Kubernetes Deployment Project

meghana avadhanam

OVERVIEW

In this mini project, I aim to deploy an nginx deployment and pod on Kubernetes, using kubernetes manifest on minikube.

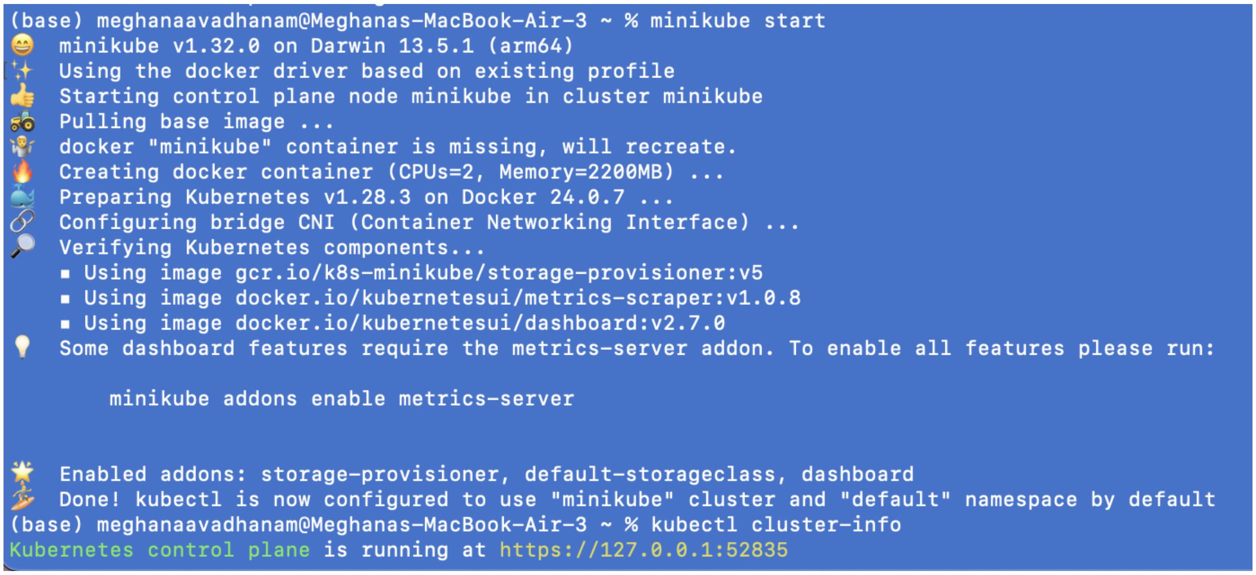
Nginx is a lightweight open-source web server that is widely used for sample pods, deployments and ingresses. It serves as a reverse proxy and load balancer, making it very useful for Kuberenetes applications.

PREREQUISITES

Minikube, Docker, kubectl

STEPS

1. `% minikube start`



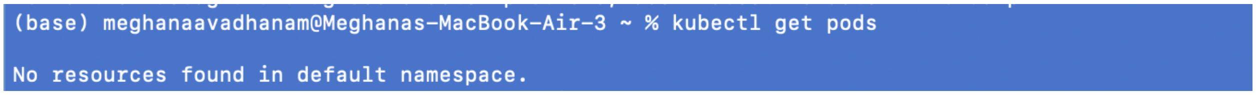
```
(base) meghanaavadhanam@Meghanas-MacBook-Air-3 ~ % minikube start
😄  minikube v1.32.0 on Darwin 13.5.1 (arm64)
✨  Using the docker driver based on existing profile
👍  Starting control plane node minikube in cluster minikube
🚜  Pulling base image ...
🤷  docker "minikube" container is missing, will recreate.
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🐳  Preparing Kubernetes v1.28.3 on Docker 24.0.7 ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
    ▪ Using image docker.io/kubernetesui/metrics-scraper:v1.0.8
    ▪ Using image docker.io/kubernetesui/dashboard:v2.7.0
💡  Some dashboard features require the metrics-server addon. To enable all features please run:

        minikube addons enable metrics-server


🌟  Enabled addons: storage-provisioner, default-storageclass, dashboard
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
(base) meghanaavadhanam@Meghanas-MacBook-Air-3 ~ % kubectl cluster-info
Kubernetes control plane is running at https://127.0.0.1:52835
```

2. `% kubectl get pods`

`No pods present in the cluster initially`



```
(base) meghanaavadhanam@Meghanas-MacBook-Air-3 ~ % kubectl get pods

No resources found in default namespace.
```

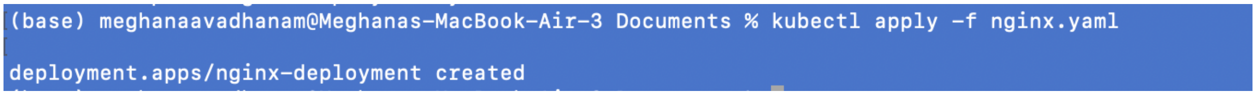
Notes : A deployment is an object that manages a set of identical pods.

