Homework 5 – Automated Deployment with Ansible

ECE 592 – High Performance Cloud Services

Instructors: I. Papapanagiotou & M. Devetsikiotis

Teaching Assistant: Barry Peddycord III

Introduction

The purpose of this Homework exercise is to introduce you to the use of automation and devops tools. So far, you have learned how to provision virtual machines to host applications in the cloud. Automation tools allow you to develop a standard configuration that can be applied to machines when they are created, making it easy to stamp out new instances of a cluster easily or to make a VM operational quickly after it has been destroyed. While there are many automation tools available for different use cases, this exercise will focus on the development of an **Ansible Playbook**.

Goal

Write an ansible playbook that can be run on your laptop and configure 3 VMs in Amazon AWS as a MongoDB cluster (one primary and two secondaries). Your playbook should be written in a way such that it uses group_vars and templates to avoid using hard-coded variables in configuration files.

NOTE: Even though Docker was used in Homework 4, you should not be using Ansible to provision Docker containers in this exercise.

Deliverable

A zip file of an Ansible Playbook. The playbook must contain a README with your team member names and instructions on how to run the playbook. The playbook must:
- Must be written for Ansible 2.0 or newer
- Have an inventory with two groups, the primary and secondary groups
- Have a common role that applies to all nodes.
- Have separate roles for the master and secondary nodes.
- Use variables in group_vars and templates to create generic playbooks. You must have variables for at least 2 different secondary nodes.
- Be successful in a single run against a fresh VM

Instructions

Read the Ansible guide around creating a playbook:
https://docs.ansible.com/ansible/intro_getting_started.html

Use the grading rubric below for suggestions on the order of tasks to complete to be most successful.

Grading Rubric

You will be graded on your playbook's ability to complete the following tasks.

Task	Points
Create a playbook with a common role that writes your team member's names to the file /root/team.txt on all nodes	30
Create a role that provisions a MongoDB primary node • Creates config files • Starts the MongoDB service	20
Make your Primary Node role generic to use variables and templates instead of hard-coded config files	10
Create a role that provisions at least one MongoDB secondary node • Creates config files • Starts the MongoDB service	20
Create a generic MongoDB secondary node role that uses variables in group_vars and templates to customize each node. Must provide variables for at least two secondary nodes	20
EXTRA CREDIT: When configuration on the primary or secondary changes, a handler is used to restart the daemon.	10

If your playbook runs with errors, 20 points will be deducted from your score.