



链滴

kubernetes 部署 Minio Operator 部署

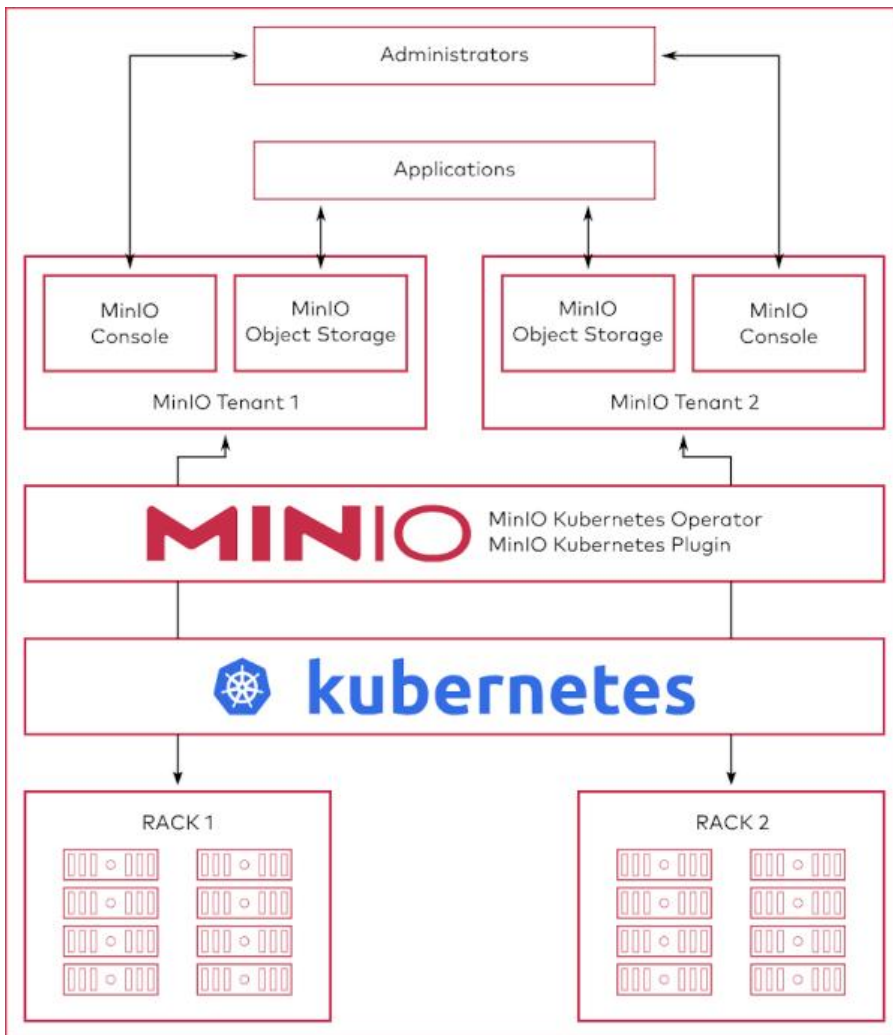
作者: [wangjunjack](#)

原文链接: <https://ld246.com/article/1616561969694>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

一、系统架构



实现机制

基于kubernetes中的CSI中的stroageClass，实现动态申请pvc作为minio的存储节点，来实现minio象存储的动态扩容机制。

二、测试环境准备

1. 服务器硬件和软件

硬件

型号	CPU	Memory	Disk
Dell R410 B*1	Intel E5620 Nic 1GB	16GB	SATA 300
Dell R410 B*2	Intel E562 Nic 1GB	16GB	SATA 300
Dell R410	Intel E562	16GB	SATA 300

B*2	Nic 1GB		
Dell R410	Intel E562	16GB	SATA 300
B*2	Nic 1GB		

软件

Role	IP	Kubernetes	Docker
Master 5	192.168.1.240 CentOS Linux release 7.9.2009 (Core)	V1.18.16	19.03.
Worker 5	192.168.1.241 CentOS Linux release 7.9.2009 (Core)	V1.18.16	19.03.
Worker 5	192.168.1.242 CentOS Linux release 7.9.2009 (Core)	V1.18.16	19.03.
Worker 5	192.168.1.243 CentOS Linux release 7.9.2009 (Core)	V1.18.16	19.03.

