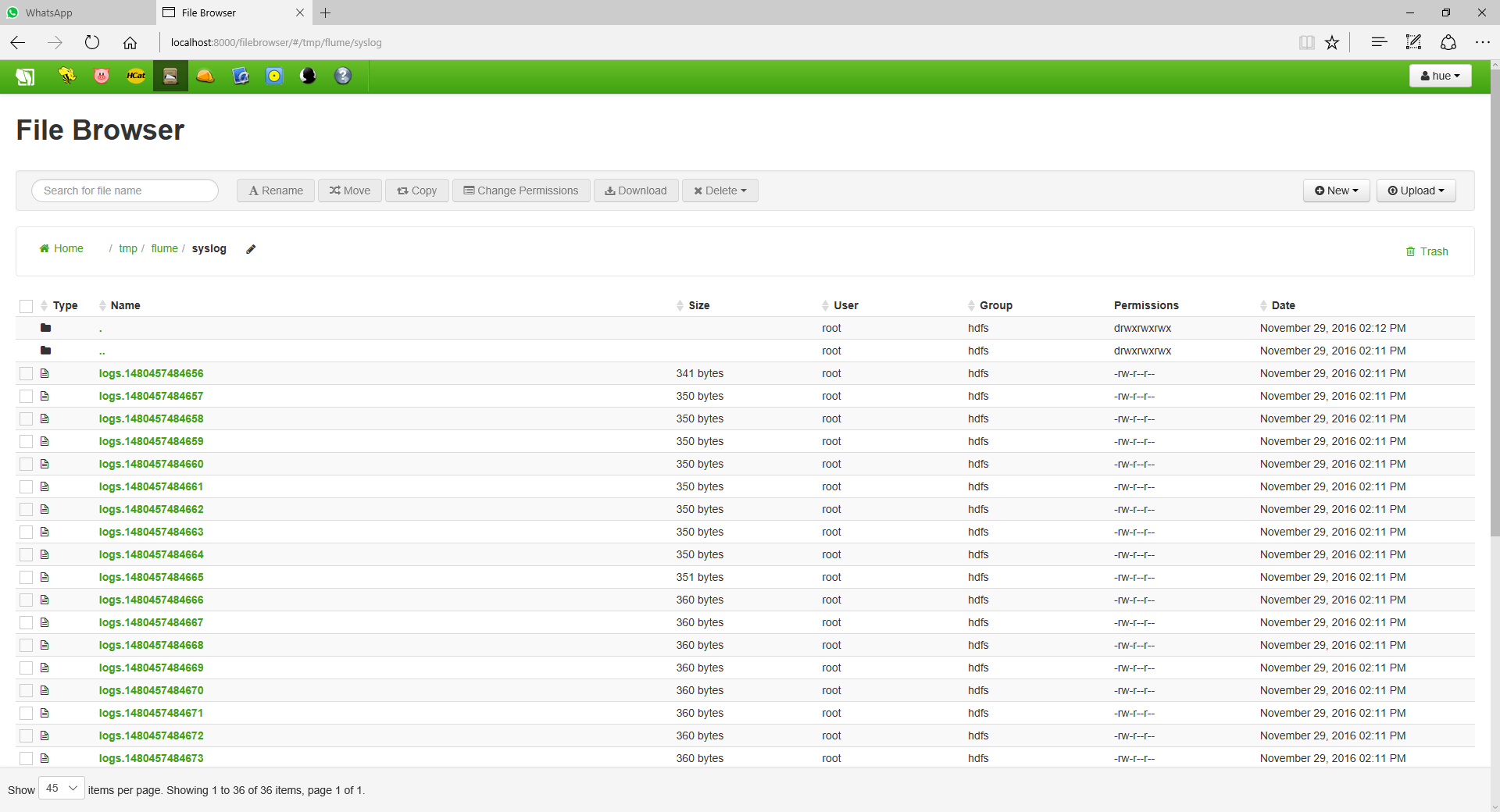
Parte 2:

