Twyla Candidate Challenge: SRE

Preparation & Resources

OpenShift Origin Development Cluster

In order to complete this exercise you will need an OpenShift Origin (OKD) cluster available for use. This can be setup locally if you already have a container ready environment by following instructions on https://www.okd.io/. There is no requirement for this cluster to be deployed in a particular location.

Resources

1. https://docs.ansible.com/ansible/latest/scenario_guides/guide_kubernetes.html
2. https://docs.openshift.com/container-platform/3.10/dev_guide/dev_tutorials/openshift_pipeline.html
3. https://jenkins.io/doc/book/pipeline/getting-started/

Ansible Challenges

Challenges below needs to be completed with Ansible automation. Deliverable is a git-tarball or link to a ***private*** git repository with the final solution. You are only required to complete one of the following challenges.

Deploying the rabbit

We use RabbitMQ in our infrastructure. This needs to be deployed on an OpenShift cluster as a StatefulSet.

For this exercise, you can make use an already existing OpenShift template available at https://github.com/abn/rabbitmq-openshift. You will have to write an Ansible role/play that;

1. creates a project/namespace for the deployment
2. instantiates the template with any environment specific (assume you only have one environment) parameters that are required

The outcome of executing this Ansible role would be a RabbitMQ cluster running on the OpenShift cluster.

Deploying a pipeline

Create an Ansible role/play that would deploy a an OpenShift Pipeline for an application. You can use the example pipeline/application provided in the reference (2). The role should;

1. create a project/namespace for the deployment if one does not exist
2. ensure any assets required for the pipeline are pre-created (image streams etc)

Research Challenges

Responses to these challenges can be submitted as a markdown document within a doc directory in the git repository with your Ansible challenges. These do not require any form of code/deployments, just a write up describing a solution and making your case for it.

Authentication Gateway/Proxy

Assuming there are 2 stateless services served as separate endpoints on the same host "*foo.bar.com*". Endpoints being ***api*** and ***assets***. The host root serves a client side JS application. Propose a solution using OpenSource components where possible that meets the following requirements.

1. All requests to api endpoint are authenticated and authorised.
2. Solution should be highly available and redundant.
3. Authorisation logic should be managed by the application.
4. Authentication accounts would be managed by the client side application.

Continues Integration & Deployment

Given the following conditions for a platform composed of several micro-services, design (at a high-level) how code gets from a developer's machine to the production cluster.

1. All services are written in Python
2. All services run on a Kubernetes Distro with all services dependencies (queues, gateways etc) already deployed and ready.

3. Application development follows the tweleve-factor app methodology.
4. Promotion to production requires sign-off. No other environments require sign-offs.