

Name: Shweta Singh

Sap Id: 500098159

RollNo: R2142211484

Branch & Batch: Btech CSE(DevOps) & B4

Lab Exercise 9– Creating Replicaset in Kubernetes

Below is a lab exercise that will help you understand and practice creating a Replicaset in Kubernetes:

Step 1: Create a ReplicaSet Configuration File

Create a file named replicaset.yaml with the following configuration:

Link of file: (Copy following code from my GitHub repo)

<https://github.com/hkshitesh/ACO-LAB-2021-25/blob/main/scripts/replicaset.yaml>

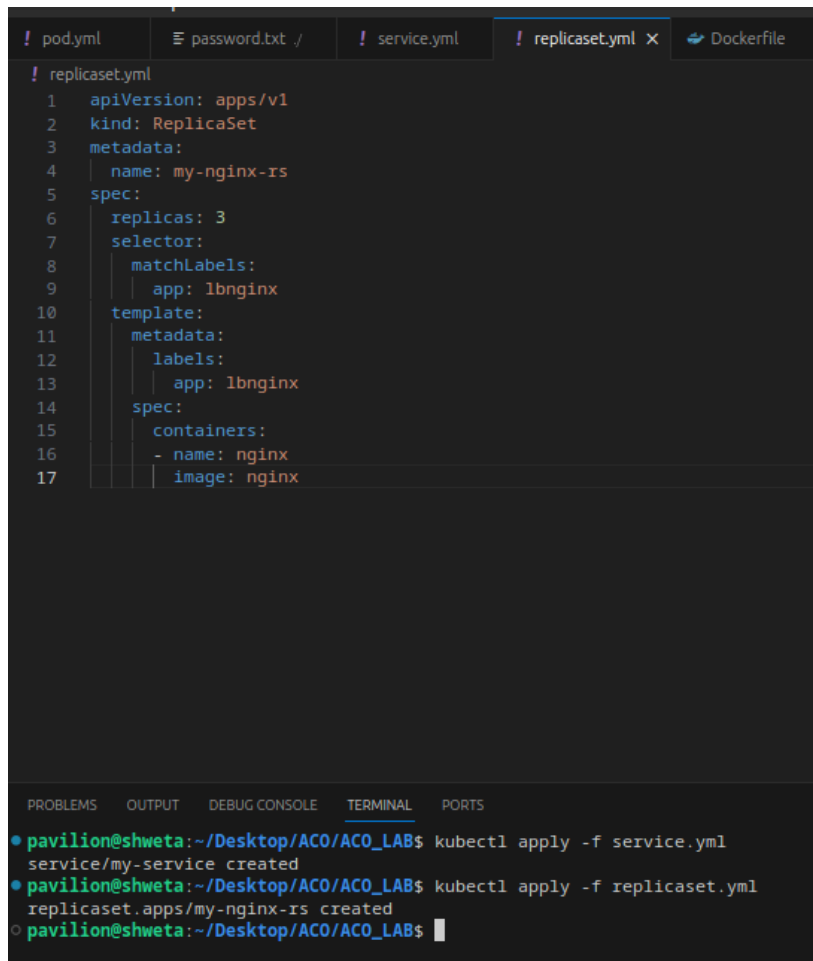
```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: my-nginx-rs
spec:
  replicas: 3
  selector:
    matchLabels:
      app: lbnginx
  template:
    metadata:
      labels:
        app: lbnginx
```

```
spec:

  containers:

  - name: nginx

    image: nginx
```



The screenshot shows a code editor with a dark theme. The top bar has tabs for 'pod.yml', 'password.txt /', 'service.yml', 'replicaset.yml', and 'Dockerfile'. The 'replicaset.yml' tab is active, showing a YAML configuration for a ReplicaSet. The configuration includes apiVersion, kind, metadata (name: my-nginx-rs), and spec (replicas: 3, selector: matchLabels: app: lbnginx, template: metadata: labels: app: lbnginx, spec: containers: - name: nginx, image: nginx). Below the editor is a terminal window with the following output:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
• pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl apply -f service.yml
service/my-service created
• pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl apply -f replicaset.yml
replicaset.apps/my-nginx-rs created
○ pavilion@shweta:~/Desktop/ACO/ACO_LAB$
```

Step 2: Apply the ReplicaSet Configuration

Apply the configuration to create the ReplicaSet:

```
kubectl apply -f replicaset.yaml
```

Step 3: View the ReplicaSet and Pods

View the created ReplicaSet and the associated Pods:

```
kubectl get replicaset
```

```
kubectl get pods
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
• pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl apply -f service.yml
service/my-service created
• pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl apply -f replicaset.yml
replicaset.apps/my-nginx-rs created
• pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl get replicaset
NAME          DESIRED  CURRENT  READY  AGE
my-nginx-rs   3        3        3      75s
○ pavilion@shweta:~/Desktop/ACO/ACO_LAB$
```

Step 4: Scale the ReplicaSet

Scale the ReplicaSet to 5 replicas:

```
kubectl scale replicaset my-nginx-rs --replicas=5
```

```
• pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl scale replicaset my-nginx-rs --replicas=5
replicaset.apps/my-nginx-rs scaled
○ pavilion@shweta:~/Desktop/ACO/ACO_LAB$
```

Step 5: Delete the ReplicaSet

Delete the ReplicaSet:

```
kubectl delete replicaset my-replicaset
```

```
• pavilion@shweta:~/Desktop/ACO/ACO_LAB$ kubectl delete replicaset my-nginx-rs
replicaset.apps "my-nginx-rs" deleted
○ pavilion@shweta:~/Desktop/ACO/ACO_LAB$
```

Conclusion

This exercise demonstrated how to create, manage, and update a ReplicaSet in Kubernetes. You learned how to scale the ReplicaSet, update the image, and delete the ReplicaSet from the cluster. Experiment further with different configurations and scaling options to deepen your understanding of managing ReplicaSets in Kubernetes.