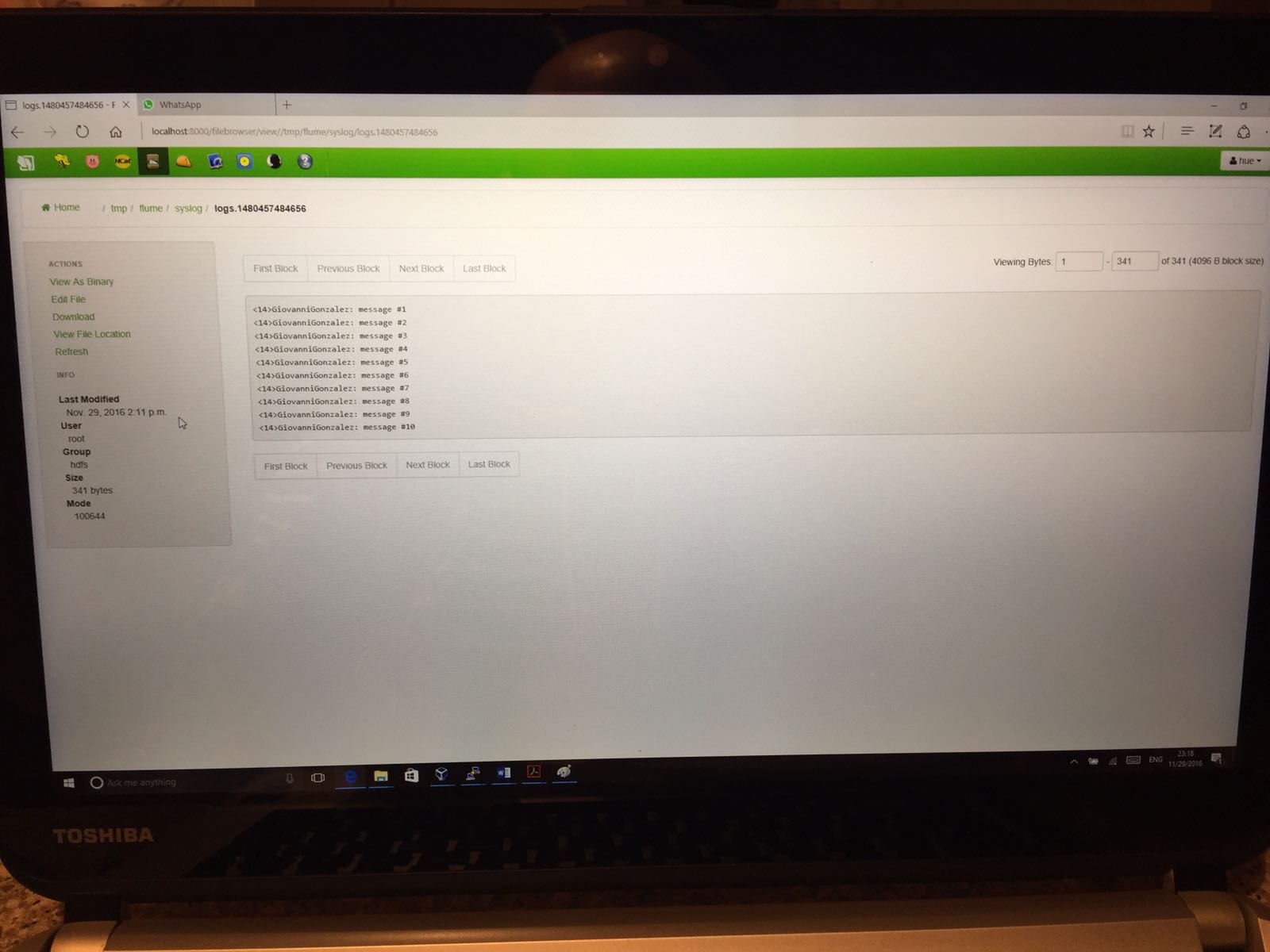
flume-ng agent --conf-file /etc/flume/conf/Flume-Ejercicio1.properties --name agent1 --conf /etc/flume/conf/ -Dflume.root.logger=INFO,console

