**CS 4287-5287: Principles of Cloud Computing**

**Programming Assignment #2, Fall 2021**

**On-ground: Handed out 9/28/2021; Due 10/19/2021 (11:59 pm central time in Brightspace)**

**2U: Handed out 9/29/2021; Due 10/20/2021 (11:59 pm central time in Brightspace)**

# **Theme: DevOps Technologies for Provisioning Automation of a Simple Data Analytics Pipeline in Cloud-based Virtual Machines**

If anything in this writeup seems ambiguous/not clear, please let me know and I will fix it.

You will repeat exactly what we did in Assignment #1 but this time everything will be driven through a combination of Vagrant and Ansible. So, you will write the appropriate Vagrantfile, which will spawn a local VM on your laptop (say VM1.1 but VM1.2 and VM1.3 can still be manually created and reused from Assignment 1). VM1.1 in turn will automatically run a variety of Ansible playbooks from a master playbook (all of which you will design) to spawn the VM2 and VM3 in Chameleon cloud (and likewise in AWS or GCP – your choice). The process should also deploy the application code and start the applications. Our aim is to achieve 100% automation. See how much you are able to.

For Chameleon, we will use the clouds.yaml file to store our Openstack CLI credentials. For AWS/GCP, please find the equivalent approach to do so. Reuse your modified producer and consumer code that you developed in Assignment 1 as is for this assignment.

**For Chameleon, I strongly recommend that all teams use the same floating IPs that they used in the previous assignment. For this assignment, you will have to tear down existing cloud VMs of Assignment 1 and start fresh using the automated process. However, wait to tear down your manually created cloud VMs and do it after the Milestone 1 submission. On AWS/GCP, because our credits get deducted if you keep the VM running, please take appropriate actions of teardown (maybe take snapshot and bring things up from snapshot).**

**Scaffolding Code Help**

Please see Github Scaffolding code (<https://github.com/asgokhale/CloudComputingCourse>) that you have “git cloned” and look under folders VagrantOnly (for vagrantfile sample), AnsibleOnly\_Local\_and\_Cloud (for individual ansible playbooks and a demo master playbook), and AnsibleVagrant\_Combo (for combining vagrant and ansible). Moreover, we will go over hands-on in class.

**Why these technologies?**

* Vagrant and Ansible are representative DevOps technologies and are widely used. Several other technologies exist like Chef, Puppet, Terraform, AWS CloudFormation etc but it will be hard to learn all these technologies and hence we choose some representative sample among them.

**Data collection**

As you are doing this assignment, although I have provided scaffolding code and slides to help you get started, there will still be enough learning curve, and trial/error. Please keep track of the effort in the learning curve and trial/error to get the scripts right, but then check how seamless is it to move between clouds and how much automation did you achieve.

Please understand that the learning curve to learn a new technology cannot be avoided but the hope is that the benefits will accrue over time as more projects use such DevOps technologies to generate Infrastructure-as-Code.

**Rubrics**

* Automation and Correctness (program works): 45%
* README file explaining how to run the code: 5%

(i.e., vagrant up, and then what else needs to be done to get everything working)

* Teamwork (who did what and how much): 20%
* Report on effort expended (learning curve, etc): 15%
* Zoom-demo/video to TA (or grading team): 15%

**Submission:**

All the deliverables needed for this solution including Vagrantfile, Ansible playbooks, README, effort and teamwork documentation and any other information you have should be compressed in a zip file and uploaded to Brightspace under appropriate milestones folders.

**Milestones**

**Note:** there is a learning curve including searching Ansible documentation to get things right and hence trial/error will be involved. So please do not wait till the last minute.

Note that complete automation is the goal. As before, we will have three milestones as follows:

***Milestone 1:***

* VM1.1 (or 1.2 or 1.3, you decide) should be created via Vagrant and the others will still be manual from previous assignment.
* Do not teardown your VM2 and VM3 from assignment 1 (on Chameleon; for AWS/GCP you will probably have to in order to avoid credits getting deducted)
* All necessary packages including private key to access Cloud, Ansible installation, producer.py file, etc should all be available on Vagrant-created VM so that they are accessible to both vagrant and ansible master playbook
* Use the different playbooks from our Scaffolding code AnsibleOnly\_Local\_and\_Cloud and demo that you can run these scripts manually from the generated VM
* Upload progress made as video and documentation till that point to Brightspace milestone 1

***Milestone 2:***

* Update your Vagrantfile so that it can execute a master Ansible playbook thru the vagrantfile and then invoke vagrant provision (not vagrant up)
* This master Ansible playbook should be able to create VM2 and VM3 in the cloud (using a child playbook)
* The master Ansible playbook should then invoke additional child playbooks to install Kafka and all the underlying needed packages, copy the consumer.py file to one of the cloud VMs, as well as set the server.properties file appropriately.
* Then, the master Ansible playbook via one of its child playbooks should be able to
  + Start ZooKeeper on VM2 (or VM3, your choice)
  + Start Apache Kafka brokers on both VM2 and VM3
  + Start Consumer on VM2 (or VM3, your choice)
* Now, both or all three team members should log into their respective laptop VMs (one of which is created by Vagrant) and run the producer code on each side, all of which should be able to stream their respective unique topic data with the realistic datasets that you used in Assignment 1 to Kafka exactly like we did for Assignment #1.
* As before, Consumer should be able to receive all topic data from Kafka brokers.
* Upload progress made via video and documentation till that point to Brightspace

***Milestone 3 (final part):***

* Design additional child Ansible playbooks to install CouchDB on VM2 (or VM3) using the same configuration as we did in Assignment #1
* Once again execute vagrant provision so that the updated ansible master playbook can install and configure the CouchDB. You do not want to recreate the earlier parts so Ansible playbook should treat the previous parts from Milestone 1 and 2 as idempotent because the previous parts will already be running on the cloud (at least on the chameleon cloud). If you do not want to do this step to show idempotent operations, then everything done in milestones 1 and 2 should be terminated and the entire process automated (which should not be a big deal as it was already shown to work before). For that, Ansible supports a way in which you can skip tasks and start at a desired task.
* As before, Consumer code should be able to dump the received contents into CouchDB
* Data should be viewable in CouchDB via web console.
* All documentation needed for the assignment should be ready and submitted to Brightspace.
* **Note:** Keep a separate master playbook that is able to clean up everything after the demo is done.
* Demo the entire assignment to the TA including a final cleanup of everything.