K8s Yaml编写小技巧

学习使用k8s的童鞋都知道我们在部署pod的时候有时候需要手动去编写一些yaml文件；比如我需要编写deployment,那除了在其他地方粘贴拷贝外有没有其他方法呢？答案是有的

1.用run命令生成，然后作为模板进行编辑。

```
kubectl run --image=nginx my-deploy -o yaml --dry-run > my-deploy.yaml
```

2.用get命令导出，然后作为模板进行编辑。

```
# 注意: --export 是为了去除当前正在运行的这个deployment生成的一些状态，我们用不到就过滤掉
kubectl get deployment/nginx -o=yaml --export  > new.yaml
```

3.Pod亲和性下面字段的拼写忘记了

```
kubectl explain pod.spec.affinity.podAffinity
```

示例:

我想生成一个有三个副本的redis pod的yaml，然后我想把这三个pod 通过node亲和性调度到同一个node节点上面；

1. 我这里用kubectl run来生成:

```
kubectl run redis --image=redis --replicas=3 --dry-run -o yaml > redis_node_affinity.yaml
```

2. 手写亲和性策略:

额 问题来了亲和性策略的字段我记不住啊，怎么办？那就需要通过

`kubectl explain RESOURCE [options]`来获取资源文档

怎么用?

比如我这里是要为pod做node的亲和性，那么一定是这个api接口下面的配置文档:想看pod的资源文档:

```
[root@k8s-ml ~]# kubectl explain pod.spec.affinity
KIND:     Pod
VERSION:  v1

RESOURCE: affinity <Object>

DESCRIPTION:
     If specified, the pod's scheduling constraints

     Affinity is a group of affinity scheduling rules.

FIELDS:
   nodeAffinity <Object>
     Describes node affinity scheduling rules for the pod.

   podAffinity  <Object>
     Describes pod affinity scheduling rules (e.g. co-locate this pod in the
     same node, zone, etc. as some other pod(s)).

   podAntiAffinity  <Object>
     Describes pod anti-affinity scheduling rules (e.g. avoid putting this pod
     in the same node, zone, etc. as some other pod(s)).

[root@k8s-ml ~]#
```

上面我们通过 `pod.spec.affinity` 定位到了`nodeAffinity` 文档,这些字段也是yaml种使用的字段，随后我通过一层一层的定位就大体上知道这些字段在yaml中是怎么使用的啦~

```
[root@k8s-ml ~]# kubectl explain
pods.spec.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms.matchExpressions
```

最后快速生成并且编辑的deployment yaml就写好了。

```
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    run: redis
  name: redis
spec:
  replicas: 3
  selector:
    matchLabels:
      run: redis
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        run: redis
    spec:
      containers:
      - image: redis
        name: redis
        resources: {}
      # 以下内容就是我通过explain参数来查询到的我想要的字段写的
      affinity:
        nodeAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            nodeSelectorTerms:
              - matchExpressions:
                  - key: kubernetes.io/hostname
                    operator: In
                    values:
                      - k8s-m1
status: {}
```

```
[root@k8s-m1 tmp]# kubectl get pods -o wide | grep redis
redis-57b6b69b77-c49sq   1/1       Running   0          40m       10.244.2.19   k8s-m1    <none>
redis-57b6b69b77-gdf27   1/1       Running   0          40m       10.244.2.20   k8s-m1    <none>
redis-57b6b69b77-ghzr6   1/1       Running   0          40m       10.244.2.21   k8s-m1    <none>
[root@k8s-m1 tmp]#
```

还是非常快速高效的。

最后更新时间：2018-12-19 22:39:54

该文链接：https://blog.sctux.com/2018/12/09/k8s-yaml-bian-xie-xiao-ji-qiao/