

# nfs storageClass

---

此时省略,nfs server的部署过程, 如有需要请参考:<https://blog.51cto.com/13701082/2342117> 需要注意的是, 需要将该/etc/sysconfig/nfs中以下注释的行打开

```
OCKD_TCPPORT=32803
LOCKD_UDPPORT=32769
MOUNTD_PORT=892
STATD_PORT=662
STATD_OUTGOING_PORT=2020
```

## 1. nfs server 服务器firewalld 开放端口

```
firewall-cmd --zone=public --add-port=111/tcp \
--add-port=2049/tcp \
--add-port=2049/udp \
--add-port=111/udp \
--add-port=4046/udp \
--add-port=32803/udp \
--add-port=32803/tcp \
--add-port=32769/tcp \
--add-port=32769/udp \
--add-port=892/tcp \
--add-port=892/udp \
--add-port=2020/tcp \
--add-port=2020/udp \
--add-port=662/tcp \
--add-port=662/udp --permanent
```

## 2. k8s集群主机安装nfs-utils

3. 修改nfs\_external/deployment.yaml文件,如下图所示:

```
template:
  metadata:
    labels:
      app: nfs-client-provisioner
  spec:
    serviceAccountName: nfs-client-provisioner
    containers:
      - name: nfs-client-provisioner
        image: 10.10.64.88/tools/quay.io/external_storage/nfs-client-provisioner:latest
        volumeMounts:
          - name: nfs-client-root
            mountPath: /persistentvolumes
        env:
          - name: PROVISIONER_NAME
            value: fuseim.pri/ifs
          - name: NFS_SERVER
            value: 10.10.64.88
          - name: NFS_PATH
            value: /k8s_store
    volumes:
      - name: nfs-client-root
        nfs:
          server: 10.10.64.88
          path: /k8s_store
```

Diagram annotations:

- image: 10.10.64.88/tools/quay.io/external\_storage/nfs-client-provisioner:latest (labeled "image")
- value: fuseim.pri/ifs (labeled "记住这个value,接下来要用到")
- value: 10.10.64.88 (labeled "nfs服务地址")
- value: /k8s\_store (labeled "nfs 共享路径")
- server: 10.10.64.88 (labeled "同上")
- path: /k8s\_store (labeled "同上")

4. 修改nfs\_external/class.yaml 中provisioner的值与deployment.yaml中PROVISIONER\_NAME的值一致
5. 在修改nfs\_external下执行deply.sh

```
sh deply.sh
```

6. 执行测试样例

```
kubectl apply -f test-claim.yaml
kubectl apply -f test-pod.yaml
```

如果在nfs共享目录中发现SUCCESS, 表明部署成功

注:创建PersistentVolumeClaim后, 会自动在nfs共享目录下创建以namespace + PersistentVolumeClaim name name + id 为格式的目录。