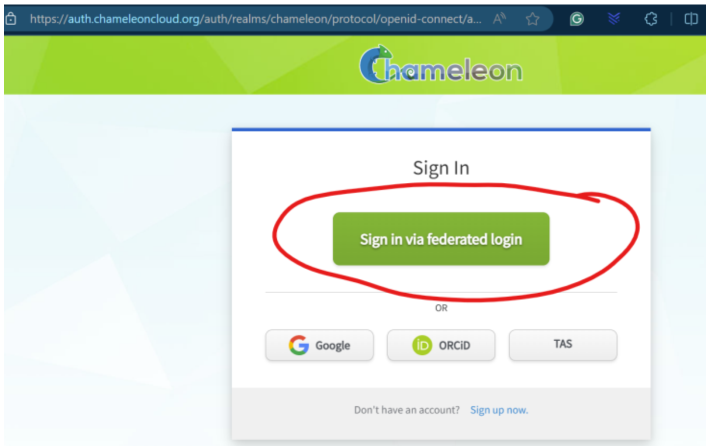
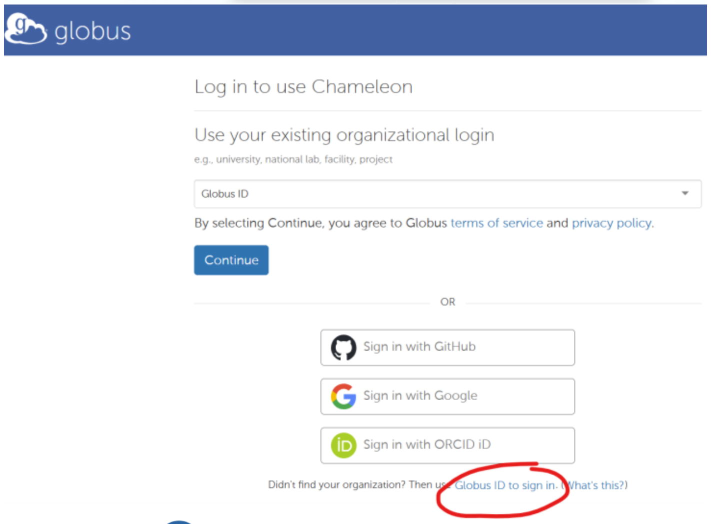
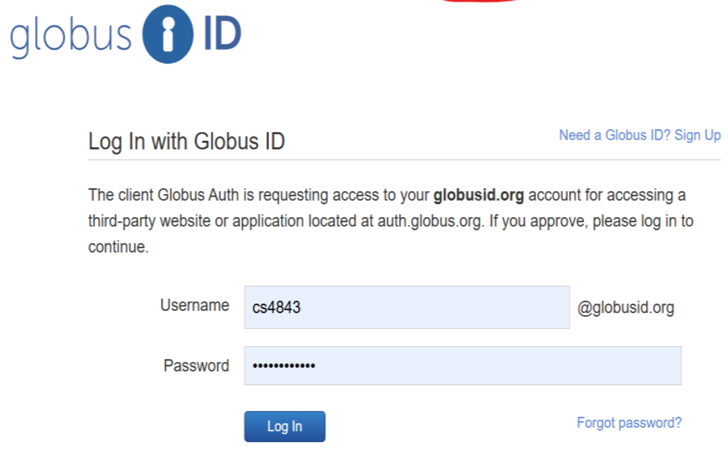
CS 4843 Cloud Computing
Assignment 1: Hadoop Cluster Setup and MapReduce Programming

Cloud Access:

- *Login to OpenStack dashboard in Chameleon Cloud at https://kvm.tacc.chameleoncloud.org*
  ***username**: cs4843 **password**: Mycscloud123*


https://auth.chameleoncloud.org/auth/realms/chameleon/protocol/openid-connect/a...
Chameleon
Sign In
Sign in via federated login
OR
Google
ORCiD
TAS
Don't have an account? Sign up now.


globus
Log in to use Chameleon
Use your existing organizational login
e.g., university, national lab, facility, project
Globus ID
By selecting Continue, you agree to Globus terms of service and privacy policy.
Continue
OR
Sign in with GitHub
Sign in with Google
Sign in with ORCID iD
Didn't find your organization? Then use Globus ID to sign in. (What's this?)



The client Globus Auth is requesting access to your **globusid.org** account for accessing a third-party website or application located at auth.globus.org. If you approve, please log in to continue.

Username: cs4843 @globusid.org

Password: ••••••••••••

1. Setup a Hadoop cluster consisting of **2 VMs** in the Chameleon cloud and start the Hadoop daemon processes as described in Module 5. Each student group needs to create exactly 2 VMs in total. VM name should be chosen according to the group name (e.g group1-1, group1-2, group2-1, group2-2 etc.) Make sure that you do not accidentally delete other group's VMs.