三、安装 minio/operator

github项目地址 <https://github.com/minio/operator/>

部署前提条件

1. 准备好kubernetes集群，本次测试使用4节点集群

```
[root@ds-n1 ~]# kubectl get node
NAME      STATUS   ROLES    AGE   VERSION
ds-n1    Ready   controlplane,etcd,worker 46h   v1.18.16
ds-n2    Ready   worker   46h   v1.18.16
ds-n3    Ready   worker   46h   v1.18.16
ds-n4    Ready   worker   46h   v1.18.16
```

2. 查看准备好的动态存储，测试用的是longhorn (longhorn,openebs)

```
[root@ds-n1 ~]# kubectl get sc
NAME                PROVISIONER          RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
longhorn (default)  driver.longhorn.io  Delete          Immediate            true                    46h
```

部署minio operator

1. 安装minio插件

本次使用二进制安装，从github下载安装。

```
# Operator Version 4.0.3
wget https://github-releases.githubusercontent.com/154578184/e9d8c680-87f4-11eb-89a6-8250108d7b0?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4SVEH53A%2F20210319%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20210319T02196Z&X-Amz-Expires=300&X-Amz-Signature=4610252db79ae812f349682609152a0f2000cff77111bb392da3e0ac8d49006&X-Amz-SignedHeaders=host&actor_id=30251247&key_id=0&re
```

```
o_id=154578184&response-content-disposition=attachment%3B+filename%3Dkubectl-mini_4.0.3_linux_amd64&response-content-type=application%2Foctet-stream
```

给kubectl-minio_4.0.3_linux_amd64赋予执行权限，并copy到/usr/local/bin目录下并重命名

```
chmod +x kubectl-minio_4.0.3_linux_amd64
cp kubectl-minio_4.0.3_linux_amd64 /usr/local/bin/kubectl-minio
```

通过kubectl plugin list命令查看minio插件安装状态

```
[root@ds-n1 ~]# kubectl plugin list
The following compatible plugins are available:

/usr/local/bin/kubectl-minio
Unable read directory "/root/bin" from your PATH: open /root/bin: no such file or directory. Sipping...
```

如果不显示插件，手动添加环境变量

3. 初始化Minio operator

首先需要初始化minio operator，operator创建完成后，便可以继续进行Tenant创建

```
[root@ds-n1 ~]# kubectl minio init

[root@ds-n1 ~]# kubectl get pod -n minio-operator
NAME                                READY STATUS RESTARTS AGE
console-5fcbf4b7df-m6wfg            1/1   Running 0      46h
minio-operator-9c79bb668-djvn5     1/1   Running 0      46h
```

部署minio存储服务

```
kubectl minio tenant create tenant1 --namespace minio-tenant-1 --storage-class longhorn --servers 4 --volumes 4 --capacity 480Gi
```

查看创建的资源

```
[root@ds-n1 ~]# kubectl get pod -n minio-tenant-1
NAME                                READY STATUS RESTARTS AGE
minio-tenant-1-console-7df4467c65-hvzf8  1/1   Running 0      45h
minio-tenant-1-console-7df4467c65-z6qpf  1/1   Running 0      45h
minio-tenant-1-ss-0-0                    1/1   Running 0      45h
minio-tenant-1-ss-0-1                    1/1   Running 0      45h
minio-tenant-1-ss-0-2                    1/1   Running 0      45h
minio-tenant-1-ss-0-3                    1/1   Running 0      45h
```

修改service类型为NodePort，方便访问minio UI以及tenant-console UI:

