Lesson 4 Demo 7: Create a Pod with echo hostname Command

This section will guide you to:

* Create a pod with echo hostname command

This lab has one sub-section, namely:

1. Creating a pod with echo hostname command

**Note:** If you don’t have an existing Kubernetes cluster, refer to the Demo 1.1 of Lesson 1.

**Step 1:** Creating a pod with echo hostname command

* Start the kubernetes cluster in the lab
* The first thing you need to work out is how to get the hostname. There is probably an environment variable that we could access. Let’s call that env. command to see a list of all the environment variables in a container. Here is a pod template that would show information as shown below:

*cat > hostname\_super\_simple.yaml*

*apiVersion: v1*

*kind: Pod*

*metadata:*

*name: mark-super-simple-test-pod*

*spec:*

*containers:*

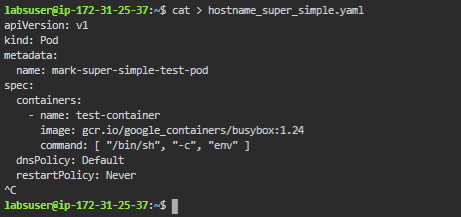
*- name: test-container*

*image: gcr.io/google\_containers/busybox:1.24*

*command: [ "/bin/sh", "-c", "env" ]*

*dnsPolicy: Default*

*restartPolicy: Never*



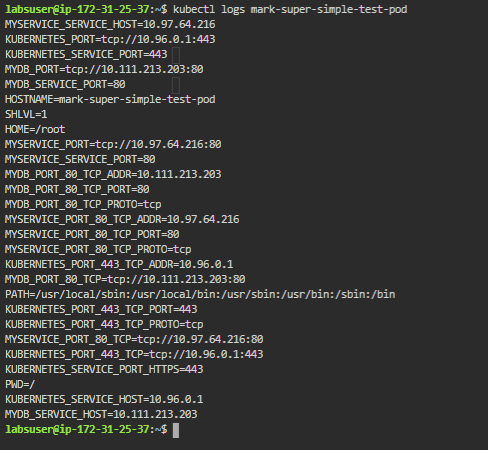
* Next, let’s create a pod from that template using kubectl as shown below:

*kubectl create -f hostname\_super\_simple.yaml*



* Now let’s check the logs of that pod, where we can get the hostname as shown below:

*kubectl logs mark-super-simple-test-pod*



* The information needed is in the $HOSTNAME. So, the next thing we need to do is create a pod template which puts that into a file as shown below:

*cat > hostname\_simple.yaml*

*apiVersion: v1*

*kind: Pod*

*metadata:*

*name: mark-test-pod*

*spec:*

*containers:*

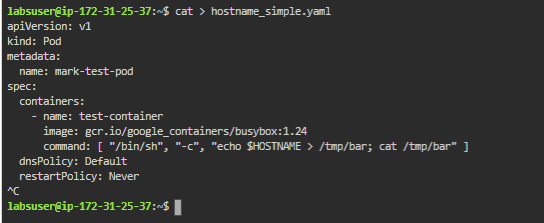
*- name: test-container*

*image: gcr.io/google\_containers/busybox:1.24*

*command: [ "/bin/sh", "-c", "echo $HOSTNAME > /tmp/bar; cat /tmp/bar" ]*

*dnsPolicy: Default*

*restartPolicy: Never*



* Now, we can create a pod using the template by running the following command:

*kubectl create -f hostname\_simple.yaml*

****

* Check the logs of the instance to see if the created pod is running as expected

*kubectl logs mark-test-pod*

