



链滴

Kubernetes 的 yaml 文件注释

作者: [gongdear](#)

原文链接: <https://ld246.com/article/1544589802655>

来源网站: 链滴

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

```

#test-pod
apiVersion: v1 #指定api版本, 此值必须在kubectl apiversion中
kind: Pod #指定创建资源的角色/类型
metadata: #资源的元数据/属性
name: test-pod #资源的名字, 在同一个namespace中必须唯一
labels: #设定资源的标签
k8s-app: apache
version: v1
kubernetes.io/cluster-service: "true"
annotations:      #自定义注解列表
- name: String    #自定义注解名字
spec: #specification of the resource content 指定该资源的内容
restartPolicy: Always #表明该容器一直运行, 默认k8s的策略, 在此容器退出后, 会立即创建一个相
的容器
nodeSelector:     #节点选择, 先给主机打标签kubectl label nodes kube-node1 zone=node1
zone: node1
containers:
  • name: test-pod #容器的名字
image: kvmdocker:5000/test/chat:latest #容器使用的镜像地址
imagePullPolicy: Never #三个选择Always、Never、IfNotPresent, 每次启动时检查和更新 (从regi
tery) images的策略,
# Always, 每次都检查
# Never, 每次都不检查 (不管本地是否有)
# IfNotPresent, 如果本地有就不检查, 如果没有就拉取
command: ['sh'] #启动容器的运行命令, 将覆盖容器中的Entrypoint,对应Dockerfile中的ENTRYPOI
T
args: ["$(str)"] #启动容器的命令参数, 对应Dockerfile中CMD参数
env: #指定容器中的环境变量
  • name: str #变量的名字
value: "/etc/run.sh" #变量的值
resources: #资源管理
requests: #容器运行时, 最低资源需求, 也就是说最少需要多少资源容器才能正常运行
cpu: 0.1 #CPU资源 (核数), 两种方式, 浮点数或者是整数+m, 0.1=100m, 最少值为0.001核 (1
)
memory: 32Mi #内存使用量
limits: #资源限制
cpu: 0.5

```

memory: 1000Mi

ports:

- containerPort: 80 #容器开发对外的端口

name: httpd #名称

protocol: TCP

livenessProbe: #pod内容器健康检查的设置

httpGet: #通过httpget检查健康, 返回200-399之间, 则认为容器正常

path: / #URI地址

port: 80

#host: 127.0.0.1 #主机地址

scheme: HTTP

initialDelaySeconds: 180 #表明第一次检测在容器启动后多长时间后开始

timeoutSeconds: 5 #检测的超时时间

periodSeconds: 15 #检查间隔时间

#也可以用这种方法

#exec: 执行命令的方法进行监测, 如果其退出码不为0, 则认为容器正常

command:

- cat

- /tmp/health

#也可以用这种方法

#tcpSocket: //通过tcpSocket检查健康

port: number

lifecycle: #生命周期管理

postStart: #容器运行之前运行的任务

exec:

command:

- 'sh'

- 'yum upgrade -y'

preStop:#容器关闭之前运行的任务

exec:

command: ['service httpd stop']

volumeMounts: #挂载持久存储卷

- name: volume #挂载设备的名字, 与volumes[*].name 需要对应

mountPath: /data #挂载到容器的某个路径下

readOnly: True

volumes: #定义一组挂载设备

- name: volume #定义一个挂载设备的名字

#emptyDir: {}

hostPath:

path: /opt #挂载设备类型为hostPath, 路径为宿主机下的/opt,这里设备类型支持很多种

#nfs

kubernetes日志查看

kubectl logs -f --tail 100 gateway-dev-678b7988d8-wfhsr --namespace=development