# Storage LAB

07 January 2021 12:12

**Admin = PV**
**User = PVC**

**Persistent Volume**
A PersistentVolume (PV) is a piece of storage in the cluster that has been provisioned by an administrator or dynamically provisioned using Storage Classes

Three Access modes:

- ReadWriteOnce -- the volume can be mounted as read-write by a single node
- ReadOnlyMany -- the volume can be mounted read-only by many nodes
- ReadWriteMany -- the volume can be mounted as read-write by many nodes

1) Create PV with volume on Host ( not ideal for Production)

```
[root@master ~]# cat pv.yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: pv-vol
spec:
  accessModes:
    - ReadWriteOnce
  capacity:
    storage: 1Gi
  hostPath:
    path: /tmp/data
```

# kubectl create -f pv.yaml

```
[root@master ~]# kubectl create -f pv.yaml
persistentvolume/pv-vol created
```

2) View Persistent volume
   a. Check status
   b. **Retain** is default policy

# kubectl get pv

```
[root@master ~]# kubectl get pv
NAME     CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS      CLAIM   STORAGECLASS   REASON   AGE
pv-vol   1Gi        RWO            Retain           Available                                   7s
```

3) Persistent Volume Claim

A **PersistentVolumeClaim** (PVC) is a request for storage by a **user**. It is similar to a Pod. Pods consume node resources and PVCs consume PV resources.

3.1 Create persistent volume claim

```
[root@master ~]# cat pvc.yml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: myclaim
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 500Mi
```

```
[root@master ~]# kubectl apply -f pvc.yml
persistentvolumeclaim/myclaim created
```

3.2 It claim from persistent volume which was created earlier.

# kubectl get pvc

```
[root@master ~]# kubectl get pvc
NAME      STATUS   VOLUME   CAPACITY   ACCESS MODES   STORAGECLASS   AGE
myclaim   Bound    pv-vol   1Gi        RWO                           11s
```

4) Claim as Volume

4.1 Firstly create index.html on worker nodes as we are using "hostPath"

```
[root@worker1 data]# cat /mnt/data/index.html
This is test K8S storage page
```

4.2 Launch a POD to consume PVC

```
[root@master ~]# cat pod_pvc.yaml
apiVersion: v1
kind: Pod
metadata:
  name: mypod
spec:
  containers:
    - name: myfrontend
      image: nginx
      ports:
      - containerPort: 80
        name: "http-server"
      volumeMounts:
      - mountPath: "/usr/share/nginx/html"
        name: mypd
  volumes:
    - name: mypd
      persistentVolumeClaim:
        claimName: myclaim
```

PVC created in earlier step

```
[root@master ~]# kubectl get pods
NAME                              READY   STATUS    RESTARTS   AGE
mypod                             1/1     Running   0          3m4s
nginx                             1/1     Running   0          107m
nginx-deploy-598b589c46-frrgv     1/1     Running   0          2d15h
```

4.3 Access pod and run apt update, apt install curl command.

# kubectl exec -it mypod -- /bin/bash

```
[root@master ~]# kubectl exec -it mypod -- /bin/bash
root@mypod:/# apt update
Get:1 http://security.debian.org/debian-security buster/updates InRelease [65.4 kB]
Get:2 http://deb.debian.org/debian buster InRelease [121 kB]
Get:3 http://deb.debian.org/debian buster-updates InRelease [51.9 kB]
Get:4 http://security.debian.org/debian-security buster/updates/main amd64 Packages
Get:5 http://deb.debian.org/debian buster/main amd64 Packages [7907 kB]
Get:6 http://deb.debian.org/debian buster-updates/main amd64 Packages [7860 B]
Fetched 8414 kB in 6s (1461 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
2 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@mypod:/# apt install curl
Reading package lists... Done
Building dependency tree
Reading state information... Done
curl is already the newest version (7.64.0-4+deb10u1).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
root@mypod:/#
```

5) From inside pod, access local host. You will get page from file which is on local node ( base machine)

```
root@mypod:/# curl http://localhost
This is test K8S storage page
root@mypod:/#
```

5.1 Delete Pod the pod now

```
[root@master ~]# kubectl delete pod mypod
pod "mypod" deleted
```

5.2 Delete PVC

5.3 Notice Reclaim Policy and status in Persistent volume status

```
[root@master ~]# kubectl delete pvc myclaim
persistentvolumeclaim "myclaim" deleted
[root@master ~]#
[root@master ~]#
[root@master ~]#
[root@master ~]# kubectl get pvc
No resources found in default namespace.
[root@master ~]#
[root@master ~]# kubectl get pv
NAME     CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS     CLAIM             STORAGECLASS   REASON   AGE
pv-vol   1Gi        RWO            Retain           Released   default/myclaim                           36m
```

6) Delete PV
   a. PV should be in Released status to get deleted

```
[root@master ~]# kubectl delete pv pv-vol
persistentvolume "pv-vol" deleted
[root@master ~]#
[root@master ~]#
[root@master ~]# kubectl get pv
No resources found
```