**Apache Cassandra**

*Part 1: Setting up Cassandra*

Cassandra needs to be configured on Two VM’s. Hence there are some things needed to be checked before doing so. Make sure that you are using the later version of Java & python using **java –version & python –version** respectively.

Then, execute the following commands on both the virtual machines for installing Cassandra.

**echo "deb http://www.apache.org/dist/cassandra/debian 36x main" | sudo tee -a /etc/apt/sources.list.d/cassandra.sources.list**

**curl https://www.apache.org/dist/cassandra/KEYS | sudo apt-key add**

**sudo apt-get update**

**sudo apt-get install cassandra**

**nano /etc/cassandra/cassandra.yml**

Then edit these lines in Master

seeds: “master, slave”

listen\_address: master

rpc\_address: 0.0.0.0.

broadcast\_rpc\_address: master

Then edit these lines in Slave

seeds: “master, slave”

listen\_address: slave

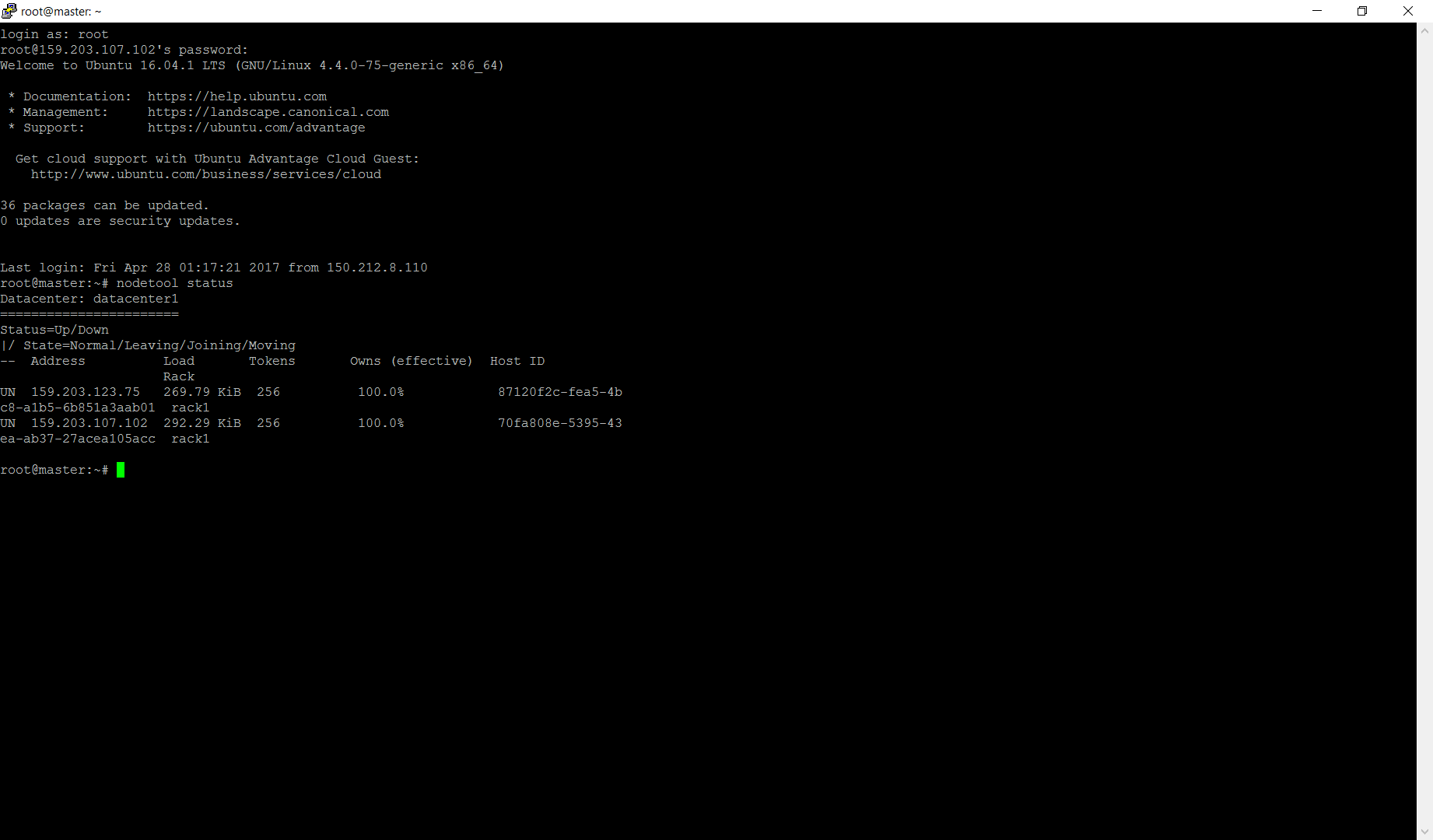
rpc\_address: 0.0.0.0.

