IEMS 5709 Fall 2024 Homework #0

Release date: Sept 4, 2023

Due date: Sept 14, 2024 (Saturday) 11:59 pm

(Note: The course add-drop period ends on Sept 15. Late-add students will **NOT** be granted extra time extension for submission.)

No late homework will be accepted!

Every Student MUST include the following statement, together with his/her signature in the submitted homework.

I declare that the assignment submitted on Elearning system is original except for source material explicitly acknowledged, and that the same or related material has not been previously submitted for another course. I also acknowledge that I am aware of University policy and regulations on honesty in academic work, and of the disciplinary guidelines and procedures applicable to breaches of such policy and regulations, as contained in the website http://www.cuhk.edu.hk/policy/academichonesty/.

Name_____ SID_____

Date_____ Signature _____

Submission notice:

- Submit your homework via the elearning system

General homework policies:

A student may discuss the problems with others. However, the work a student turns in must be created COMPLETELY by oneself ALONE. A student may not share ANY written work or pictures, nor may one copy answers from any source other than one's own brain.

Each student **MUST LIST** on the homework paper the **name of every person he/she has discussed or worked with**. If the answer includes content from any other source, the student **MUST STATE THE SOURCE**. Failure to do so is cheating and will result in sanctions. Copying answers from someone else is cheating even if one lists their name(s) on the homework.

If there is information you need to solve a problem but the information is not stated in the problem, try to find the data somewhere. If you cannot find it, state what data you need, make a reasonable estimate of its value and justify any assumptions you make. You will be graded not only on whether your answer is correct, but also on whether you have done an intelligent analysis.

Q0 [10 marks]: Secure Virtual Machines Setup on the Cloud

In this task, you are required to set up virtual machines (VMs) on a cloud computing platform. While you are free to choose any cloud platform, Google Cloud is recommended. References [1] and [2] provide the tutorial for Google Cloud and Amazon AWS, respectively.

The default network settings in each cloud platform are *insecure*. **Your VM can be hacked by external users, resulting in resource overuse which may charge your credit card a big bill of up to $5,000 USD.** To protect your VMs from being hacked and prevent any financial losses, you should set up secure network configurations for all your VMs.

In this part, you need to set up a whitelist for your VMs. You should follow one of the following options to set up the whitelist:

(1) Only traffic from the IP of your local machine is allowed to access your VM via SSH. (In this case, you may need to update your whitelist from time to time if your local machine is not assigned a fixed public IP.)

(2) Only users in the CUHK network can access your VMs via SSH. Traffic outside CUHK should be blocked. The IP range of the CUHK network is 137.189.0.0/16. (In this case, you should first **physically** connect to the CUHK network, instead via CUHK VPN, and then be allowed to access your VMs.)

a. [**10 marks**] Secure Virtual Machine Setup
Reference [4] and [5] are the user guides for the network security configuration of AWS and Google Cloud respectively. You can go through the document with respect to the cloud platform you use. Then follow the listed steps to configure your VM's network:
   i. locate or create the security group/ firewall of your VM;
   ii. remove all rules of inbound/ ingress and outbound/ egress, except for the default rule(s) responsible for internal access within the cloud platform;
   iii. add a new rule to the inbound/ ingress, with the SSH port(s) of VMs (default: 22) and source specified, e.g., if you choose option (2) above, the source should be '137.189.0.0/16' ;
   iv. (Optional) more ports may be further permitted based on your needs (e.g., when completing Q1 below).

Q1 [90 marks + 20 bonus marks]: Hadoop Cluster Setup

Hadoop is an open-source software framework for distributed storage and processing. In this problem, you are required to set up a Hadoop cluster using the VMs you instantiated in Q0.