****

Parte 3

python syslog\_generator.py GiovanniGonzalez





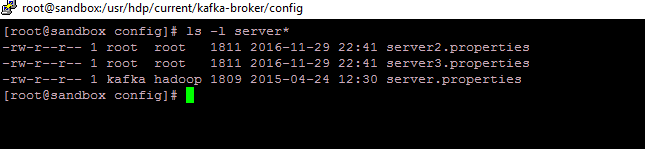
# Ejercicio 2 (Kafka)

**Parte 1**

En el directorio /usr/hdp/current/kafka-broker/config he creado 2 archivos adicionales al existentes, se modificaron los campos:

* broker.id
* port
* log.dirs

por lo que tengo los siguientes archivos:



Procedemos a iniciar los brokers:

cd /usr/hdp/current/kafka-broker/

bin/kafka-server-start.sh config/server.properties

bin/kafka-server-start.sh config/server2.properties

bin/kafka-server-start.sh config/server3.properties

Parte 2

Creacion del topic

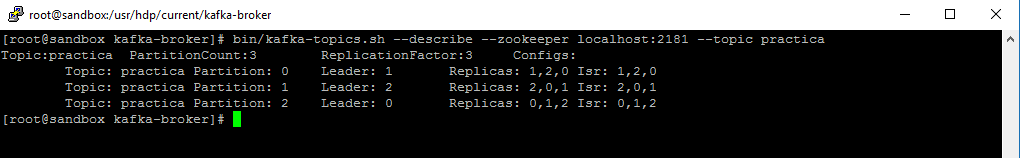
cd /usr/hdp/current/kafka-broker/

*bin/kafka-topics.sh --create --zookeeper localhost:2181 --topic practica --partitions 3 --replication-factor 3*

Parte 3

Validamos que tenga 3 particiones

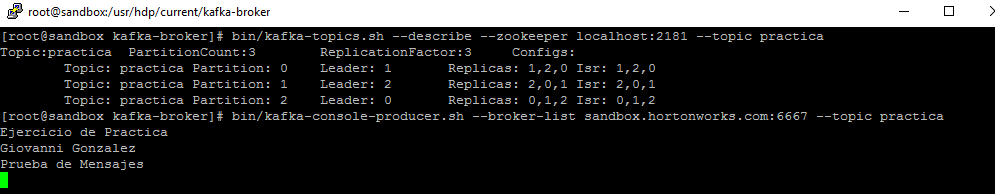
bin/kafka-topics.sh --describe --zookeeper localhost:2181 --topic practica



Parte 4

Insertamos mensajes.

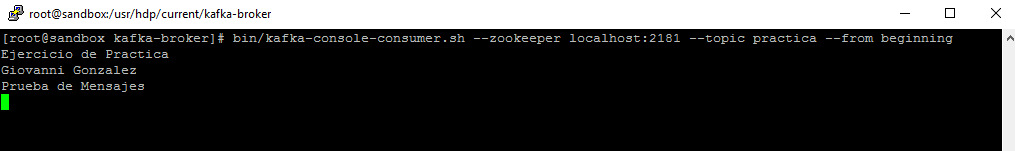
*bin/kafka-console-producer.sh --broker-list sandbox.hortonworks.com:6667 --topic practica*