broadcast\_rpc\_address: slave

Execute the following commands after everything above is done

service cassandra stop

cassandra –Rf



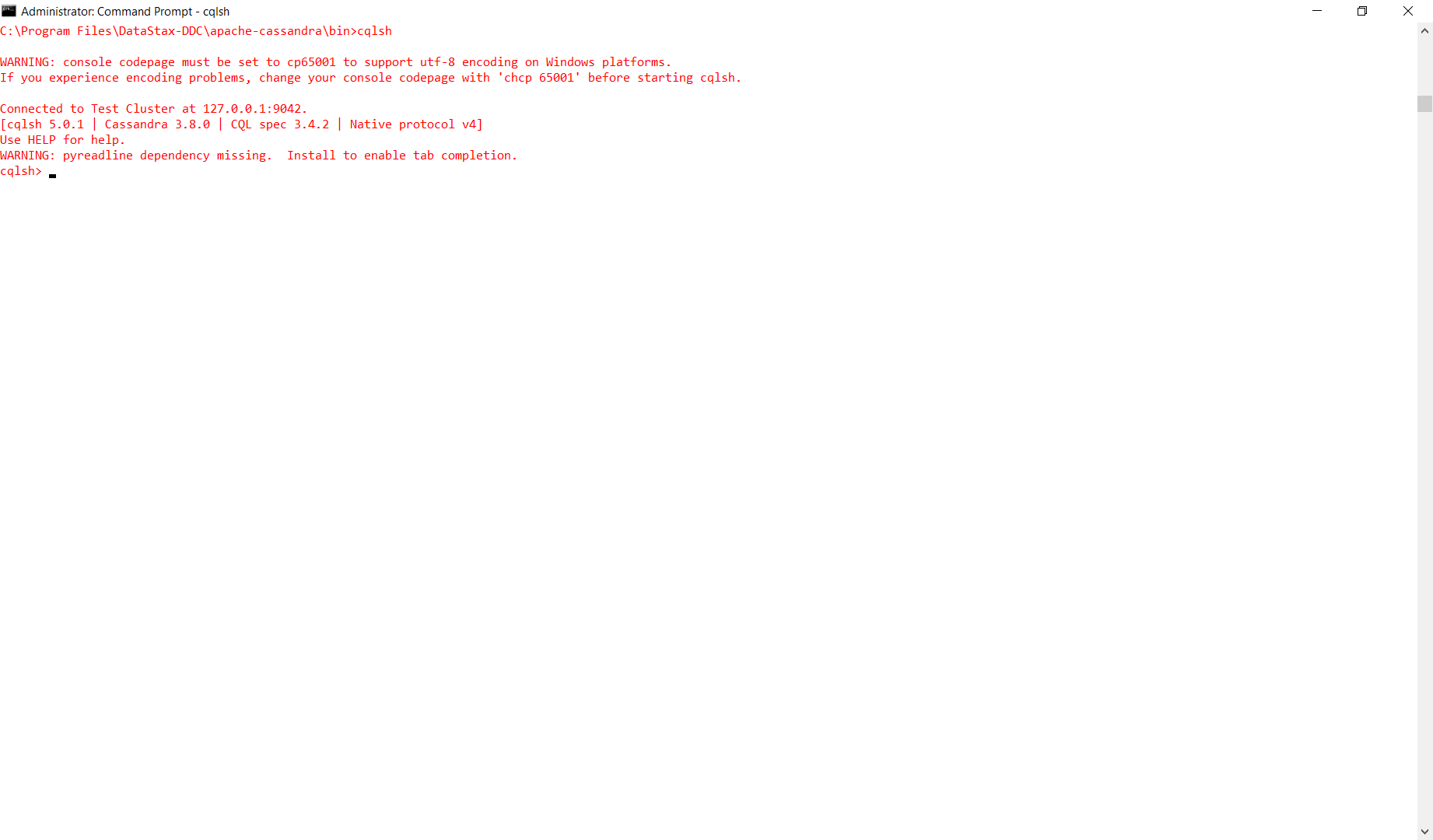
Check the status using nodetool status & then start the CQL client!