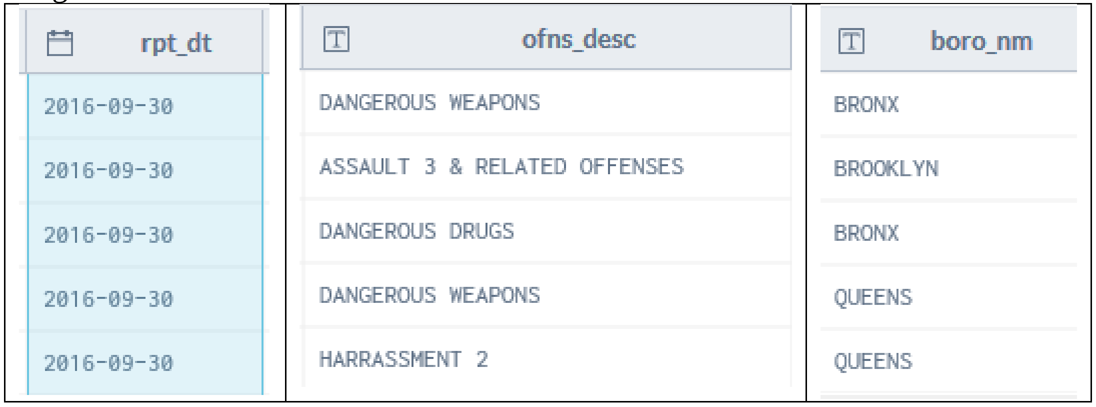
2. The port numbers used by Hadoop processes are blocked by Chameleon Cloud's strict firewall policy. You need to unblock them by running the following commands on both VMs.

```
sudo firewall-cmd --zone=public --add-port=50010/tcp
sudo firewall-cmd --zone=public --add-port=50030/tcp
sudo firewall-cmd --zone=public --add-port=50060/tcp
sudo firewall-cmd --zone=public --add-port=50070/tcp
sudo firewall-cmd --zone=public --add-port=50075/tcp
sudo firewall-cmd --zone=public --add-port=50090/tcp
sudo firewall-cmd --zone=public --add-port=54310/tcp
sudo firewall-cmd --zone=public --add-port=54311/tcp
```

3. Download the New York City Crime Data from 2016 on your master VM (groupX-1) as follows:

```
wget http://cs.utsa.edu/~plama/CS5573/NYPD_Complaint_Data_Current_YTD.csv
```

The file contains 361,740 rows and 25 columns, total size 89.48 MB. The columns relevant to this assignment are shown below:



rpt_dt	ofns_desc	boro_nm
2016-09-30	DANGEROUS WEAPONS	BRONX
2016-09-30	ASSAULT 3 & RELATED OFFENSES	BROOKLYN
2016-09-30	DANGEROUS DRUGS	BRONX
2016-09-30	DANGEROUS WEAPONS	QUEENS
2016-09-30	HARRASSMENT 2	QUEENS

4. Create an input directory in HDFS, and copy the downloaded New York City Crime data file to HDFS as follows:

```
cd $HADOOP_PREFIX
bin/hadoop fs -mkdir /hw1-input
bin/hadoop fs -put ~/NYPD_Complaint_Data_Current_YTD.csv /hw1-input/
```

Note: the environment variable $HADOOP_PREFIX can be activated by running the following command:
```
source ~/.bashrc
```

5. Write a MapReduce program in Python 3 (hw1-mapper1.py and hw1-reducer1.py) that will answer the following based on New York City Crime Data 2016. Run the program with only one reduce task.

   a) Where is most of the crime happening in New York? (e.g BRONX, QUEENS, BROOKLYN, etc.)
   b) What is the total number of crimes reported in that location ?
   c) What types of crime are happening in that location (show unique crime types) ?

   Sample Output:
   Most of the crimes were reported in XYZ.

Total number of crimes reported in XYZ is ….
Crime types reported in XYZ are ….

6. Write a MapReduce program in Python 3 (hw1-mapper2.py and hw1-reducer2.py) that will answer the following based on New York City Crime Data 2016. Run the program with two reduce tasks.

   How many crimes of type "DANGEROUS WEAPONS" were reported on each month of the year 2016 ?

   Sample Output:
   DANGEROUS WEAPONS reported per month:
   January ###
   February ###
   ..
   ..
   December ###

Reading CSV file:

The NYPD police report is a CSV file. Please note that some of the comma separated values in this file have commas embedded inside double quotes. Therefore, a simple split(",") function will incorrectly split those special values. In order to avoid this issue, you need to import and use Python's CSV module in hw1-mapper.py as follows:

```
#!/usr/bin/env python3
from csv import reader
import sys

for line in reader(sys.stdin):
     boro, crime = (line[13].strip(), line[7].strip())
     if not boro or not crime or boro == "BORO_NM":
       continue

     # rest of the code goes here …
```

Note: It is essential to include (#!/usr/bin/env python3) as the first line of code in both hw1-mapper.py and hw1-reducer.py. The mapper program must read from **sys.stdin**. DO NOT try to open the input file directly.

