A CLOUD GURU

Lesson Guide - Running a Pod from a Podman-Generated Kubernetes YAML File Using Podman

Once you have used Podman to generate a Kubernetes YAML configuration file, you will want to use this file to run a pod. In this lesson, we will examine how to run the pod in the Podman environment, using the YAML configuration file we generated with Podman. Upon completion of this lesson, you will be able to use Podman to execute the pod we defined in the YAML file.

Resources

podman-play-kube

Instructions

Let's take our pod from YAML to `nginx`!

Now that we have our `nginx` pod defined in a Kubernetes YAML file, we'd like to test it before deploying it in our Kubernetes environment. Let's see how we can use `podman play` to do this.

Let's get to it!

Commands Covered

- `podman play kube`: creates and starts the pod and containers described in the YAML file
- `podman pod`: manages pods
- `podman ps`: displays information about containers

Using Podman to Run a Pod Using a Kubernetes YAML File

Before we start, let's check for pods and containers:

```
podman pod ps
```

```
podman ps -a --pod
```

Let's check the contents of our `test-pod.yml` file:

```
more test-pod.yml
```

We see the configuration of our `test-pod` pod.

If we want to take a look at our containers and pods:

```
grep test- test-pod.yml | grep name | uniq
```

We see our `test-pod` pod, and our `test-nginx` container.

Let's run the pod from our YAML file:

```
podman play kube test-pod.yml
```

Once again, checking for pods and containers:

```
podman pod ps
```

```
podman ps -a --pod
```

We see our pod and two containers.

Testing our `nginx` Pod

Let's test our `nginx` pod.

Checking with a `curl` command:

```
curl -s http://localhost:8080
```

We can see that the default `nignx` web page is working! We can now try with our web browser, using port 8080.

Congratulations! You just stood up a working `nginx` pod using a Kubernetes YAML file and the `podman play` command!

Notes

Recording - Environment used: Cloud Playground - Medium 3 unit RHEL 8 Cloud Server

Environment Setup:

Create your Cloud Playground server and log in.

Install the `container-tools` Application Stream:

```
sudo yum -y module install container-tools
```

Create a file named `test-pod.yaml` with the contents below:

```
# Generation of Kubernetes YAML is still under development!
#
# Save the output of this file and use kubectl create -f to import
# it into Kubernetes.
#
# Created with podman-2.2.1
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: "2021-03-23T21:15:58Z"
  labels:
    app: test-pod
  name: test-pod
spec:
  containers:
  - command:
    - nginx
    - -g
    - daemon off;
    env:
    - name: PATH
      value: /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
    - name: TERM
      value: xterm
    - name: container
      value: podman
    - name: NJS_VERSION
      value: 0.5.2
    - name: PKG_RELEASE
      value: 1~buster
    - name: NGINX_VERSION
      value: 1.19.8
    - name: HOSTNAME
      value: test-pod
    image: docker.io/library/nginx:latest
    name: test-nginx
```

```
    ports:
    - containerPort: 80
      hostPort: 8080
      protocol: TCP
    resources: {}
    securityContext:
      allowPrivilegeEscalation: true
      capabilities: {}
      privileged: false
      readOnlyRootFilesystem: false
      seLinuxOptions: {}
    workingDir: /
  restartPolicy: Always
status: {}
---
metadata:
  creationTimestamp: null
spec: {}
status:
  loadBalancer: {}
```

You're ready to go!