CS4287-5287: Principles of Cloud Computing

**Fall 2017 Programming Assignment #3: Load Balancing, High Availability and Cloud Federation using Docker containers**

**Handed out:** October 30, 2017 **Due:** Nov 16, 2017 in BrightSpace

The purpose of this assignment is twofold (a) first, be able to use Docker containers, and (b) to maximally reuse the artifacts and logic developed so far, and measure the degree to which the time spent in learning different technologies has amortized over the number of assignments we have done so far. As for the technical details, this assignment is exactly like Assignment #3, however, instead of running the webserver, HAProxy and dbservers directly as processes inside the VMs as shown, they will now be running as Docker containers in the three cloud VMs. As before, all three cloud-based VMs will have floating IP addresses. HAProxy and Apache webserver are in containers on VM1 (horizon); primary DB is in container on VM2 (on horizon); and replica DB in VM3 (on Chameleon). The client is on your vagrant-created VM on your laptop. No need to put that in a container. The figure below conveys the intent.



It may be best to try using Docker Swarm to manage the Docker containers inside each of the three VMs. Docker Swarm allows easy orchestration of containers, as well as create overlay networks. Maximally automate the entire process of package installation and deployment. You will probably need to create docker files that build off of a base image and customized to the needs of your application.

You may find it easier to use custom-build Docker images from hub.docker.com for MySQL, HAProxy, Apache, etc. Just search for it in hub and determine if you can use these right away and customize. Alternately, you may build off of Ubuntu 16.04 and create your own customized Docker for the different needs of the project.

My suggestion is to first get the program to work in three containers all of which are in the same Horizon VM, and then distribute the three containers across the three VMs in the federated cloud.

As before, use the Horizon DB server as primary while the Chameleon DB server as backup replica. “Writes” i.e., updates are always made only to the primary db. We will test both the load balancing aspects of HAProxy as well as failure of primary DB and hence the failover operation.

***Submission:***

The assignment will be submitted in Brightspace. Moreover, please fill the survey form below, and demonstrate the working of your VMs to TA Travis Brummet. Submit the survey documentation.

Please post in Piazza for any doubts. This assignment will require you to learn about Docker and its integration with Ansible.

**SURVEY Questions.**

1. How much effort did you give for the assignment? (Time)
2. How many lines of code you had to write?
3. Rate the difficulty in 1-10 scale (1- easiest; 10- hardest)
4. To understand reuse and amortization, first count the total lines you reused from Assignment #3, and how many more did you write? What percent reuse did you get? Was it seamless (if not, what changes had to be made)?