6Kubernetes系列（七）Kubectl CLI 使用

Kubernetes系列（七）Kubectl CLI 使用

本文主要介绍 Kubectl 命令行工具的基本用法。

环境配置

默认情况下，不管是 minikube 还是常规的 k8s 集群安装，都会在默认的用户主目录下面创建一个 ~/.kube/config 文件，kubectl 默认读取该配置的集群信息进行操作，大家可以本地操作远程集群，通过替换改配置文件。详细安装可参考：`https://kubernetes.io/zh-cn/docs/tasks/tools/`

基本使用

获取资源类型

集群支持的资源 CRD 类型，可以通过如下命令获取：

```
[root@master ~]# kubectl api-resources NAME                                    SHORTNAMES        APIVERSION                              NAMESPACED    KINDbindings
```

- NAME : api 资源名称
- SHORTNAMES: api 资源简称，在查询时可以使用简称
- APIVERSION: api 资源版本
- NAMESPACED: api 资源是否是命名空间范围的，比如 pv 的值就是 false 代表 pv 是全局的，不是限定于某个具体命名空间的
- KIND：api 资源类型

查询资源清单配置结构信息

在 yaml 清单配置某类资源时，碰到不知道某段配置具体的路径以及值类型、是否必填时，可以通过如下命令查看，比如，查看 pod配置：

```
# 查看 pod 第一层级的 配置信息，每段配置有详细的配置[root@master ~]# kubectl explain podKIND:     PodVERSION:  v1
DESCRIPTION:   Pod is a collection of containers that can run on a host. This resource is   created by clients and scheduled onto hosts.
FIELDS: apiVersion   <string>   APIVersion defines the versioned schema of this representation of an   object. Servers should convert recognized schemas to the latest internal
 kind <string>   Kind is a string value representing the REST resource this object   represents. Servers may infer this from the endpoint the client submits   requests to. Cannot
 metadata     <Object>   Standard object's metadata. More info:   https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata
 spec <Object>   Specification of the desired behavior of the pod. More info:   https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#spec-and-status
 status       <Object>   Most recently observed status of the pod. This data may not be up to date.   Populated by the system. Read-only. More info:   https://git.k8s.io/community
RESOURCE: spec <Object>
DESCRIPTION:   Specification of the desired behavior of the pod. More info:   https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#spec-and-status
   PodSpec is a description of a pod.
FIELDS: activeDeadlineSeconds        <integer>   Optional duration in seconds the pod may be active on the node relative to   StartTime before the system will actively try to mark
 affinity     <Object>   If specified, the pod's scheduling constraints
 automountServiceAccountToken <boolean>   AutomountServiceAccountToken indicates whether a service account token   should be automatically mounted.
 containers   <[]Object> -required-   List of containers belonging to the pod. Containers cannot currently be   added or removed. There must be at least one container in a Pod. Ca
 dnsConfig    <Object>   Specifies the DNS parameters of a pod. Parameters specified here will be   merged to the generated DNS configuration based on DNSPolicy.
 dnsPolicy    <string>   Set DNS policy for the pod. Defaults to "ClusterFirst". Valid values are   'ClusterFirstWithHostNet', 'ClusterFirst', 'Default' or 'None'. DNS   parameter


 enableServiceLinks   <boolean>   EnableServiceLinks indicates whether information about services should be   injected into pod's environment variables, matching the syntax of Do
 ephemeralContainers  <[]Object>   List of ephemeral containers run in this pod. Ephemeral containers may be   run in an existing pod to perform user-initiated actions such as deb
 hostAliases  <[]Object>   HostAliases is an optional list of hosts and IPs that will be injected into   the pod's hosts file if specified. This is only valid for non-hostNetwork
 hostIPC      <boolean>   Use the host's ipc namespace. Optional: Default to false.
 hostNetwork  <boolean>   Host networking requested for this pod. Use the host's network namespace.   If this option is set, the ports that will be used must be specified.   Defau
 hostPID      <boolean>   Use the host's pid namespace. Optional: Default to false.
 hostname     <string>   Specifies the hostname of the Pod If not specified, the pod's hostname will   be set to a system-defined value   ......      以此类推
```

查询某类资源

以查询 POD 为例，其他类型的资源类型同样的查询方法，只是替换个类型：

```
# kubectl get  pod  -n [命名空间]， 不指定命名空间，只会查询默认的命名空间：default [root@master ~]# kubectl get pod -n cosNAME                                                                                                READY   SI
# -o wide 输出更多列[root@master ~]# kubectl get pod -n cos -o wideNAME                                                                READY   STATUS    RESTARTS       AGE     IP
# 查询某个具体 pod 的明细[root@master ~]# kubectl describe pod cloud-bmp-7d688998f8-qprvw -n cosName:         cloud-bmp-7d688998f8-qprvwNamespace:    cosPriority:     0Node:
# 根据 label 选择器匹配 [root@master ~]# kubectl get pod -n cos -l app=cloud-bmpNAME                          READY   STATUS    RESTARTS   AGEcloud-bmp-7d688998f8-qprvw   1/1     Run
```

pod 日志 & 登录

- 查看 pod 日志

```
# kubectl logs [pod name] -n [命名空间][root@master ~]# kubectl logs cloud-bmp-7d688998f8-qprvw -n cos[INFO ] 2022-07-21 17:50:12.695 [org.springframework.amqp.rabbit.listener.Simp
```

- 登录，进入 pod 容器

```
# kubectl exec -it [pod name] -c [container name] -n [namespace] [command] (command 可以是 bash 这种直接登录的，也可以直接执行远程命令)[root@master ~]# kubectl exec -it cloud-bmp-7d68899
# 远程命令 （无需登录，执行 ls 命令）[root@master ~]# kubectl exec -it cloud-bmp-7d688998f8-qprvw -c cloud-bmp -n cos ls kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed
```

与宿主机拷贝文件

- 从 pod拷贝文件到 宿主机

```
# kubectl cp [namespace]/[podname]:容器中绝对路径 宿主机目标路径（容器绝对路径前的 / 不要加）[root@master ~]# kubectl cp default/node:etc/hosts  ./hosts[root@master ~]# lltotal 258400-rw-r--
```

- 将宿主机中的文件拷贝到容器中

-
-
-
-
-
-
-
-
-

```
# kubectl cp 宿主机文件路径 [namespace]/[podname]:容器中目标路径[root@master ~]# kubectl cp /root/test.yaml  default/node:/etc[root@master ~]# kubectl exec -it node -n default ls more
```

上一篇

欢迎关注我的公众号“ **云原生拓展** ”，原创技术文章第一时间推送。