Candidate Exercise

Instructions

Phase 2 of the ASE interview process is to complete the candidate exercise outlined in this document. After you have completed the exercise and published it to github, you will then have your next interview with an F5 Mentor Team. At this interview, you should be prepared to deliver:

- A demo of your final product
- A description of your approach to the problem

This exercise can be completed with any laptop. You will have approximately one week from when you receive the assignment until your next interview.

Part 1: Create an application and run it in containers

In this exercise, you will create an application that will run in containers.

Required components

The final application should:

- Be web based
- Manipulate some data store/Database (like Redis)
- Run in containers (one container per service/function)
- Can be deployed with docker-compose
- Be published on GitHub

You can use any language to develop the application. Be as creative as you would like. If you need some additional guidelines to get started, consider creating a basic web application where:

- The web interface will allow the user to do the following
    - Search for a user by name
    - Add a user to the directory
- The data store should be used to store all the users

If you are not able to fulfill all the requirements it is not considered failure. We care more about how you approach the problem than the actual outcome. If you want to enhance the application/solution, feel free to do so. The purpose of this exercise is to see:

- How you approach the problem
- How you create and publish the application

Optional components

If you want to show off your orchestration skills, create an orchestration workflow (leveraging any automation/orchestration tools such as Ansible, Jenkins, scripts, python, etc.) that will:

1. Retrieve the Github repository of the application
2. Build the relevant containers. You can use a docker registry or not.
3. Deploy the containers
4. Test the solution is working as expected or notify of any errors

Part 2: Deploy an application via orchestration/coding

The purpose of this exercise is to show how to automatically deploy an application in multiple systems for scalability purposes. You can use any tools you want to do this exercise (scripts, python, ansible, etc.) We have published a sample application here: https://github.com/nmenant/basic-app. This repository contains a python-based web application and the relevant Dockerfile to deploy it in a container. Feel free to use any other web-based application for this exercise if you prefer.

Required components

Some requirements related to the deployment:

- You'll need to install automatically the required tools on the systems hosting the application
- You'll need to deploy automatically the application you want to leverage
- The application must run in container(s)
- No manual intervention is expected during the deployment process

If you are not able to fulfill all the requirements it is not considered failure. If you want to enhance the application/solution, feel free to do so. The purpose of this exercise is to see:

- How you approach the problem
- How you update the systems and deploy the application in an orchestrated manner

Optional components

Using Ansible to do this exercise is a plus.