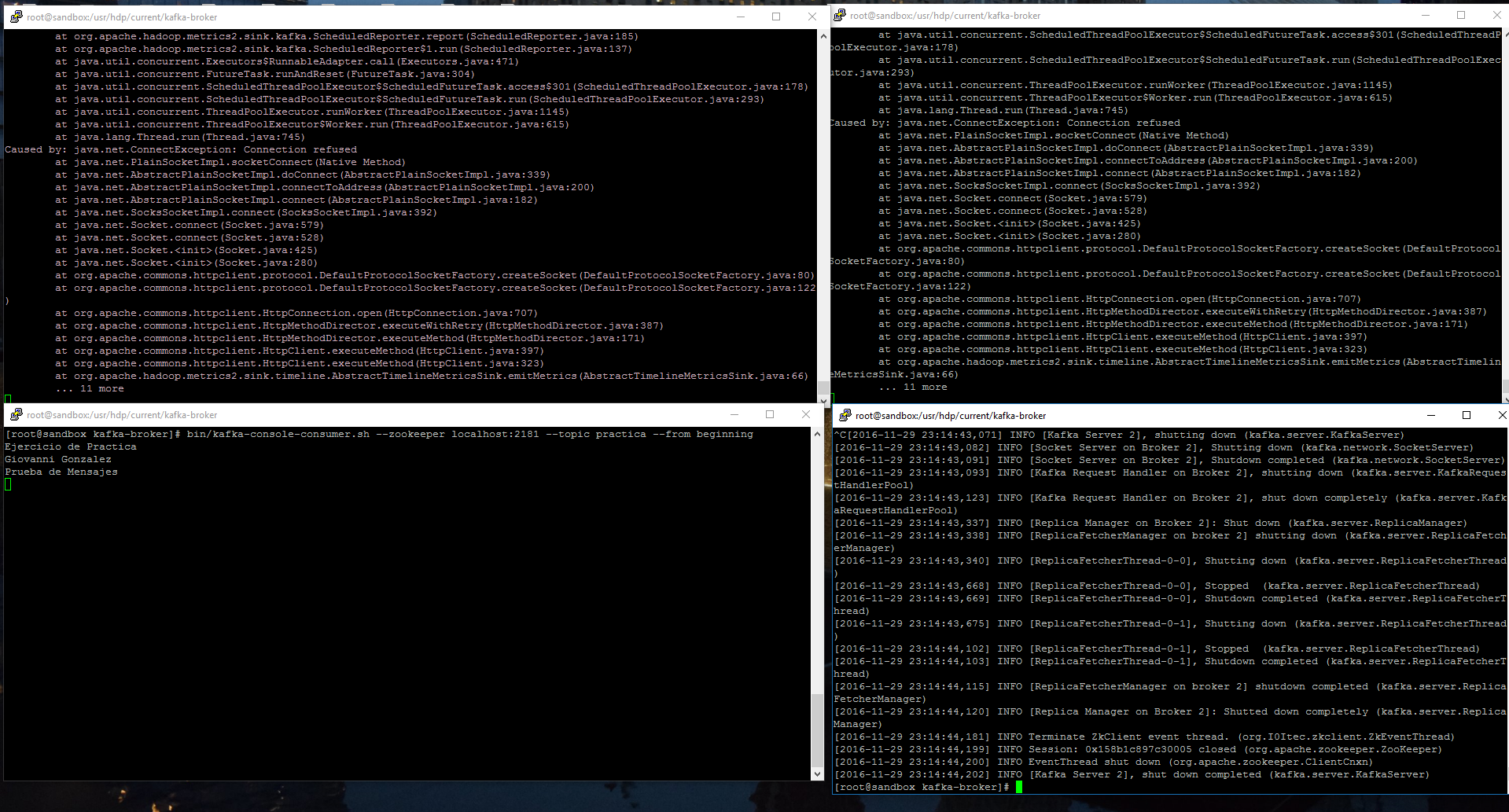
A partir de esta parte, consumimos mensajes y se van matando brokers.