In order to set up a Hadoop cluster with multiple virtual machines (VM), you can first set up a single-node Hadoop cluster for each VM [6]. Then modify the configuration file in each node to set up a Hadoop cluster with multiple nodes. References [7], [9], [10], [11] provide the setup instructions for a Hadoop cluster. Some important notes/ tips on instantiating VMs are given at the end of this section.

a. [**20 marks**] Single-node Hadoop Setup

In this part, you need to set up a single-node Hadoop cluster in a pseudo-distributed mode, and run the Terasort example on your Hadoop cluster.

i. Set up a single-node Hadoop cluster (recommended **Hadoop version: 2.9.x**, all versions available in [16]). Attach the screenshot of http://localhost:50070 (or http://<VM ip>:50070 if opened in the browser of your local machine) to verify that your installation is successful.

ii. After installing a single-node Hadoop cluster, you need to run the Terasort example [8] on it. You need to record all your key steps, including your commands and output. The following commands may be useful:

```
$ ./bin/hadoop jar  \
      ./share/hadoop/mapreduce/hadoop-mapreduce-
examples-2.9.2.jar  \
      teragen 120000 terasort/input
                                  //generate the data for sorting
$ ./bin/hadoop jar  \
      ./share/hadoop/mapreduce/hadoop-mapreduce-
examples-2.9.2.jar  \
      terasort terasort/input terasort/output
                                  //terasort the generated data
$ ./bin/hadoop jar  \
      ./share/hadoop/mapreduce/hadoop-mapreduce-
examples-2.9.2.jar  \
      teravalidate terasort/output terasort/check
                                  //validate the output is sorted
```

Notes: To monitor the Hadoop service via Hadoop NameNode WebUI (http://<VM ip>:50070) on your local browser, based on steps in Q0, you may further allow traffic from the firewall whitelist to access port 50070 of VMs.

b. [**40 marks**] Multi-node Hadoop Cluster Setup

After the setup of a single-node Hadoop cluster in each VM, you can modify the configuration files in each node to set up the multi-node Hadoop cluster.

i. Install and set up a multi-node Hadoop cluster **with 4 VMs (1 Master and 3 Slaves).** Use the 'jps' command to verify all the processes are running.

ii. In this part, you need to use the 'teragen' command to generate 2 different datasets to serve as the input for the Terasort program. You should use the following two rules to determine the size of the two datasets of your own:

- Size of dataset 1: (Your student ID % 3 + 1) GB
- Size of dataset 2: (Your student ID % 20 + 10) GB

Then, run the terasort and teravalidate code again for these two different datasets and compare their running time.

Hints: Keep an image for your Hadoop cluster. You would need to use the Hadoop cluster again for subsequent homework assignments.

Notes:

1. You may need to add each VM to the whitelist of your security group/ firewall, and further permit traffic towards more ports needed by Hadoop/YARN services (reference [17] [18]).
2. For step i, the resulting cluster should consist of 1 namenode and 4 datanodes. More precisely, 1 namenode and 1 datanode would be running on the master machine, and each slave machine runs one datanode.

3. Please ensure that after the cluster setup, the number of “Live Nodes” shown on Hadoop NameNode WebUI (port 50070) is 4.

c. [**30 marks**] Running Python Code on Hadoop
Hadoop streaming is a utility that comes with the Hadoop distribution. This utility allows you to create and run MapReduce jobs with any executable or script as the mapper and/or the reducer. In this part, you need to run the Python wordcount script to handle the Shakespeare dataset [12] via Hadoop streaming.
   i. Reference [13] introduces the method to run a Python wordcount script via Hadoop streaming. You can also download the script from the reference [14].
   ii. Run the Python wordcount script and record the running time. The following command may be useful:

```
$ ./bin/hadoop jar  \
      ./share/hadoop/tools/lib/hadoop-streaming-2.9.2.jar  \
      -file mapper.py -mapper mapper.py  \
      -file reducer.py -reducer reducer.py  \
      -input input/*  \
      -output output
                        //submit a Python program via Hadoop
streaming
```

d. [**Bonus 20 marks**] Compiling the Java WordCount program for MapReduce
The Hadoop framework is written in Java. You can easily compile and submit a Java MapReduce job. In this part, you need to compile and run your own Java wordcount program to process the Shakespeare dataset [12].
   i. In order to compile the Java MapReduce program, you may need to use “hadoop classpath” command to fetch the list of all Hadoop jars. Or you can simply copy all dependency jars in a directory and use them for compilation. Reference [15] introduces the method to compile and run a Java wordcount program in the Hadoop cluster. You can also download the Java wordcount program from reference [14].
   i. Please specify the steps for configuring the environment and compiling JAVA programs. Run the Java wordcount program and compare the running time with part c.

