

# K8S 常见命令100个汇总

---

## 集群信息：

1. 显示 Kubernetes 版本: `kubectl version`
2. 显示集群信息: `kubectl cluster-info`
3. 列出集群中的所有节点: `kubectl get nodes`
4. 查看一个具体的节点详情: `kubectl describe node <node-name>`
5. 列出所有命名空间: `kubectl get namespaces`
6. 列出所有命名空间中的所有 pod: `kubectl get pods --all-namespaces`

## Pod 诊断：

1. 列出特定命名空间中的 pod:  
`kubectl get pods -n <namespace>`
2. 查看一个 Pod 详情:  
`kubectl describe pod <pod-name> -n <namespace>`
3. 查看 Pod 日志:  
`kubectl logs <pod-name> -n <namespace>`
4. 尾部 Pod 日志:  
`kubectl logs -f <pod-name> -n <namespace>`
5. 在 pod 中执行命令:  
`kubectl exec -it <pod-name> -n <namespace> -- <command>`

## Pod 健康检查：

1. 检查 Pod 准备情况:  
`kubectl get pods <pod-name> -n <namespace> -o jsonpath='{.status.conditions[?(@.type=="Ready")].status}'`
2. 检查 Pod 事件:  
`kubectl get events -n <namespace> --field-selector involvedObject.name=<pod-name>`

## Service诊断：

1. 列出命名空间中的所有服务:  
`kubectl get svc -n <namespace>`
2. 查看一个服务详情:  
`kubectl describe svc <service-name> -n <namespace>`

## Deployment诊断：

1. 列出命名空间中的所有Deployment:

```
kubectl get deployments -n <namespace>
```

2. 查看一个Deployment详情:

```
kubectl describe deployment <deployment-name> -n <namespace>
```

3. 查看滚动发布状态:

```
kubectl rollout status deployment/<deployment-name> -n <namespace>
```

4. 查看滚动发布历史记录:

```
kubectl rollout history deployment/<deployment-name> -n <namespace>
```

## StatefulSet诊断:

1. 列出命名空间中的所有 StatefulSet:

```
kubectl get statefulsets -n <namespace>
```

2. 查看一个 StatefulSet详情:

```
kubectl describe statefulset <statefulset-name> -n <namespace>
```

## ConfigMap 和Secret诊断:

1. 列出命名空间中的 ConfigMap:

```
kubectl get configmaps -n <namespace>
```

2. 查看一个ConfigMap详情:

```
kubectl describe configmap <configmap-name> -n <namespace>
```

3. 列出命名空间中的 Secret:

```
kubectl get secrets -n <namespace>
```

4. 查看一个Secret详情:

```
kubectl describe secret <secret-name> -n <namespace>
```

## 命名空间诊断:

1. 查看一个命名空间详情:

```
kubectl describe namespace <namespace-name>
```

## 资源使用情况:

1. 检查 pod 的资源使用情况:

```
kubectl top pod <pod-name> -n <namespace>
```

2. 检查节点资源使用情况:

```
kubectl top nodes
```

## 网络诊断:

1. 显示命名空间中 Pod 的 IP 地址:

```
kubectl get pods -n <namespace> -o custom-columns=POD:metadata.name,IP:status.podIP --no-headers
```

2. 列出命名空间中的所有网络策略:

```
kubectl get networkpolicies -n <namespace>
```

3. 查看一个网络策略详情:

```
kubectl describe networkpolicy <network-policy-name> -n <namespace>
```

## 持久卷 (PV) 和持久卷声明 (PVC) 诊断:

1. 列出PV: `kubectl get pv`

2. 查看一个PV详情: `kubectl describe pv <pv-name>`

3. 列出命名空间中的 PVC: `kubectl get pvc -n <namespace>`

4. 查看PVC详情: `kubectl describe pvc <pvc-name> -n <namespace>`

## 节点诊断:

1. 获取特定节点上运行的 Pod 列表:

```
kubectl get pods --field-selector spec.nodeName=<node-name> -n <namespace>
```

## 资源配额和限制:

1. 列出命名空间中的资源配额:

```
kubectl get resourcequotas -n <namespace>
```

2. 查看一个资源配额详情:

```
kubectl describe resourcequota <resource-quota-name> -n <namespace>
```

## 自定义资源定义 (CRD) 诊断:

1. 列出命名空间中的自定义资源:

```
kubectl get <custom-resource-name> -n <namespace>
```

2. 查看自定义资源详情:

```
kubectl describe <custom-resource-name> <custom-resource-instance-name> -n <namespace>
```

## 资源伸缩和自动伸缩

1. Deployment伸缩:

```
kubectl scale deployment <deployment-name> --replicas=<replica-count> -n <namespace>
```

2. 设置Deployment的自动伸缩:

```
kubectl autoscale deployment <deployment-name> --min=<min-pods> --max=<max-pods> --cpu-percent=<cpu-percent> -n <namespace>
```

3. 检查水平伸缩器状态:

```
kubectl get hpa -n <namespace>
```

## 容量诊断

1. 列出按容量排序的持久卷 (PV):

```
kubectl get pv --sort-by=.spec.capacity.storage
```

2. 查看PV回收策略:

```
kubectl get pv <pv-name> -o=jsonpath='{.spec.persistentVolumeReclaimPolicy}'
```

3. 列出所有存储类别:

```
kubectl get storageclasses
```

## Pod 网络故障排除:

1. 运行网络诊断 Pod (例如 busybox) 进行调试:

```
kubectl run -it --rm --restart=Never --image=busybox net-debug-pod -- /bin/sh
```

2. 测试从 Pod 到特定端点的连接:

```
kubectl exec -it <pod-name> -n <namespace> -- curl <endpoint-url>
```

3. 跟踪从一个 Pod 到另一个 Pod 的网络路径:

```
kubectl exec -it <source-pod-name> -n <namespace> -- traceroute <destination-pod-ip>
```

4. 检查 Pod 的 DNS 解析:

```
kubectl exec -it <pod-name> -n <namespace> -- nslookup <domain-name>
```

## 节点污点:

1. 列出节点污点: `kubectl describe node <node-name> | grep Taints`