kubernetes pv pvc与nfs 测试

1. 准备存储(NFS)

```
$ sudo yum install nfs-utils

设置 NFS 服务开机启动
$ sudo systemctl enable rpcbind
$ sudo systemctl enable nfs

启动 NFS 服务
$ sudo systemctl start rpcbind
$ sudo systemctl start nfs

服务启动之后，我们在服务端配置一个共享目录
 mkdir  -p /data/volumes/linuxea-{1,2,3,4,5}

[root@kube-node1 volumes]# cat /etc/exports
/data/volumes/linuxea-1 10.0.0.0/8(rw,no_root_squash)
/data/volumes/linuxea-2 10.0.0.0/8(rw,no_root_squash)
/data/volumes/linuxea-3 10.0.0.0/8(rw,no_root_squash)
/data/volumes/linuxea-4 10.0.0.0/8(rw,no_root_squash)
/data/volumes/linuxea-5 10.0.0.0/8(rw,no_root_squash)


[root@kube-node1 volumes]#   exportfs -arv
exporting 10.0.0.0/8:/data/volumes/linuxea-5
exporting 10.0.0.0/8:/data/volumes/linuxea-4
exporting 10.0.0.0/8:/data/volumes/linuxea-3
exporting 10.0.0.0/8:/data/volumes/linuxea-2
exporting 10.0.0.0/8:/data/volumes/linuxea-1


[root@kube-node1 volumes]# showmount -e
Export list for kube-node1:
/data/volumes/linuxea-5 10.0.0.0/8
/data/volumes/linuxea-4 10.0.0.0/8
/data/volumes/linuxea-3 10.0.0.0/8
/data/volumes/linuxea-2 10.0.0.0/8
/data/volumes/linuxea-1 10.0.0.0/8
```

2. 创建pv

```
[k8s@kube-node1 ~]$ cat pv-demo.yaml
apiVersion: v1
kind: PersistentVolume
metadata:
```

```
    name: linuxea-1
    labels:
        name: v1
spec:
    nfs:
        path: /data/volumes/linuxea-1
        server: 10.0.19.152
    accessModes: ["ReadWriteMany","ReadWriteOnce"]
    capacity:
        storage: 1Gi
---
apiVersion: v1
kind: PersistentVolume
metadata:
    name: linuxea-2
    labels:
        name: v2
spec:
    nfs:
        path: /data/volumes/linuxea-2
        server: 10.0.19.152
    accessModes: ["ReadWriteMany","ReadWriteOnce"]
    capacity:
        storage: 2Gi
---
apiVersion: v1
kind: PersistentVolume
metadata:
    name: linuxea-3
    labels:
        name: v3
spec:
    nfs:
        path: /data/volumes/linuxea-3
        server: 10.0.19.152
    accessModes: ["ReadWriteMany","ReadWriteOnce"]
    capacity:
        storage: 3Gi
---
apiVersion: v1
kind: PersistentVolume
metadata:
    name: linuxea-4
    labels:
        name: v4
spec:
```

```
    nfs:
        path: /data/volumes/linuxea-4
        server: 10.0.19.152
    accessModes: ["ReadWriteMany","ReadWriteOnce"]
    capacity:
        storage: 4Gi
---
apiVersion: v1
kind: PersistentVolume
metadata:
    name: linuxea-5
    labels:
        name: v5
spec:
    nfs:
        path: /data/volumes/linuxea-5
        server: 10.0.19.152
    accessModes: ["ReadWriteMany","ReadWriteOnce"]
    capacity:
        storage: 5Gi
```

定义完成后apply启动

```
kubectl apply -f pv-demo.yaml
persistentvolume/linuxea-1 created
persistentvolume/linuxea-2 created
persistentvolume/linuxea-3 created
persistentvolume/linuxea-4 created
persistentvolume/linuxea-5 created
```

3. 创建pvc

```
[k8s@kube-node1 ~]$   cat pvc-demo.yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
    name: linuxea-pvc
    namespace: default
spec:
    accessModes: ["ReadWriteMany"]
    resources:
        requests:
            storage: 5Gi
---
```

```
apiVersion: v1
kind: Pod
metadata:
     name: linuxea-pvc-pod
     namespace: default
spec:
     containers:
     - name: linuxea-pod1-pvc
         image: "marksugar/nginx:1.14.a"
         ports:
             - containerPort: 88
         volumeMounts:
         - name: linuxea-image
             mountPath: /data/wwwroot/
     volumes:
     - name: linuxea-image
         persistentVolumeClaim:
             claimName: linuxea-pvc
```

apply创建

```
[root@kube-node1]# kubectl apply -f pvc-demo.yaml
persistentvolumeclaim/linuxea-pvc created
pod/linuxea-pvc-pod created
```

可使用kubectl get pvc查看已经创建好的pvc已经被Bound

```
[root@kube-node1]# kubectl get pvc
NAME          STATUS    VOLUME      CAPACITY   ACCESS MODES   STORAGECLASS   AGE
linuxea-pvc   Bound     linuxea-5   5Gi        RWO,RWX                       6s
```

以及pod

```
[root@kube-node1]# kubectl get pods -o wide
NAME
 READY         STATUS       RESTARTS     AGE             IP                          NODE
linuxea-pvc-pod
1/1             Running     0                   3h               172.30.1.19       kube-node2
```

而后创建pvc之后，可查看pv已经被绑定到linuxea-5上的pv上（大于等于5G）

```
[root@kube-node1]# kubectl get pv
NAME        CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS      CLAIM
STORAGECLASS   REASON    AGE
linuxea-1   1Gi        RWO,RWX        Retain           Available
2m
linuxea-2   2Gi        RWO,RWX        Retain           Available
2m
linuxea-3   3Gi        RWO,RWX        Retain           Available
2m
linuxea-4   4Gi        RWO,RWX        Retain           Available
2m
```

```
linuxea-5   5Gi        RWO,RWX        Retain           Bound       default/linuxea-pvc
2m
```

也可以使用kubectl describe pods linuxea-pvc-pod|grep root查看信息

```
[root@kube-node1]# kubectl describe pods linuxea-pvc-pod|grep root
      /data/wwwroot/ from linuxea-image (rw)
```

pv写入测试

在集群内访问

```
[k8s@kube-node1 ~]$ curl 172.30.1.19
linuxea-linuxea-pvc-pod.com-127.0.0.1/8 172.30.1.19/24
```

而后回到nfs修改

```
[root@kube-node1 volumes]#   echo `date` >> /data/volumes/linuxea-5/index.html
```

在集群内第二次访问查看

```
[k8s@kube-node1 ~]$ curl 172.30.1.19
linuxea-linuxea-pvc-pod.com-127.0.0.1/8 172.30.1.19/24
Wed Apr 3 10:45:11 CST 2019
```

创建多大的pv，可能需要事先设定好，pvc才能适配