[CMSC 426] Assignment #4: DevOps

Due: End of Day, November 10th, 2024

The goal of this assignment is experiment with a few different techniques used in DevOps/Continuous Deployment. We won't implement an entire pipeline as we did in class, as that may be too time consuming, but we will use some of the popular constutuent pieces. Here are the requirements:

Requirements

1. **Prerequisite:** Ensure Docker is installed on your machine. If you plan to use Ansible, ensure it is installed as well.
2. Using Flask, write the code for a REST endpoint called `shuffle` that:
   (a) Responds to HTTP POST requests.
   (b) Accepts a JSON object containing a list of integers with the key data.
   (c) Returns a JSON object with two keys:
      - original_list: the original list of integers.
      - shuffled_list: the shuffled list of integers.
3. Write a `Dockerfile` that would build a Docker container out of the REST endpoint in the previous step. Build the container ensuring everything works up to this point. Make sure that the relevant port needed to talk to your endpoint is exposed from within the container, otherwise you won't be able to reach it.
4. Separately, write a single PyTest unit test that calls the REST endpoint, deployed on localhost, and confirms (i.e., asserts) that the order of the integers provided as input is not the same as the output. That is, that the endpoint actually shuffles the list. Make sure that the test works correctly.
5. Create a second, separate, Docker container, and `Dockerfile` that builds and runs the unit test from the previous step.
6. Write an Ansible playbook (or a Docker-Compose script) that automates the building and running of each of the two containers in order.

Security Vulnerability and GitHub Issues

As part of our code review process, if we find known software security vulnerabilities in your code, we will post them on GitHub as issues for you to address:

- After you commit your code in the repo, clicking on the "Issues" tab in the GitHub repository, you will see a list of issues created for you. Each issue will outline a specific vulnerability that needs to be fixed. *If you don't see any issues it means there were no vulnerabilities so your task is done.*

- Please fix each vulnerability (issue) or if not possible or not necessary, provide a comment in the issue listing why this is the case.
- For each issue, there will also be an embedded link to survey that you must complete. Please make sure to fill out the survey after fixing each issue.
- *Addressing each vulnerability (issue) is part of your assignment grade.*

Submission Instructions

Submission is via GitHub. To accept the assignment, visit `https://classroom.github.com/a/my05icaL` and create your personal repository. We will grade your submissions directly on GitHub. There will be no submissions on Canvas. Make sure to test your application thoroughly. Submissions that do not run or are missing key components will lose points. Make sure to address any security vulnerabilities and complete the survey (see above).