On facing memory issues, one can try running Cassandra on windows by executing following steps

1) Download & Install DataStax Community.

2) Then open the command prompt & change it to the corresponding directory. Try running cqlsh <ip address> <port number>

3) A new cql shell will be opened.



*PART2: Import Data into Cassandra*

Atleast one keyspace and table needs to be created for performing data operations.

Use the following steps to import data into Table from a csv file.

CREATE KEYSPACE access WITH

strategy\_class = 'SimpleStrategy'

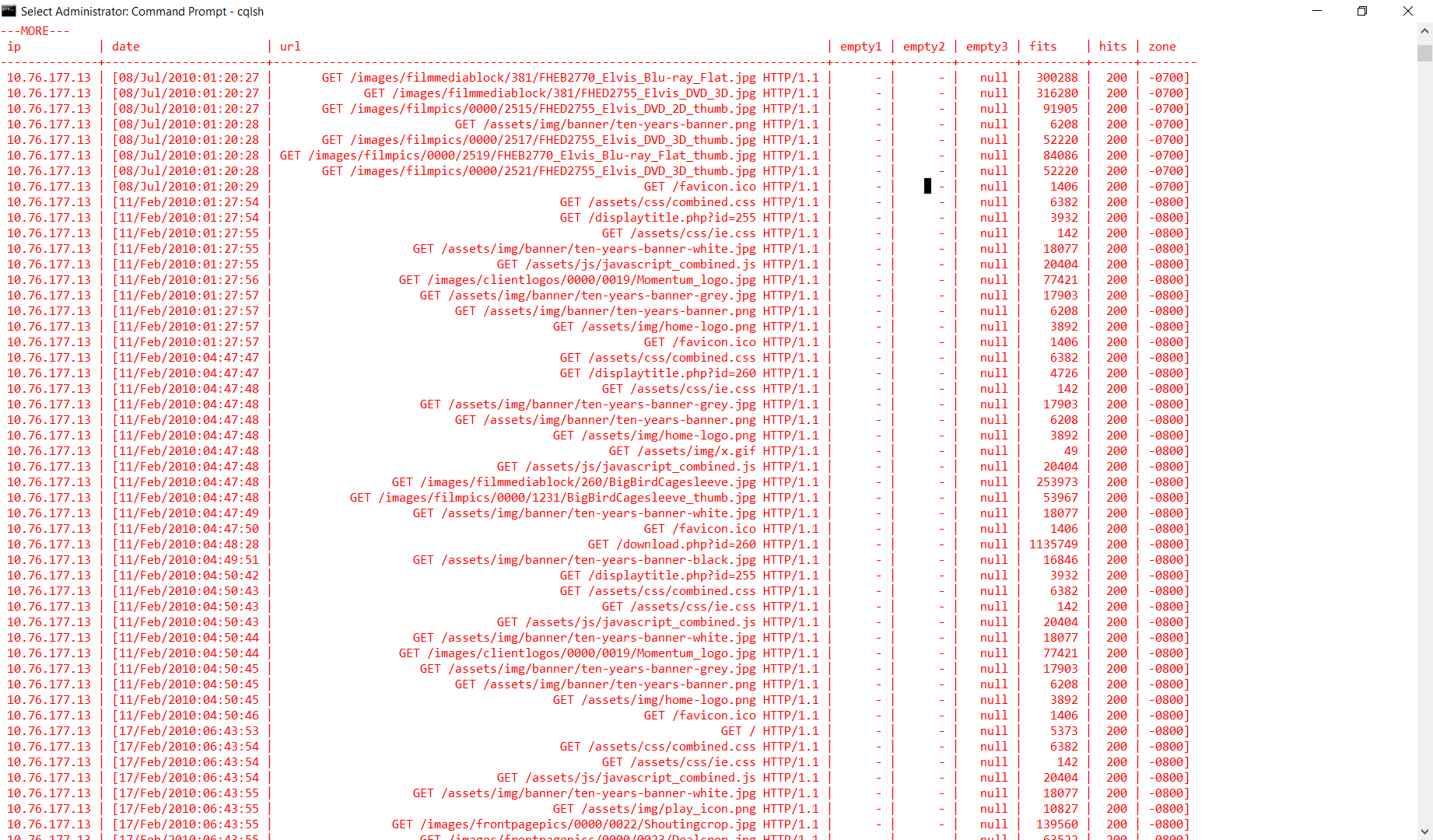
AND strategy\_options:replication\_factor = '3';

