

# kubernetes中使用NFS作为共享存储

笔记本：新课程笔记

创建时间：2018/9/16 星期日 上午 11:02

更新时间：2018/10/14 星期日 下午 10:25

作者：306798658@qq.com

Persistent Volume(持久化卷)简称PV, 是一个K8S资源对象, 我们可以单独创建一个PV, 它不和Pod直接发生关系, 而是通过 Persistent Volume Claim, 简称PVC来实现动态绑定, 我们会在Pod定义里指定创建好的PVC, 然后PVC会根据Pod的要求去自动绑定合适的PV给Pod使用。

## 持久化卷下PV和PVC概念:

Persistent Volume ( PV ) 是由管理员设置的存储, 它是群集的一部分。就像节点是集群中的资源一样, PV 也是集群中的资源。PV 是 Volume 之类的卷插件, 但具有独立于使用 PV 的 Pod 的生命周期。此 API 对象包含存储实现的细节, 即 NFS、iSCSI 或特定于云供应商的存储系统。

PersistentVolumeClaim ( PVC ) 是用户存储的请求。它与 Pod 相似, Pod 消耗节点资源, PVC 消耗 PV 资源。Pod 可以请求特定级别的资源 ( CPU 和内存 )。PVC声明可以请求特定的大小和访问模式 ( 例如, 可以以读/写一次或只读多次模式挂载 )。

## 它和普通Volume的区别是什么呢?

普通Volume和使用它的Pod之间是一种静态绑定关系, 在定义Pod的文件里, 同时定义了它使用的Volume。Volume是Pod的附属品, 我们无法单独创建一个Volume, 因为它不是一个独立的K8S资源对象。

## 1 准备一台机器, 搭建NFS服务

```
yum install nfs-utils
vim /etc/exportfs
/data/k8s/ 172.7.15.0/24(sync,rw,no_root_squash)
systemctl start nfs
systemctl enable nfs
```

## 2 在node节点上测试

```
yum install nfs-utils
showmount -e 172.7.15.113
```

## 3 创建pv ( master上 )

vim mypv.yaml //内容如下

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: pv001
spec:
  capacity:
    storage: 10Gi
  accessModes:
    - ReadWriteMany
  nfs:
    path: /data/k8s/
    server: 172.7.15.113
```

```
kubectl create -f mypv.yaml
```

```
kubectl get pv
```

NAME	CAPACITY	ACCESS MODES	RECLAIM POLICY	STATUS	CLAIM	STORAGECLASS	REASON	AGE
pv001	10Gi	RWX	Retain	Available				10m

状态为Available, 这是因为它还没有绑定到任何的pvc上面, 当定义完pvc后, 就可以自动绑定了。

## 4 创建pvc ( master上 )

vim mypvc.yaml //内容如下

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: myclaim
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 8Gi
```

kubectl create -f mypvc.yaml

kubectl get pvc

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	AGE
myclaim	Bound	pv001	10Gi	RWX		2m

可以看到，pvc状态为Bound，它绑定了pv001

## 5 定义pod

vim pvpod.yaml //内容如下

```
apiVersion: v1
kind: Pod
metadata:
  name: httpd-pvpod
spec:
  containers:
    - image: httpd
      name: httpd-withpvc-pod
      imagePullPolicy: Always
      volumeMounts:
        - mountPath: "/usr/local/apache2/htdocs/"
          name: httpd-volume
  volumes:
    - name: httpd-volume
      persistentVolumeClaim:
        claimName: myclaim
```

kubectl create -f pvpod.yaml

kubectl describe pod httpd-pvpod //查看Volumes那部分里的ClaimName

## 6 验证

1) 到NFS的共享目录下创建一个文件

cd /data/k8s/

echo "Test file" > 1.html

2) 进入到httpd-pod里

kubectl exec -it httpd-pvpod bash

cat /usr/local/apache2/htdocs/1.html

3) 删除httpd-pvpod

kubectl delete pod httpd-pod

cat /data/k8s/1.html

4) 重建httpd-pod

kubectl create -f pvpod.yaml

5) curl访问

```
kubectrl get pod httpd-pvpod -o wide //查看其对应的IP
```

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED NODE
httpd-pod	1/1	Running	0	2m	172.20.3.5	172.7.15.114	<none>

```
curl 172.20.3.5/1.html
```

参考 <http://www.showerlee.com/archives/2280>