Using K8s Persistent Volumes

Relevant Documentation

- Persistent Volumes

Lesson Reference

Create a StorageClass that supports volume expansion.

```
vi localdisk-sc.yml
```

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: localdisk
provisioner: kubernetes.io/no-provisioner
allowVolumeExpansion: true
```

```
kubectl create -f localdisk-sc.yml
```

Create a PersistentVolume.

```
vi my-pv.yml
```

```
kind: PersistentVolume
apiVersion: v1
metadata:
  name: my-pv
spec:
  storageClassName: localdisk
  persistentVolumeReclaimPolicy: Recycle
  capacity:
    storage: 1Gi
  accessModes:
    - ReadWriteOnce
  hostPath:
    path: /var/output
```

```
kubectl create -f my-pv.yml
```

Check the status of the PersistentVolume.

```
kubectl get pv
```

Create a PersistentVolumeClaim that will bind to the PersistentVolume.

```
vi my-pvc.yml
```

```yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
spec:
  storageClassName: localdisk
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 100Mi
```

```
kubectl create -f my-pvc.yml
```

Check the status of the PersistentVolume and PersistentVolumeClaim to verify that they have been bound.

```
kubectl get pv

kubectl get pvc
```

Create a Pod that uses the PersistentVolumeClaim.

```
vi pv-pod.yml
```

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: pv-pod
spec:
  restartPolicy: Never
  containers:
  - name: busybox
    image: busybox
    command: ['sh', '-c', 'echo Success! > /output/success.txt']
    volumeMounts:
    - name: pv-storage
      mountPath: /output
  volumes:
  - name: pv-storage
    persistentVolumeClaim:
      claimName: my-pvc
```

```
kubectl create -f pv-pod.yml
```

Expand the PersistentVolumeClaim and record the process.

```
kubectl edit pvc my-pvc --record
```

```
...

spec:
```

```
...

resources:
  requests:
    storage: 200Mi
```

Delete the Pod and the PersistentVolumeClaim.

```
kubectl delete pod pv-pod

kubectl delete pvc my-pvc
```

Check the status of the PersistentVolume to verify that it has been successfully recycled and is available again.

```
kubectl get pv
```