

# 知 CloudOS KaaS容器平台Pod应用使用DNS解析域名方法

管理集群 王书琦 2022-01-09 发表

## 组网及说明

CloudOS E5133版本, 创建KaaS容器集群, K8S版本1.19.10

名称	状态	版本	工作节点数(可用/总数)	CPU总量	内存总量
kube.gxtv	运行中	1.19.10	6/6	216.0	424.26G

DNS服务器为10.200.8.21

要实现Pod内部可以通过域名来访问其他业务

## 配置步骤

1.修改KaaS集群每台主机的/etc/resolv.conf, 包括master和worker

```
[root@moc-master-764e9 ~]# cat /etc/resolv.conf
; Created by cloud-init on instance boot automatically, do not edit.
:
nameserver 10.200.8.21
[root@moc-master-764e9 ~]#
```

2.重启coredns/nodelocaldns容器

```
kubectl get pod -n kube-system |grep -E "coredns|nodelocaldns"
```

```
kubectl delete pod -n kube-system coredns-* (此处*内容根据现场信息替换)
```

```
kubectl delete pod -n kube-system nodelocaldns-* (此处*内容根据现场信息替换)
```

以下每个pod都需要重启

```
[root@moc-master-764e9 ~]# kubectl get pod -n kube-system |grep -E "coredns|nodelocaldns"
coredns-55b9d78db5-6qww5      1/1      Running    0          18h
coredns-55b9d78db5-xnqd8      1/1      Running    0          18h
nodelocaldns-4ns82            1/1      Running    0          18h
nodelocaldns-527js            1/1      Running    0          18h
nodelocaldns-6nfk6            1/1      Running    0          18h
nodelocaldns-9nr5g            1/1      Running    0          18h
nodelocaldns-krkxz            1/1      Running    0          18h
nodelocaldns-lvdjl            1/1      Running    0          18h
nodelocaldns-nqdlr            1/1      Running    0          18h
nodelocaldns-r4jqw            1/1      Running    0          18h
nodelocaldns-vmftm            1/1      Running    0          18h
```

3.进入Pod内部验证

```
[root@moc-master-764e9 ~]# kubectl get pod -n minio
NAME          READY   STATUS    RESTARTS   AGE
minio-6c9f757464-77h2p  1/1     Running   0          20h
minio-6c9f757464-g9zrn  1/1     Running   0          20h
[root@moc-master-764e9 ~]# kubectl exec -itn minio minio-6c9f757464-77h2p bash
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.
[root@minio-6c9f757464-77h2p ~]# ping x10000.mlx.cn
PING x10000.mlx.cn (10.200.9.20) 56(84) bytes of data:
64 bytes from 10.200.9.20 (10.200.9.20): icmp_seq=1 ttl=62 time=0.360 ms
64 bytes from 10.200.9.20 (10.200.9.20): icmp_seq=2 ttl=62 time=0.200 ms
64 bytes from 10.200.9.20 (10.200.9.20): icmp_seq=3 ttl=62 time=0.115 ms
64 bytes from 10.200.9.20 (10.200.9.20): icmp_seq=4 ttl=62 time=0.145 ms
64 bytes from 10.200.9.20 (10.200.9.20): icmp_seq=5 ttl=62 time=2.56 ms
^C
--- x10000.mlx.cn ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4058ms
rtt min/avg/max/mdev = 0.115/0.676/2.564/0.947 ms
[root@minio-6c9f757464-77h2p ~]# ping moc.mlx.cn
PING moc.mlx.cn (10.200.8.11) 56(84) bytes of data:
64 bytes from 10.200.8.11.prometheus-prometheus-oper-kube-proxy.kube-system.svc.kube.gxtv (10.200.8.11): icmp_seq=1 ttl=63 time=0.595 ms
64 bytes from 10.200.8.11.prometheus-prometheus-oper-kube-proxy.kube-system.svc.kube.gxtv (10.200.8.11): icmp_seq=2 ttl=63 time=0.377 ms
64 bytes from 10.200.8.11.prometheus-prometheus-oper-kube-proxy.kube-system.svc.kube.gxtv (10.200.8.11): icmp_seq=3 ttl=63 time=0.177 ms
64 bytes from 10.200.8.11.prometheus-prometheus-oper-kube-proxy.kube-system.svc.kube.gxtv (10.200.8.11): icmp_seq=4 ttl=63 time=0.194 ms
^C
--- moc.mlx.cn ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3047ms
rtt min/avg/max/mdev = 0.177/0.335/0.595/0.170 ms
```

## 配置关键点

K8S集群内部使用coredns/nodelocaldns来解析域名，在coredns/nodelocaldns的cm配置中可以看到域名解析都转发到host的本地/etc/resolv.conf文件。

注意：修改resolv.conf配置后需要重启所有coredns/nodelocaldns POD

kubectl edit cm coredns -n kube-system

```
#
apiVersion: v1
data:
  Corefile: |
    .:53 {
      errors
      health {
        lameduck 5s
      }
      ready
      kubernetes kube.gxtv in-addr.arpa ip6.arpa {
        pods insecure
        fallthrough in-addr.arpa ip6.arpa
      }
      prometheus :9153
      forward . /etc/resolv.conf {
        prefer_udp
      }
      cache 30
      loop
      reload
      loadbalance
    }
kind: ConfigMap
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"v1","data":{"Corefile":":53 {\n  errors\n  fallthrough in-addr.arpa ip6.arpa\n  }\n  prometheus :9153\n  forward . /etc/resolv.conf {\n    prefer_udp\n  }\n  cache 30\n  loop\n  reload\n  loadbalance\n  }\n  kind: \"ConfigMap\", \"metadata\": {\"annotations\": {\"kubernetes.io/last-applied-configuration\": \"\n  creationTimestamp: \"2022-01-08T05:50:21Z\"\n  labels: {\n    addonmanager.kubernetes.io/mode: EnsureExists\n  }\n  name: coredns\n  namespace: kube-system\n  resourceVersion: \"34284\""}
kind: "ConfigMap", "metadata": {"annotations": {"kubernetes.io/last-applied-configuration": "2022-01-08T05:50:21Z"}, "labels": {"addonmanager.kubernetes.io/mode": "EnsureExists"}, "name": "coredns", "namespace": "kube-system", "resourceVersion": "34284"}
creationTimestamp: "2022-01-08T05:50:21Z"
labels:
  addonmanager.kubernetes.io/mode: EnsureExists
name: coredns
namespace: kube-system
resourceVersion: "34284"
```

kubectl edit cm nodelocaldns -n kube-system

```
    forward . 10.100.0.3 {
      force_tcp
    }
    prometheus :9253
  }
  ip6.arpa:53 {
    errors
    cache 30
    reload
    loop
    bind 169.254.25.10
    forward . 10.100.0.3 {
      force_tcp
    }
    prometheus :9253
  }
  .:53 {
    errors
    cache 30
    reload
    loop
    bind 169.254.25.10
    forward . /etc/resolv.conf
    prometheus :9253
  }
}
kind: ConfigMap
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"v1","data":{"Corefile":"kube.gxtv:5
      forward . 10.100.0.3 {\n      force_tcp\n      }\n
      54.25.10\n      forward . 10.100.0.3 {\n      force_tcp\
      10.100.0.3 {\n      force_tcp\n      }\n      prometheus
      eus :9253\n}\n"}, "kind": "ConfigMap", "metadata": {"annotat
      creationTimestamp: "2022-01-08T05:50:24Z"

```