**Persistent Volumes**

Kubernetes persistent volumes by example

A persistent volume (PV) is a cluster-wide resource that you can use to store data in a way that it persists beyond the lifetime of a pod. The PV is not backed by locally-attached storage on a worker node but by networked storage system such as EBS or NFS or a distributed filesystem like Ceph.

In order to use a PV you need to claim it first, using a persistent volume claim (PVC). The PVC requests a PV with your desired specification (size, speed, etc.) from Kubernetes and binds it then to a pod where you can mount it as a volume. So let’s create such a PVC, asking Kubernetes for 1 GB of storage using the default storage class:

$ kubectl apply -f https://raw.githubusercontent.com/openshift-evangelists/kbe/master/specs/pv/pvc.yaml

$ kubectl get pvc

NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS AGE

myclaim Bound pvc-27fed6b6-3047-11e9-84bb-12b5519f9b58 1Gi RWO gp2-encrypted 18m

To understand how the persistency plays out, let’s create a deployment that uses above PVC to mount it as a volume into /tmp/persistent:

$ kubectl apply -f https://raw.githubusercontent.com/openshift-evangelists/kbe/master/specs/pv/deploy.yaml

Now we want to test if data in the volume actually persists. For this we find the pod managed by above deployment, exec into its main container and create a file called datain the /tmp/persistent directory (where we decided to mount the PV):

$ kubectl get po

NAME READY STATUS RESTARTS AGE

pv-deploy-69959dccb5-jhxx 1/1 Running 0 16m

$ kubectl **exec** -it pv-deploy-69959dccb5-jhxxw -- bash

bash-4.2$ touch /tmp/persistent/data

bash-4.2$ ls /tmp/persistent/

data lost+found

It’s time to destroy the pod and let the deployment launch a new pod. The expectation is that the PV is available again in the new pod and the data in /tmp/persistent is still present. Let’s check that:

$ kubectl delete po pv-deploy-69959dccb5-jhxxw

pod pv-deploy-69959dccb5-jhxxw deleted

$ kubectl get po

NAME READY STATUS RESTARTS AGE

pv-deploy-69959dccb5-kwrrv 1/1 Running 0 16m

$ kubectl **exec** -it pv-deploy-69959dccb5-kwrrv -- bash

bash-4.2$ ls /tmp/persistent/

data lost+found

And indeed, the data file and its content is still where it is expected to be.

Note that the default behavior is that even when the deployment is deleted, the PVC (and the PV) continues to exist. This storage protection feature helps avoiding data loss. Once you’re sure you don’t need the data anymore, you can go ahead and delete the PVC and with it eventually destroy the PV:

$ kubectl delete pvc myclaim

persistentvolumeclaim "myclaim" deleted

The types of PV available in your Kubernetes cluster depend on the environment (on-prem or public cloud). Check out the Stateful Kubernetes reference site if you want to learn more about this topic.