LAB 5

Create Kubernetes Deployments

Lab Objectives

In this lab, we'll create a simple deployment with several replicas and a basic web server. After that, we'll deploy a series of deployments to configure a fully functional monitoring service for Kubernetes.

Lab Structure - Overview

1. Write a basic manifest that creates a deployment and a service

Lab Overview

Conventions

Lab Guide Conventions

`reboot`	Any text a student needs to enter is printed like this.
`<your.ip>`	Any time a student needs to insert their own value, the text has brackets.
	Focuses the student's attention to a particular part of an image.
File	User Interface (UI) buttons and objects are bold.
Special Font	Unusual or important words or phrases are marked with italics.

Code Blocks

Blocks of sample code are set apart from the body and marked accordingly. It is recommended that students do not copy/paste text from the lab into their files. Extra formatting is often transferred in this process and can result in failed operations.

```
# ls -l /var/www/html/index.html
-rw-rw-r--  1 root root 1872 Jun 21 09:33 /var/www/html/index.html
# date
Wed Jun 21 09:33:42 EDT 200
```

1. Create a Deployment (Pod, Service, Replicas)

Step by Step Guide

This process will take approximately 10 minutes.

Step	Action
1.	Open a terminal console (iTerm, Terminal, PowerShell, Ubuntu Bash, Git Bash, etc).
2.	In your working directory, create a file called **hello-world.yaml** with the following content: NOTE the "selector" app field is the same as the deployment metadata app field. NOTE that Kind is Service and Deployment, also, note the statically declared NodePort.

```
apiVersion: v1
kind: Service
metadata:
  labels:
    name: hello-world-deployment
  name: hello-world-deployment
spec:
  selector:
    app: hello-world
  type: NodePort
  ports:
  - name: hello-world-deployment
    protocol: TCP
    port: 80
    nodePort: 30901
---
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: hello-world-deployment
spec:
  replicas: 3
  template:
    metadata:
      labels:
        app: hello-world
    spec:
      containers:
      - name: hello-world
        image: lindison/hello-world:k8s
        ports:
        - containerPort: 80
```

3. Delete all Kubernetes objects: **kubectl delete all --all**

```
$ kubectl delete all --all
pod "hw-deployment-2133595761-0qndd" deleted
service "hw-deployment" deleted
service "kubernetes" deleted
deployment "hw-deployment" deleted
```

4. Confirm all objects are deleted: **kubectl get all**

```
$ kubectl get all
NAME             CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
svc/kubernetes   10.0.0.1     <none>        443/TCP   55s
```

5. Deploy the basic deployment: **kubectl create -f hello-world.yaml**

```
$ kubectl create -f hello-world.yaml
service "hw-deployment" created
deployment "hw-deployment" created
```

6. Show the deployments: **kubectl get deployment**

```
$ kubectl get deployment
NAME            DESIRED   CURRENT   UP-TO-DATE   AVAILABLE   AGE
hw-deployment   1         1         1            1           13s
```

7. Show information on the Kubernetes cluster: **kubectl get all**
Note all the created objects, pods, service, deployments, and replica set.
This will show all the created Kubernetes Objects. Take note of the port, in this example, the port is **30901**.

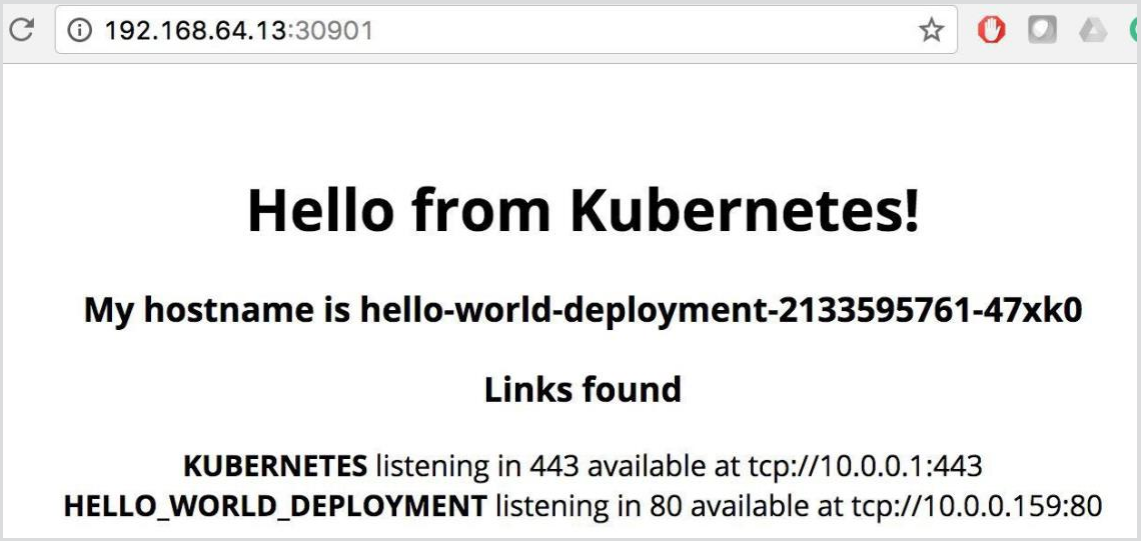
```
$ kubectl get all
NAME                                READY     STATUS    RESTARTS   AGE
po/hw-deployment-2133595761-ngl1h   1/1       Running   0          17s

NAME                 CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
svc/hw-deployment    10.0.0.84    <nodes>       80:30901/TCP     17s
svc/kubernetes       10.0.0.1     <none>        443/TCP          1m

NAME                    DESIRED   CURRENT   UP-TO-DATE   AVAILABLE   AGE
deploy/hw-deployment    1         1         1            1           17s

NAME                               DESIRED   CURRENT   READY     AGE
rs/hw-deployment-2133595761        1         1         1         17s
```

