加入语雀，获得更好的阅读体验
注册 或 登录 后可以收藏本文随时阅读，还可以关注作者获得最新文章推送
立即加入

yuque.com/grep

管理资源的基本方法

陈述式管理方法

主要依赖命令行CLI工具进行管理

查看相关信息

get

基本信息查看

```
Usage:  kubectl get resource [-o wide|json|yaml] [-n namespace]
Man:    获取资源的相关信息，-n 指定名称空间，-o 指定输出格式
        resource可以是具体资源名称，如pod nginx-xxx；也可以是资源类型，如pod；或者all(仅展示几种核心资源，并不完整)
        -A, --all-namespace 表示显示所有名称空间
```

```
~]# kubectl get cs
~]# kubectl get node -o wide
~]# kubectl get svc -o wide -n kube-system
~]# kubectl get pod -A
~]# kubectl get pod nginx-ds-jdp7q -o yaml # 可以导出yaml文件
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: "2020-01-13T13:13:02Z"
  generateName: nginx-ds-
  labels:
    app: nginx-ds
......
```

根据标签筛选

Plain Text 复制代码

```
--show-labels  显示所有标签
-l app         仅显示标签为app的资源
-l app=nginx   仅显示包含app标签，且值为nginx的资源
```

Plain Text 复制代码

```
~]# kubectl get pod -n app --show-labels
NAME       READY   STATUS    RESTARTS   AGE   LABELS
pod-02     1/1     Running   0          9h    app=nginx,release=stable,version=v1.12
pod-demo   1/1     Running   9          9h    app=centos7,environment=dev,release=stable
~]# kubectl get pod -n app --show-labels -l app
NAME       READY   STATUS    RESTARTS   AGE   LABELS
pod-02     1/1     Running   0          9h    app=nginx,release=stable,version=v1.12
pod-demo   1/1     Running   9          9h    app=centos7,environment=dev,release=stable
~]# kubectl get pod -n app --show-labels -l app=nginx
NAME     READY   STATUS    RESTARTS   AGE   LABELS
pod-02   1/1     Running   0          9h    app=nginx,release=stable,version=v1.12
```

describe

Plain Text 复制代码

```
Usage: kubectl describe (-f FILENAME | TYPE [NAME_PREFIX | -l label] | TYPE/NAME) [-n namespace]
Man:   描述某个资源信息
```

Plain Text 复制代码

```
~]# kubectl describe svc nginx-web
Name:              nginx-web
......

~]# kubectl describe pod -l app=nginx-web
Name:           nginx-web-796c86d7cd-8kst5
Namespace:      default
......
```

yuque.com/grep

其它集群信息

yuque.com/grep

Plain Text 复制代码

```
~]# kubectl version  # 集群版本
Client Version: version.Info{Major:"1", Minor:"15", GitVersion:"v1.15.2", GitCommit:"f6278300bebbb750328ac16ee6dd3aa7d3549568", GitTreeState:"clean", BuildDate:"2019-08-05T09:23:26Z", GoVersion:"go1.12.5", Compiler:"gc", Platform:"linux/amd64"}
Server Version: version.Info{Major:"1", Minor:"15", GitVersion:"v1.15.2", GitCommit:"f6278300bebbb750328ac16ee6dd3aa7d3549568", GitTreeState:"clean", BuildDate:"2019-08-05T09:15:22Z", GoVersion:"go1.12.5", Compiler:"gc", Platform:"linux/amd64"}
```

Plain Text 复制代码

```
~]# kubectl cluster-info  # 集群信息
Kubernetes master is running at http://localhost:8080
CoreDNS is running at http://localhost:8080/api/v1/namespaces/kube-system/services/coredns:dns/proxy
kubernetes-dashboard is running at http://localhost:8080/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'
```

创建资源

create

Plain Text 复制代码

```
Uage:   kubectl create -f filename.yaml
        kubectl create resourece [options]
Man:    根据清单文件或者指定的资源参数创建资源
```

创建名称空间

Plain Text 复制代码

```
~]# kubectl create namespace app
~]# kubectl get ns app
NAME   STATUS   AGE
app    Active   10s
```

创建deployment

yuque.com/grep

Plain Text 复制代码

```
~]# kubectl create deployment app-deploy --image=harbor.od.com/public/nginx:latest -n app
~]# kubectl get all -n app
NAME                             READY   STATUS    RESTARTS   AGE
pod/app-deploy-5b5649fc4-plbxg   1/1     Running   0          13s

NAME                         READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/app-deploy   1/1     1            1           13s

NAME                                   DESIRED   CURRENT   READY   AGE
replicaset.apps/app-deploy-5b5649fc4   1         1         1       13s
```

创建service资源

Plain Text 复制代码

```
Usage: Usage:  kubectl expose TYPE NAME [--port=port] [--protocol=TCP|UDP|SCTP] [--target-port=n] [--name=name] [--external-ip=external-ip-of-service] [options]
Man:    TYPE为deployment,NAME为depoly资源名称，port和target-port分别为集群和pod的端口
```

