

实际上，OCP 集群在刚装完后，是没有 image-registry 这个 deployment 的，只有 cluster-image-registry-operator 这个 deployment。

```
# oc project openshift-image-registry
```

```
# oc get deployments
```

安装后，我们需要修改 Image Registry Operator 的配置，将 ManagementState 从 Removed 修改为 Managed：

```
# oc editconfigs.imageregistry.operator.openshift.io cluster
```

将

```
managementState:Removed
```

修改为：

```
managementState:Managed
```

如果只是 PoC 环境，可以将 Image Registry 设置为空目录，如下所示。但笔者建议所有环境都为其配置持久化存储。

```
#oc patch configs.imageregistry.operator.openshift.io cluster \--type merge --patch '{"spec":{"storage":{"emptyDir":{}}}}'
```

执行如下命令，指定 registry image 需要 pvc：

```
#oc editconfigs.imageregistry.operator.openshift.io
```

在如下位置增加 pvc.claim。注意添加的位置要对，在 spec 下。

```
spec:
  httpSecret: 0365e61252d809ccf27e59886c12261181a6
  66b15655ea4e8ca490c
  logLevel: Normal
  logging: 2
  managementState: Managed
  observedConfig: null
  operatorLogLevel: Normal
  proxy: {}
  replicas: 1
  requests:
    read:
      maxWaitInQueue: 0s
    write:
      maxWaitInQueue: 0s
  rolloutStrategy: RollingUpdate
  storage:
    managementState: Managed
  pvc:
    claim:
```

这时候，可以通过查看 co:

```
# oc describe co image-registry
```

这时候，会有类似如下的报错：

Unable to apply resources: storage backend not configured"

执行如下命令，可以看到 pending 的 pvc，名称为：image-registry-storage

```
#oc get pvc
```

此时，查看 pvc 需要的 pv 的读写模式和大小，通常为 ReadWriteMany、100G。

我们手工创建 pv 即可。

例如：

```
[root@bastion ~]# cat 6.yaml
```

```
apiVersion: v1
```

```
kind: PersistentVolume
```

```
metadata:
```

```
name: image-registry-pv
```

```
spec:
```

```
accessModes:
```

```
- ReadWriteMany
```

capacity:
storage: 100Gi
nfs:
path: /var/lib/nfsserver/pvregistry1
server: 172.16.32.88
persistentVolumeReclaimPolicy: Retain
storageClassName: nfs-storage-provisioner

然后，确保 pvc 和 pv 成功 bound。

最后，增加 image registry 的副本数量：

#oc edit configs.imageregistry.operator.openshift.io cluster

```
spec:  
  httpSecret: 0365e61252d809ccf27e59886c1  
66b15655ea4e8ca490c  
  logLevel: Normal  
  logging: 2  
  managementState: Managed  
  observedConfig: null  
  operatorLogLevel: Normal  
  proxy: {}  
  replicas: 1  
  requests:  
    read:  
      maxWaitInQueue: 0s
```