8. Using a browser, open the web page (e.g. **http://<MasterIP>:30901**)

	A page similar to the below should display. 

9.	Let's scale the hello-world application from the command line. First, get a list of pods.
	kubectl get pods
	
	`$ kubectl get pods`
	`NAME READY STATUS RESTARTS AGE`
	`hello-world-deployment-2133595761-47xk0 1/1 Running 0 hello-world-deployment-2133595761-hbzmp 1/1 Running 0 hello-world-deployment-2133595761-k5102 1/1 Running 0`

10.	Show the deployments.
	kubectl get deployments
	
	`$ kubectl get deployment`
	`NAME DESIRED CURRENT UP-TO-DATE AVAILABLE`
	`hello-world-deployment 3 3 3 3`

11.	Scale the deployment:
	kubectl scale deployment hello-world-deployment --replicas=5
	
	Note: --replicas was originally defined in the manifest from step 2.
	--replicas sets the number of replicas that'll be hosted in the Kubernetes cluster. Note, this DOES NOT automatically handle replication of state.
	
	`$ kubectl scale deployment hello-world-deployment --replicas=5`
	`deployment "hello-world-deployment" scaled`

12.	Confirm there are now 5 replicas:

	`kubectl get pods`
	
	`$ kubectl get pods`
	`NAME READY STATUS RESTARTS`
	`hello-world-deployment-2133595761-47xk0 1/1 Running 0`
	`hello-world-deployment-2133595761-49m59 1/1 Running 0`
	`hello-world-deployment-2133595761-5vq2f 1/1 Running 0`
	`hello-world-deployment-2133595761-hbzmp 1/1 Running 0`
	`hello-world-deployment-2133595761-k5102 1/1 Running 0`

13.	Run the appropriate command to scale the deployment back to its original target of 3.

14.	Show all pods: `kubectl get pods`
	
	`$ kubectl get pods`
	`NAME READY STATUS RESTARTS`
	`hello-world-deployment-2133595761-47xk0 1/1 Running 0`
	`hello-world-deployment-2133595761-hbzmp 1/1 Running 0`
	`hello-world-deployment-2133595761-k5102 1/1 Running 0`

15.	Run a set of commands that will delete all pods and get pods.
	`kubectl delete pods --all && kubectl get pods`
	
	Why do the containers keep getting created even though we deleted all pods?
	
	`$ kubectl delete pods --all && kubectl get pods`
	`pod "hello-world-deployment-2133595761-7cnvd" deleted`
	`pod "hello-world-deployment-2133595761-jwxkp" deleted`
	`pod "hello-world-deployment-2133595761-wx58j" deleted`
	`NAME READY STATUS`
	`hello-world-deployment-2133595761-5j07h 0/1 Pending`
	`hello-world-deployment-2133595761-7cnvd 1/1 Terminating`
	`hello-world-deployment-2133595761-flwst 0/1 ContainerCreating`
	`hello-world-deployment-2133595761-jwxkp 1/1 Terminating`
	`hello-world-deployment-2133595761-wx58j 1/1 Terminating`
	`hello-world-deployment-2133595761-z7qjb 0/1 ContainerCreating`

16.	Clean up the environment: `kubectl delete all --all`

Lab Complete!