Program Execution:

(a) Test your python programs locally on your master VM before running it in the Hadoop cluster.

```
cat NYPD_Complaint_Data_Current_YTD.csv | python hw1-mapper.py | sort | python hw1-reducer.py
```

(b) After uploading the crime data to HDFS, run your program in Hadoop cluster as follows:

```
$HADOOP_PREFIX/bin/hadoop  jar $HADOOP_PREFIX/contrib/streaming/hadoop-streaming-*.jar \
-input /hw1-input \
-output /hw1-output \
-file /home/cc/hw1-mapper.py \
-mapper /home/cc/hw1-mapper.py \
-file /home/cc/hw1-reducer.py \
-reducer /home/cc/hw1-reducer.py
```

Troubleshooting Tips:

(a) Hadoop does not allow you to run the same program with the same output directory more than once. To run the program multiple times, you need to either delete the previously used output directory, or use a new output directory.

(b) If a job is long-running, and you want to free the shell for other activities, you can let the job run in the background by using:

```
Ctrl-C
```

(c) If a job hangs (making no progress), you can fetch the job id and kill the job as follows:

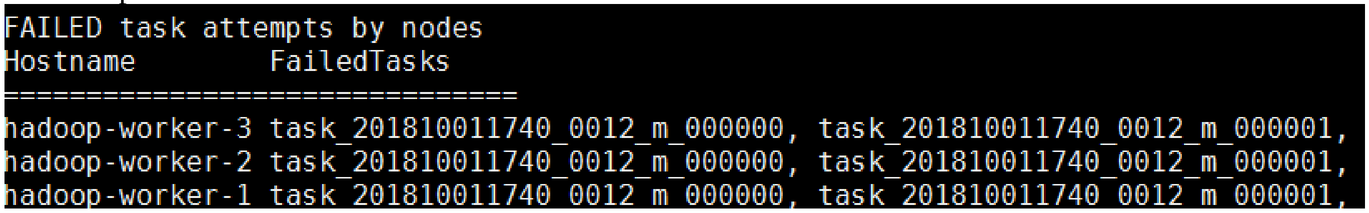
```
bin/hadoop job –list

bin/hadoop job –kill job_2014----
```

(d) To troubleshoot the task that took too long or failed, take the following steps:

- Check which tasks have failed on which worker nodes by using the following command

```
bin/hadoop job –history <output-directory>
```

for example:



```
FAILED task attempts by nodes
Hostname        FailedTasks
===============================
hadoop-worker-3 task_201810011740_0012_m_000000, task_201810011740_0012_m_000001,
hadoop-worker-2 task_201810011740_0012_m_000000, task_201810011740_0012_m_000001,
hadoop-worker-1 task_201810011740_0012_m_000000, task_201810011740_0012_m_000001,
```

- Check the corresponding “stderr” file under the directory, `/usr/local/hadoop-1.2.1/logs/userlogs`

Common Mistakes to Avoid:

(a) Make sure that the mapper and reducer code has the following line as the 1st line in the file.

```
#!/usr/bin/env python3
```

(b) Make sure that the print command uses parenthesis if you are using python3. e.g print (“hello”)

(c) If you copy the mapper and reducer code from a Windows machine to your linux VM instead of directly using the vi editor, make sure to run the following commands to fix the incompatibility issues.

```
sudo apt install dos2unix
```

```
dos2unix *.py
```

(d) The mapper program must read from sys.stdin. Do not try to open the input file directly. This is because Hadoop streaming utility automatically reads file from HDFS and pipes it to the mapper program's standard input.

Submission Policy and Deliverables

Only one submission per group is required on the Canvas. Submission must include:

1. MapReduce programs (hw1-mapper1.py, hw1-reducer1.py, hw1-mapper2.py, hw1-reducer2.py)

2. The resulting output files of your MapReduce programs should be downloaded from HDFS and submitted as a deliverable. To download output from HDFS to your VM's home directory, run the following command: **`$HADOOP_PREFIX/bin/hadoop fs -get /hw1-output/part-00000 /home/cc`**

3. A PDF report that includes:
    a. Representative Screenshots of the console output when you execute your programs.
    b. Describe how the work was divided among your group members. It is critical to accurately describe the contribution made by each member. **If a student does not contribute to the group, they will not get any points for the homework**.
4. Each student needs to submit a Self & Peer Evaluation form available on Canvas. The evaluation score will have a significant impact on the assignment grade received by individual members of the group.