Plain Text 复制代码

```
~]# kubectl expose deployment app-deploy --port=80 --target-port=80 --name=app-svc -n app
~]# kubectl describe svc app-svc -n app
Name:              app-svc
Namespace:         app
Labels:            app=app-deploy
Annotations:       <none>
Selector:          app=app-deploy
Type:              ClusterIP
IP:                192.168.28.124
Port:              <unset>  80/TCP
TargetPort:        80/TCP
Endpoints:         172.7.21.8:80
```

yuque.com/grep yuque.com/grep

扩缩容

Plain Text 复制代码

```
Usage:  kubectl scale --replicas=COUNT TYPE NAME [options]
Man:    对资源进行扩缩容，即修改副本数
```

yuque.com/grep yuque.com/grep

Plain Text 复制代码

```
~]# kubectl get deploy web-deploy
NAME         READY   UP-TO-DATE   AVAILABLE   AGE
web-deploy   2/2     2            2           37m
~]# kubectl scale --replicas=5 deployment web-deploy  # 扩容
~]# kubectl get deploy web-deploy
NAME         READY   UP-TO-DATE   AVAILABLE   AGE
web-deploy   3/5     5            3           38m

~]# kubectl scale --replicas=1 deployment web-deploy  # 缩容
~]# kubectl get deploy web-deploy
NAME         READY   UP-TO-DATE   AVAILABLE   AGE
web-deploy   1/1     1            1           38m
```

删除资源

Plain Text 复制代码

```
Usage:  kubectl delete ([-f FILENAME] | [-k DIRECTORY] | TYPE [(NAME | -l label | --all)]) [options]
Man:    删除指定资源
```

Plain Text 复制代码

```
~]# kubectl get deployment -n app
NAME                        READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/app-deploy   1/1     1            1           35m
~]# kubectl delete deployment app-deploy -n app
deployment.extensions "app-deploy" deleted

~]# kubectl delete ns app
namespace "app" deleted
```

强制删除

yuque.com/grep

Plain Text 复制代码

```
~]# kubectl delete pod nginx-dp-5dfc689474-4whfn -n kube-public --force --grace-period=0
```

进入容器

yuque.com/grep

Plain Text 复制代码

```
Usage:  kubectl exec (POD | TYPE/NAME) [-c CONTAINER] [flags] -- COMMAND [args...] [options]
```

Plain Text 复制代码

```
~]# kubectl exec nginx-web-796c86d7cd-zx2b9 -it -- /bin/bash  # 交互式
root@nginx-web-796c86d7cd-zx2b9:/# exit
exit

~]# kubectl exec nginx-web-796c86d7cd-zx2b9 -- cat /etc/resolv.conf # 不进入容器执行命令
nameserver 192.168.0.2
search default.svc.cluster.local svc.cluster.local cluster.local host.com
options ndots:5

~]# kubectl exec nginx-web-796c86d7cd-zx2b9 cat /etc/resolv.conf  # 不进入容器执行命令
nameserver 192.168.0.2
search default.svc.cluster.local svc.cluster.local cluster.local host.com
options ndots:5
```

如果一个pod有两个container需要使用-c参数选择进入的container

查看资源清单文档

Plain Text 复制代码

```
~]# kubectl api-versions  # 查看api-version信息
apps/v1
node.k8s.io/v1beta1
v1
......
```

Plain Text 复制代码

```
Usage:  kubectl explain RESOURCE [options]
Man:    查看各个字段的解释，随着k8s版本的更新，不同资源的Apiversion会发生变化，当出现API错误时，需要查询当前的API版本

~]# kubectl explain pod.spec.containers
KIND:     Pod
VERSION:  v1

RESOURCE: containers <[]Object>
```

陈述式资源管理方法小结：

1.kubernetes集群管理集群资源的唯一入口是通过相应的方法调用apiserver的接口

yuque.com/grep

2.kubectl是官方的CLI命令行工具，用于与apiserver进行通信，将用户在命令行输入的命令，组织并转化为apiserver能识别的信息，进而实现管理k8s各种资源的一种有效途径

3.kubectl的命令大全

kubectl --help

k8s中文文档 <http://docs.kubernetes.org.cn/>

yuque.com/grep

yuque.com/grep

声明式管理方法

主要依赖统-资源配置清单(manifest)进行管理

Plain Text 复制代码

```
~]# kubectl explain deployment.metadata
~]# kubectl explain service.metadata
~]# kubectl edit daemonset nginx-ds

~]# kubectl apply -f nginx-ds.yaml
~]# kubectl delete -f nginx-ds.yaml
```

yuque.com/grep

yuque.com/grep

GUI式管理方法

主要依赖图形化操作界面(web页面)进行管理