Parte 5

*bin/kafka-console-consumer.sh --zookeeper localhost:2181 --topic practica --from beginning*

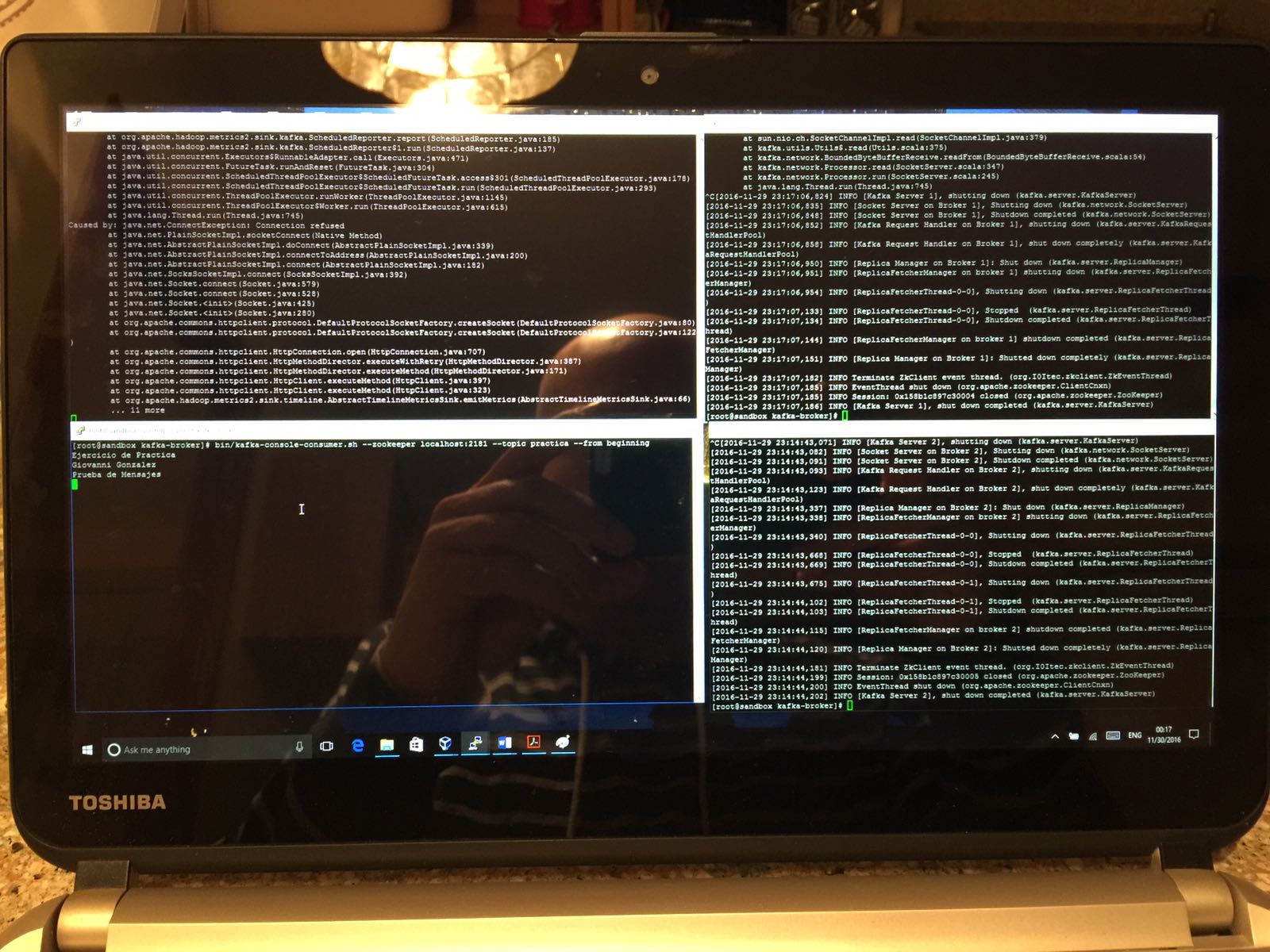


Parte 6 y 7



Parte 8 y 9

Se observa que se mataron 2 brokers y aun sigue consumiendo los mensajes.



# Ejercicio 3 (sqoop)

Comando Utilizado

sqoop import --connect jdbc:mysql://localhost/employees --table employees --where "hire\_date < '1990-01-01'" --fields-terminated-by "\t" --warehouse-dir /sqoop/veteran\_employees