IMPORTANT NOTES:

1. Since AWS will not provide free credits anymore, we recommend you to use Google Cloud (which offers a 90-day, $300 free trial) for this homework.
2. If you use Putty for SSH client, please download from the website https://www.putty.org/ and avoid using the default private key. Failure to do so will subject your AWS account/ Hadoop cluster to hijacking.
3. Launching instances with Ubuntu (version >= 18.04 LTS) is recommended. Hadoop version 2.9.x is recommended. Older versions of Hadoop may have vulnerabilities that can be exploited by hackers to launch DoS attacks.
4. (AWS) For each VM, you are recommended to use the t2.large instance type with 100GB hard disk, which consists of 2 CPU cores and 8GB RAM.

5. (Google) For each VM, you are recommended to use the n2-standard-2 instance type with 100GB hard disk, which consists of 2 CPU cores and 8GB RAM.
6. When following the given references, you may need to modify the commands according to your own environment, e.g., file location, etc.
7. After installing a single-node Hadoop, you can save the system image and launch multiple copies of the VM with that image. This can simplify your process of installing the single-node Hadoop cluster on each VM.
8. Keep an image for your Hadoop cluster. You will need to use the Hadoop cluster again for subsequent homework assignments.
9. Always refer to the logs for debugging single/multi-node Hadoop setup, which contains more details than CLI outputs.

Submission Requirements:

1. Include all the key steps/ commands, your cluster configuration details, source codes of your programs, etc., together with screenshots, into a SINGLE PDF report.
2. Package all the source codes (as you included in step 1) into a zip file individually.

References:

1. Google Compute Engine Tutorial: https://cloud.google.com/compute/docs/quickstart
2. AWS Tutorial: https://aws.amazon.com/getting-started
3. ~~CUHK VPN user guide: https://www.itsc.cuhk.edu.hk/all-it/wifi-and-network/cuhk-vpn/~~
4. User guide of Amazon EC2 security groups for Linux instances: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html
5. User guide of Google Cloud firewall rules: https://cloud.google.com/vpc/docs/firewalls
6. Single-Node Hadoop setup: https://hadoop.apache.org/docs/r2.9.2/hadoop-project-dist/hadoop-common/SingleCluster.html
7. Multi-node Hadoop cluster setup: https://hadoop.apache.org/docs/r2.9.2/hadoop-project-dist/hadoop-common/ClusterSetup.html
8. Terasort example: https://hadoop.apache.org/docs/r2.9.2/api/org/apache/hadoop/examples/terasort/package-summary.html
9. Hadoop multi-node cluster setup in Google cloud platform (Video): https://www.youtube.com/watch?v=OxEtUAmb2SQ&list=PLYEGL9-7r3BPfwlMFt1lARZrjmqu0OGpF
10. Hadoop multi-node cluster setup in AWS (Video): https://www.youtube.com/watch?v=XkDhf1pwPtA&list=PLWsYJ2ygHmWjPsg-6MnQO6WxVWFl8OzK_
11. Multi-node Hadoop cluster setup: https://www.edureka.co/blog/setting-up-a-multi-node-cluster-in-hadoop-2.X
    backup URL for the above link: https://emulationsofttech.wordpress.com/2018/04/06/setting-up-a-multi-node-cluster-in-hadoop-2-x/
12. Shakespeare dataset https://mobitec.ie.cuhk.edu.hk/iems5709Fall2024/static_files/shakespeare.zip
13. Writing a Hadoop MapReduce program in python

http://www.michael-noll.com/tutorials/writing-an-hadoop-mapreduce-program-in-python/

14. MapReduce wordcount program
    https://www.dropbox.com/s/kdhlzkcajq1g5h1/MapReduce_WordCount.zip?dl=0
15. Compile and run Java MapReduce program
    https://hadoop.apache.org/docs/stable/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html
16. The archive of different versions of Hadoop
    https://archive.apache.org/dist/hadoop/core/
17. HDFS Service Ports
    https://docs.cloudera.com/HDPDocuments/HDP2/HDP-2.6.5/bk_reference/content/hdfs-ports.html
18. YARN service ports
    https://docs.cloudera.com/HDPDocuments/HDP3/HDP-3.1.0/administration/content/yarn-ports.html