CREATE TABLE access.logs(ip TEXT, empty1 TEXT, empty2 TEXT, date TEXT, zone TEXT, url TEXT, hits TEXT, fits TEXT, PRIMARY KEY(ip, date, url));

COPY access.logs(ip, empty1, empty2, date, zone, url, hits, fits) FROM 'C:/Users/aswat/Desktop/access\_log.csv' WITH DELIMITER = ' ' AND HEADER = FALSE;



Once, the data has been loaded into the CQL table, it will look like below,



*PART-3 Operate Data In Cassandra*

We are using CQL to operate on access logs & the results for following questions are calculated.

1) How many hits were made to the website item “/assets/img/release-schedule-logo.png”?

2) How many hits were made from the IP: 10.207.188.188 ?

3) Which path in the website has been hit most? How many hits were made to the path?

4) Which IP accesses the website most? How many accesses were made by it?

The corresponding screenshots & commands for the questions are below,

IP:

CREATE KEYSPACE logs WITH strategy\_class = 'SimpleStrategy' AND strategy\_options:replication\_factor = '3';

CREATE TABLE logs.info(ip TEXT, empty1 TEXT, empty2 TEXT, date TEXT, zone TEXT, url TEXT, hits TEXT, fits TEXT, empty3 TEXT PRIMARY KEY(ip, date, url));

COPY logs.info(ip, empty1, empty2, date, zone, url, hits, fits, empty3) FROM 'C:/Users/aswat/Desktop/access\_log.csv' WITH DELIMITER = ',' AND HEADER = FALSE;

SELECT COUNT (\*) FROM logs.info WHERE url = " GET /assets/img/release-schedule-logo.png HTTP/1.1";

HITS:

CREATE KEYSPACE websitehits WITH replication = {'class' : 'SimpleStrategy' , 'replication\_factor' : 3};

CREATE TABLE websitehits.access(ip TEXT, empty1 TEXT, empty2 TEXT, date TEXT, zone TEXT, url TEXT, hits TEXT, fits TEXT, empty3 TEXT PRIMARY KEY(url));

COPY websitehits.access(ip, empty1, empty2, date, zone, url, hits, fits, empty3) FROM 'C:/Users/aswat/Desktop/access\_log.csv' WITH DELIMITER = ',' AND HEADER = FALSE;

SELECT

`url`,

FROM `websitehits.access`

GROUP BY LOWER(`date`)

ORDER BY `count` DESC



