IERG4330/ESTR4316/IEMS5730 Spring 2022
Homework #1 (K8s)

Release date: Jan 24, 2021

Due date: Feb 13, 2021 (Sun) 11:59pm.

The solution will be posted soon after the deadline. 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 the Blackboard 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/.

Signed (Student_____) Date:_____

Name_____ SID_____

Submission notice:

- Submit your homework via the blackboard system
- Only the following students are required to submit this assignment:
  - Students who HAVE taken IERG4300/ESTR4300/IEMS5709
  - IERG4330/ ESTR431 students who have been granted the prerequisite exemption.

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.

Q1[100 marks + 10 bonus]: Hadoop over Kubernetes

Kubernetes, also known as k8s, is an open-source system for automating deployment, scaling, and management of containerized applications. This is also known as the enhanced version of Borg which was developed at Google to manage both batch-processing and stream-processing applications (i.e., a resource manager for mainstream distributed computing systems).

In this question, you need to set up a Kubernetes cluster and run Hadoop MapReduce programs on top of it. **Please keep an image for your Kubernetes cluster**. You would need to use it again for subsequent homework assignments.

a) [**20 marks]** Single-node Kubernetes Cluster Setup

In this part, you need to set up a single-node Kubernetes cluster and get familiar with "kubectl"[3], where "kubectl" is the command-line tool to let you control a Kubernetes cluster.

i) Launch an instance in AWS EC2/ Google Cloud and install "docker", "kubeadm", " kubelet", and "kubectl" following [4][5].
ii) Then create a single-node Kubernetes cluster with "kubeadm"[6].
iii) Run the hello-world example [7] in your cluster. You can open the Hello World application in a browser and take a screenshot to verify your success.

Tips:

1. Launching instances with Ubuntu 18.04 LTS is recommended
2. For the VM, you are recommended to use the t2.large instance type, which consists of 2 CPU cores and 8GB RAM each.
3. For Windows and macOS, Docker Desktop already includes a standalone Kubernetes server and client [8]. You can use it to finish part (a). However, this kind of single-node cluster is difficult to extend to multi-node.
4. For students in Mainland, you may need a VPN when downloading the source of "kubeadm" and hello-world docker.
5. The YAML file used in the official guide [7] may not be working properly since some cloud providers do not support the LoadBalancer very well. In that case, we recommend applying the yaml file in [16] instead. You can then visit your hello-world example by accessing http://<instance-ip>:30123 in your browser.

b) **[40 marks]** Multi-node Kubernetes Cluster Setup

In this part, you need to setup a multi-node Kubernetes cluster with 4VMs (1 Master and 3 Slaves on AWS/Google Cloud) and deploy a Hadoop cluster over Kubernetes on **Session Mode**. Where a Hadoop cluster is executed as a long-running

Kubernetes Deployment. You can run multiple MapReduce jobs simultaneously on a Hadoop session cluster.

i) In order to setup a Kubernetes cluster with multiple virtual machines (VM), you need to install “docker”, “kubeadm”, “ kubelet”, and “kubectl” for each VM as part (a)_i.

ii) Use “kubeadm init -args” to set up a single-node Kubernetes cluster as the mast node. For the remaining three VMs, use “kubeadm join -args” to join the single-node cluster as slave nodes[6]. Check the status of your cluster via “kubectl get node”.

iii) Deploy a Hadoop cluster on top of Kubernetes. We have already provided a YAML file (i.e., hadoop.yaml [10]) based on [9], you can directly use the following command line to deploy a Hadoop session cluster over Kubernetes.

```
$ kubectl create -f hadoop.yaml
```

iv) Run Hadoop Terasort on your Kubernetes cluster[9]. Use the ‘Teragen’ command to generate 2 different data-sets of size 2GB and 20GB to serve as input for the Terasort program. Then, run the Terasort code again for these different datasets and record their running time.

c) **[20 marks] Serverless** Kubernetes Service

In this part, you need to use Amazon EKS (Elastic Kubernetes Service) [11]/ Google GKE (Google Kubernetes Engine) [12] to create a serverless Kubernetes cluster. It removes the need to provision and manage VMs (servers) so that you only pay for the resources required to run your Kubernetes pods.

i) On your local PC, install “AWS CLI”, “eksctl” and “kubectl” [14][15].

ii) Create a serverless Kubernetes cluster in Amazon EKS via the following command line:

```
$ eksctl create cluster \
  --name my-cluster \
  --node-type t2.large \
  --nodes 3 \
  --managed \
  --timeout=999h
```

iii) Deploy a Hadoop cluster on top of your serverless Kubernetes cluster and run Hadoop Terasort in the same setting as part b.

Tips:

1. You would better launch a Ubuntu virtual machine with GUI on your own PC (Windows/ macOS) to finish part (c).

d) **[20 marks]** YARN vs. Multi-node Kubernetes vs. Serverless Kubernetes

Compare the performance of Hadoop over YARN, multi-node Kubernetes cluster, and serverless Kubernetes cluster based on the Terasort example. **You can use your result of Hadoop Terasort on YARN in the homework of IERG4300.** Describe and explain your observations. Please consider the data-locality issue, the traffic between the pods in the same server/ across multiple servers.

e) **[10 marks bonus]** Fault-tolerance in YARN and Kubernetes

By building fault tolerance mechanisms into every layer of a distributed computing system, users do not need to pay much attention to the complexity of detection and recovery from hardware faults. In this part, you need to figure out the fault-tolerance mechanism in YARN and Kubernetes.

i) When running a hello world example over Kubernetes, try to suddenly kill a hello-world pod via the command line “kubectl delete pods hello-world-1”. Describe and explain your observations. You should use “kubectl get pods” to check the status of each pod with timestamps.

ii) When running a 20GB Terasort program over Kubernetes, suddenly kill a mapper/yarn-node pod/data-node pod. Describe and explain your observation. Please check the status of each pod with timestamps, record the locality of each mapper/ reducer, and figure out the fault-tolerance mechanism from logs.

Reference

[1] Google Compute Engine Tutorial: https://cloud.google.com/compute/docs/quickstart

[2] AWS Tutorial: https://aws.amazon.com/getting-started

[3] Introduction of kubectl: https://kubernetes.io/docs/reference/kubectl/overview/

[4] Docker: https://docs.docker.com/engine/install/ubuntu/

[5] Installing kubeadm, kubelet, and kubectl

https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/install-kubeadm/

[6] Create a cluster with kubeadm

https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/create-cluster-kubeadm/

[7] Kubernetes hello world example

https://kubernetes.io/docs/tutorials/stateless-application/expose-external-ip-address/

[8] Docker for Windows

https://docs.docker.com/docker-for-windows/kubernetes/

[9] Hadoop over Kubernetes

https://programmersought.com/article/1177348559/

[10] hadoop.yaml

https://mobitec.ie.cuhk.edu.hk/ierg4330/static_files/assignments/hadoop.yaml

[11] Amazon EKS: https://aws.amazon.com/en/eks/

[12] Google GKE: https://cloud.google.com/kubernetes-engine

[13] AWS Fargate https://aws.amazon.com/fargate/

[14] AWS eksctl
https://docs.aws.amazon.com/eks/latest/userguide/getting-started-eksctl.html

[15] AWS CLI

https://docs.aws.amazon.com/eks/latest/userguide/getting-started-console.html

[16] hello-world-demo.yaml

https://mobitec.ie.cuhk.edu.hk/ierg4330/static_files/assignments/hadoop.yaml