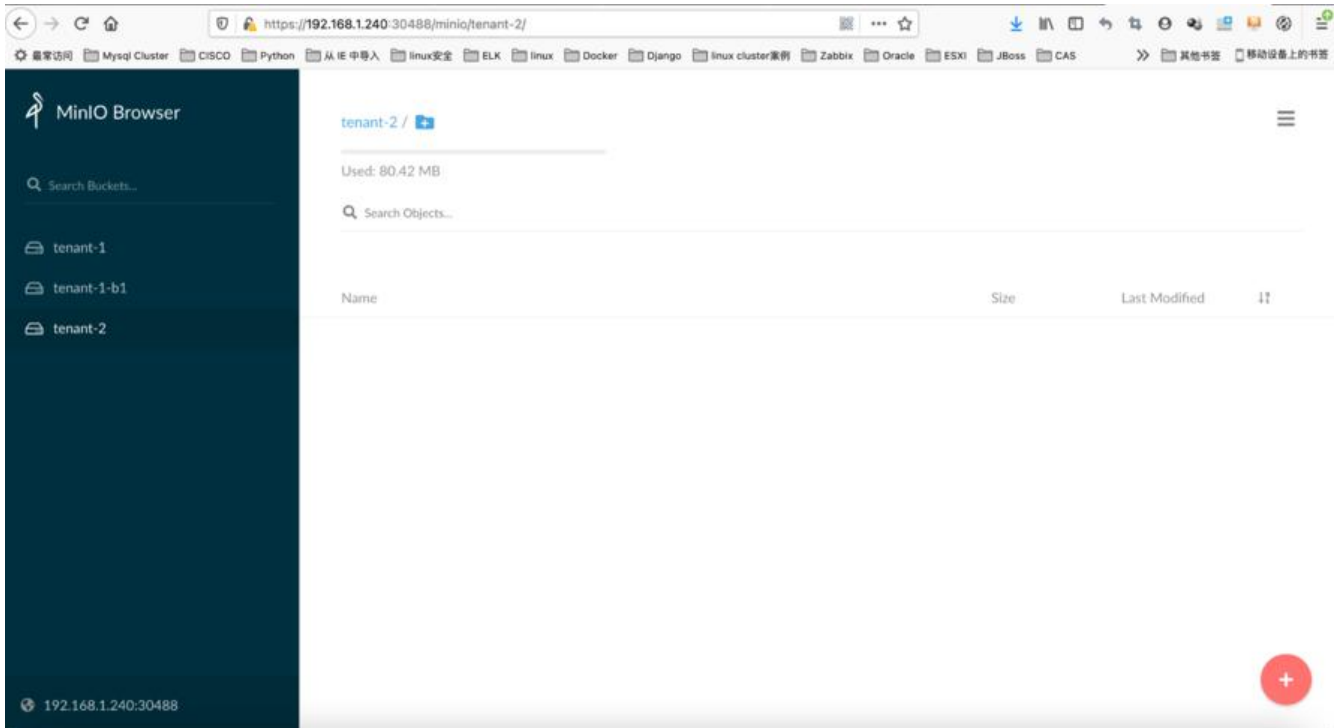
```
kubectl -n minio-tenant-1 patch svc minio -p '{"spec": {"type": "NodePort"}}'
kubectl -n minio-tenant-1 patch svc minio-tenant-1-console -p '{"spec": {"type": "NodePort"}}'
```

