<https://hub.docker.com/search?q=redis&type=image>

部署Redis到k8s上

apiVersion: apps/v1

kind: Deployment

metadata:

labels:

app: redis-single-node

name: redis-single-node

namespace: ratel-test1spec:

progressDeadlineSeconds: 600

replicas: 1

revisionHistoryLimit: 10

selector:

matchLabels:

app: redis-single-node

strategy:

rollingUpdate:

maxSurge: 1

maxUnavailable: 0

type: RollingUpdate

template:

metadata:

creationTimestamp: null

labels:

app: redis-single-node

spec:

containers:

- command:

- sh

- -c

- redis-server "/mnt/redis.conf"

env:

- name: TZ

value: Asia/Shanghai

- name: LANG

value: C.UTF-8

image: redis:5.0.4-alpine

imagePullPolicy: IfNotPresent

lifecycle: {}

livenessProbe:

failureThreshold: 2

initialDelaySeconds: 10

periodSeconds: 10

successThreshold: 1

tcpSocket:

port: 6379

timeoutSeconds: 2

name: redis-single-node

ports:

- containerPort: 6379

name: web

protocol: TCP

readinessProbe:

failureThreshold: 2

initialDelaySeconds: 10

periodSeconds: 10

successThreshold: 1

tcpSocket:

port: 6379

timeoutSeconds: 2

resources:

limits:

cpu: 100m

memory: 339Mi

requests:

cpu: 10m

memory: 10Mi

securityContext:

privileged: false

runAsNonRoot: false

terminationMessagePath: /dev/termination-log

terminationMessagePolicy: File

volumeMounts:

- mountPath: /usr/share/zoneinfo/Asia/Shanghai

name: tz-config

- mountPath: /etc/localtime

name: tz-config

- mountPath: /etc/timezone

name: timezone

- mountPath: /mnt

name: redis-conf

readOnly: true

dnsPolicy: ClusterFirst

restartPolicy: Always

schedulerName: default-scheduler

securityContext: {}

terminationGracePeriodSeconds: 30

tolerations:

- effect: NoExecute

key: node.kubernetes.io/unreachable

operator: Exists

tolerationSeconds: 30

- effect: NoExecute

key: node.kubernetes.io/not-ready

operator: Exists

tolerationSeconds: 30

volumes:

- hostPath:

path: /usr/share/zoneinfo/Asia/Shanghai

type: ""

name: tz-config

- hostPath:

path: /etc/timezone

type: ""

name: timezone

- configMap:

defaultMode: 420

name: redis-conf

name: redis-conf

Operator模板：

<https://github.com/operator-framework/awesome-operators>

<https://github.com/ucloud/redis-cluster-operator>

清理集群

kubectl delete -f deploy/example/redis.kun\_v1alpha1\_distributedrediscluster\_cr.yaml

kubectl delete -f deploy/cluster/operator.yaml

kubectl delete -f deploy/cluster/cluster\_role\_binding.yaml

kubectl delete -f deploy/cluster/cluster\_role.yaml

kubectl delete -f deploy/service\_account.yaml

kubectl delete -f deploy/crds/redis.kun\_redisclusterbackups\_crd.yaml

kubectl delete -f deploy/crds/redis.kun\_distributedredisclusters\_crd.yaml

RabbitMQ集群安装：

StatefulSet

<https://github.com/dotbalo/k8s/tree/master/k8s-rabbitmq-cluster>

helm安装文档：

<https://helm.sh/docs/intro/install/>

创建一个chart

helm create helm-test

├── charts # 依赖文件

├── Chart.yaml # 这个chart的版本信息

├── templates #模板

│   ├── deployment.yaml

│   ├── \_helpers.tpl # 自定义的模板或者函数

│   ├── ingress.yaml

│   ├── NOTES.txt #这个chart的信息

│   ├── serviceaccount.yaml

│   ├── service.yaml

│   └── tests

│   └── test-connection.yaml

└── values.yaml #配置全局变量或者一些参数

values.yaml

with:

helm install test --dry-run .

include：引入的函数或者模板，\_helpers.tpl

define：定义一个模板，

trunc：只取前多少位字符，负号代表从后往前取

trimSuffix： 字符串末尾去掉指定的字符，Prefix

$name := xxx 定义一个变量

default： 定义的变量的默认值

contains： 判断字符串是否包含某个字符串

replace： 替换字符串

常用函数：<http://masterminds.github.io/sprig/strings.html>

helm install helm-test2 --set fullnameOverride=aaaaaaa --dry-run .

--set 修改values里面的值

helm install rabbitmq-cluster --namespace public-service --set replicaCount=2 .

删除helm uninstall rabbitmq-cluster -n public-service # helm v3 --keep-history

helm delete/del NAME --purge

升级helm upgrade rabbitmq-cluster -n public-service .