3. **Creating a deployment with a ‘Kubernetes Manifest’ yaml file.**

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:latest
        ports:
        - containerPort: 80
```

4. **Apply the manifest using kubectl**

   ```
   % kubectl apply -f nginx.yaml
   ```

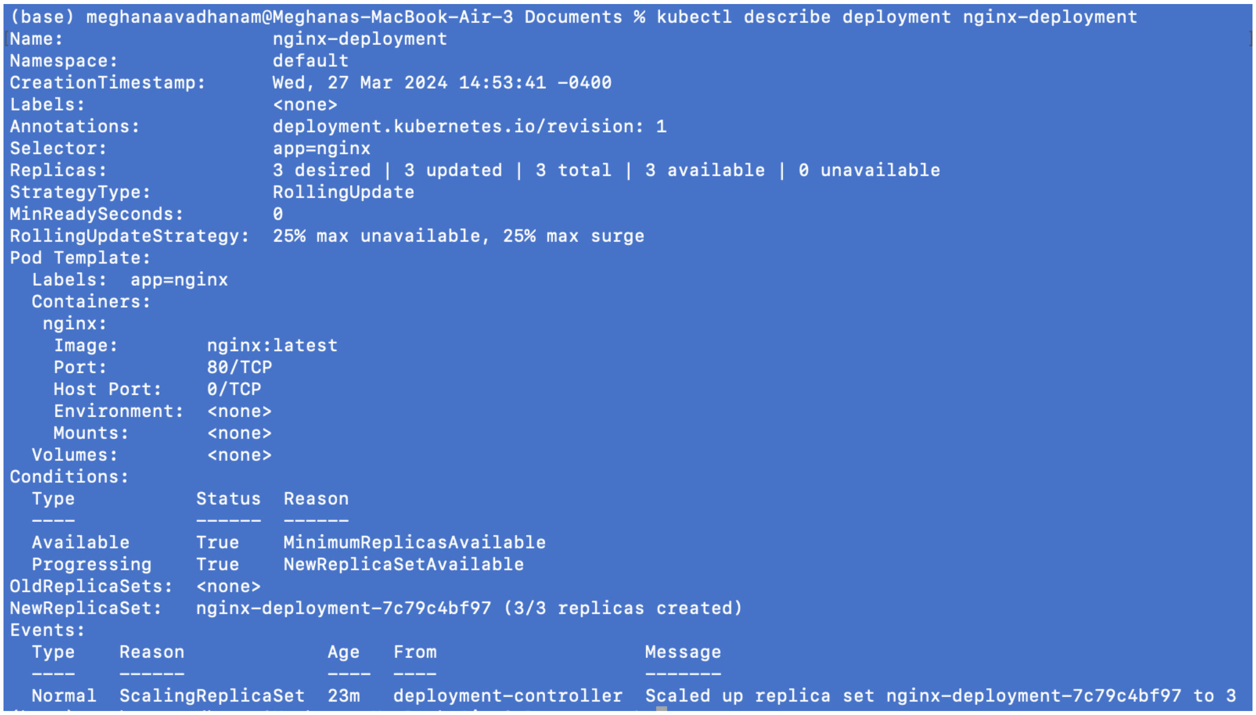


```
[(base) meghanaavadhanam@Meghanas-MacBook-Air-3 Documents % kubectl apply -f nginx.yaml
[
deployment.apps/nginx-deployment created
```

5. **After deployment is created, check the following (optional)**

   ```
   % kubectl get deployments
   ```

   ```
   % kubectl describe deployment nginx-deployment
   ```



```
(base) meghanaavadhanam@Meghanas-MacBook-Air-3 Documents % kubectl describe deployment nginx-deployment
Name:                   nginx-deployment
Namespace:              default
CreationTimestamp:      Wed, 27 Mar 2024 14:53:41 -0400
Labels:                 <none>
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               app=nginx
Replicas:               3 desired | 3 updated | 3 total | 3 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
   nginx:
    Image:        nginx:latest
    Port:         80/TCP
    Host Port:    0/TCP
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   nginx-deployment-7c79c4bf97 (3/3 replicas created)
Events:
  Type    Reason             Age   From                   Message
  ----    ------             ----  ----                   -------
  Normal  ScalingReplicaSet  23m   deployment-controller  Scaled up replica set nginx-deployment-7c79c4bf97 to 3
```

6. **Scale the Deployment up or down**

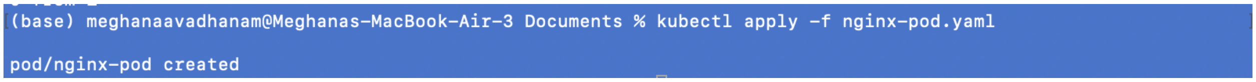
   ```
   % kubectl scale deployment <deployment-name>
   --replicas=<replica-count>
   ```

7. **Task 2 - Performing a Rolling Update on nginx deployment**

   ```
   apiVersion: apps/v1
   kind: Deployment
   metadata:
     name: nginx-deployment
   spec:
     replicas: 3
     selector:
       matchLabels:
         app: nginx
     template:
       metadata:
         labels:
           app: nginx
       spec:
         containers:
         - name: nginx
           image: nginx:1.19.6   # Updated image version
           ports:
           - containerPort: 80
   ```

Notes: A Pod is the smallest deployable unit in Kubernetes, consisting of one or more containers that share networking and storage resources.

8. **Run the pod**



```
(base) meghanaavadhanam@Meghanas-MacBook-Air-3 Documents % kubectl apply -f nginx-pod.yaml

pod/nginx-pod created
```

9. `kubectl get pods`: View all Pods in the cluster.
   `kubectl describe pod <pod-name>`: Get detailed information about a specific Pod.
   `kubectl logs <pod-name>`: View logs from a Pod's containers.
   `kubectl exec -it <pod-name> -- <command>`: Execute a command inside a running Pod.