查看service，记录minio及tenant1-console中的nodeport

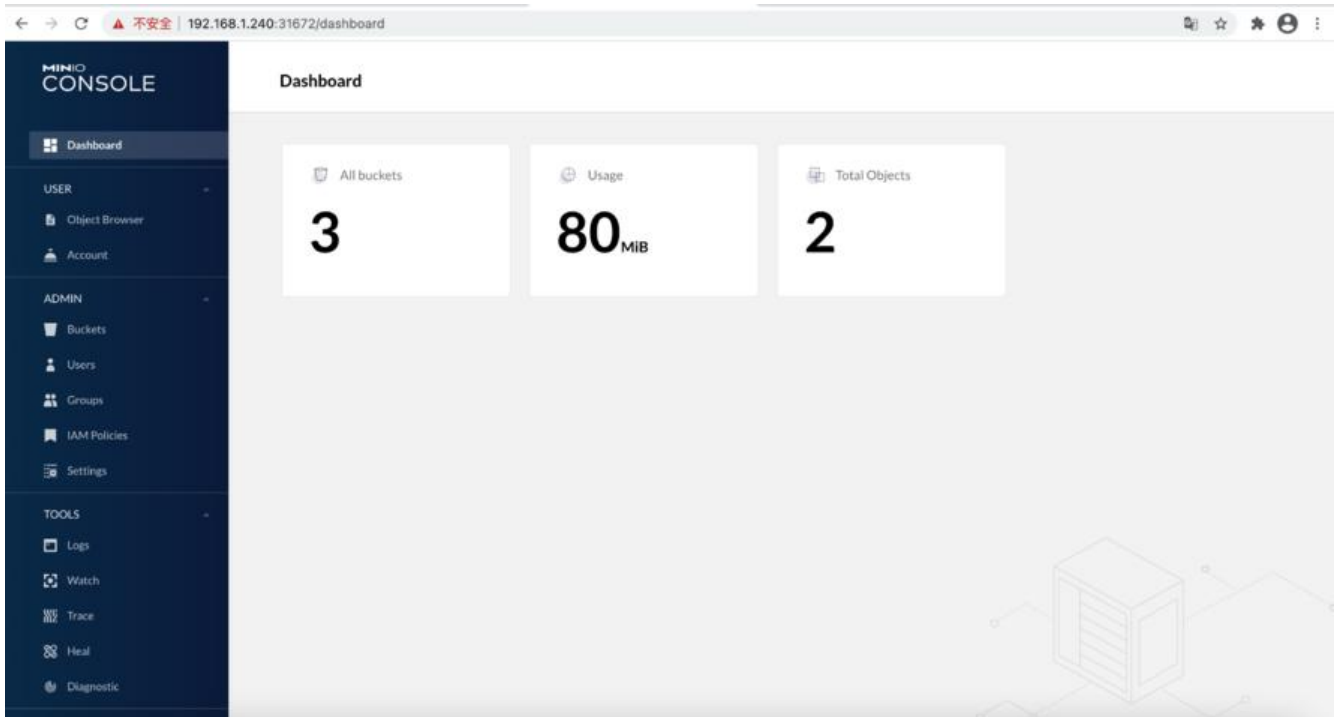
```
[root@ds-n1 ~]# kubectl get svc -n minio-tenant-1
NAME                                TYPE      CLUSTER-IP    EXTERNAL-IP  PORT(S)    AGE
```

```
minio NodePort 10.43.126.203 <none> 443:30488/TCP 45h
minio-tenant-1-console NodePort 10.43.174.32 <none> 9443:31672/TCP 45h
minio-tenant-1-hl ClusterIP None <none> 9000/TCP 45h
```

使用nodeport访问minio UI, 使用https方式:



使用nodeport访问tenant-console UI, 使用https方式:



五、Minio扩容

使用kubectl minio插件向租户添加容量

```
kubectl minio tenant expand minio-tenant-1 longhorn --servers 8 --volumes 8 --capacity 480
i --namespace minio-tenant-1 ----storage-class
```

查看资源创建结果

```
[root@ds-n1 ~]# kubectl get pod -n minio-tenant-1
NAME                                READY STATUS  RESTARTS  AGE
minio-tenant-1-console-7df4467c65-hvzf8  1/1   Running  0         45h
minio-tenant-1-console-7df4467c65-z6qpf  1/1   Running  0         45h
minio-tenant-1-ss-0-0                    1/1   Running  0         45h
minio-tenant-1-ss-0-1                    1/1   Running  0         45h
minio-tenant-1-ss-0-2                    1/1   Running  0         45h
minio-tenant-1-ss-0-3                    1/1   Running  0         45h
minio-tenant-1-ss-1-0                    1/1   Running  0         45h
minio-tenant-1-ss-1-1                    1/1   Running  0         45h
minio-tenant-1-ss-1-2                    1/1   Running  0         45h
minio-tenant-1-ss-1-3                    1/1   Running  0         45h
minio-tenant-1-ss-1-4                    1/1   Running  0         45h
minio-tenant-1-ss-1-5                    1/1   Running  0         45h
minio-tenant-1-ss-1-6                    1/1   Running  0         45h
minio-tenant-1-ss-1-7                    1/1   Running  0         45h
```

这将为租户增加8个驱动器，这些驱动器均匀分布在4台服务器上tenant1，并具有480Gi的额外容量

至此，完成了Minio Operator的部署，bucket新建和文件上传；完成了minio的容量扩容。