Kubernetes的网络实践

王炳燊(溪恒)

阿里云容器服务

Kubernetes网络

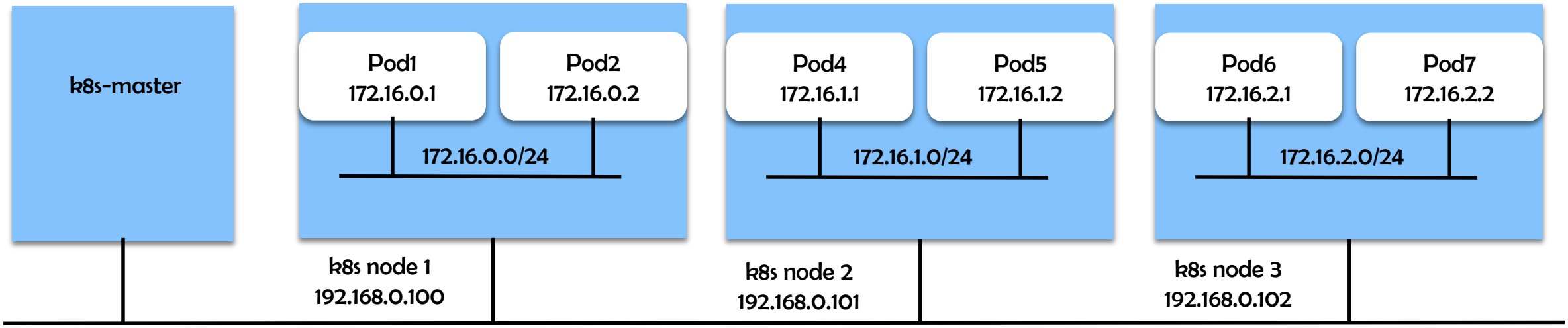
- 网络连通性(CNI) *
- 负载均衡(Service，Ingress…)
- 服务发现(DNS…)
- Service Mesh(Ingress, Istio…)

目录

- Kubernetes网络结构
- 阿里云上的场景和挑战
- 高性能网络组件Terway

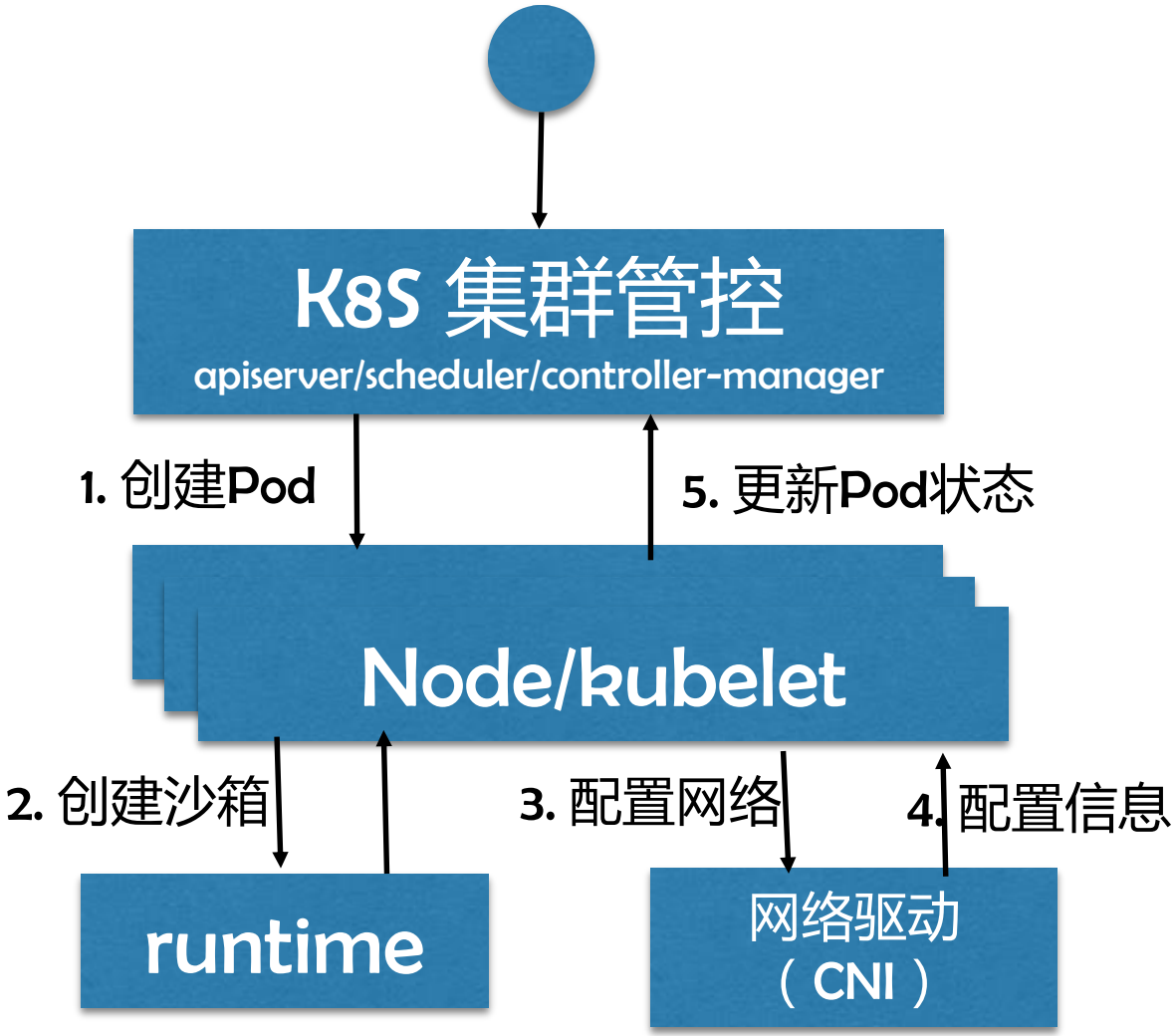
Kubernetes网络

- Pod有自己独立的网络空间和IP地址
- Pod可以通过各自的IP地址互相访问



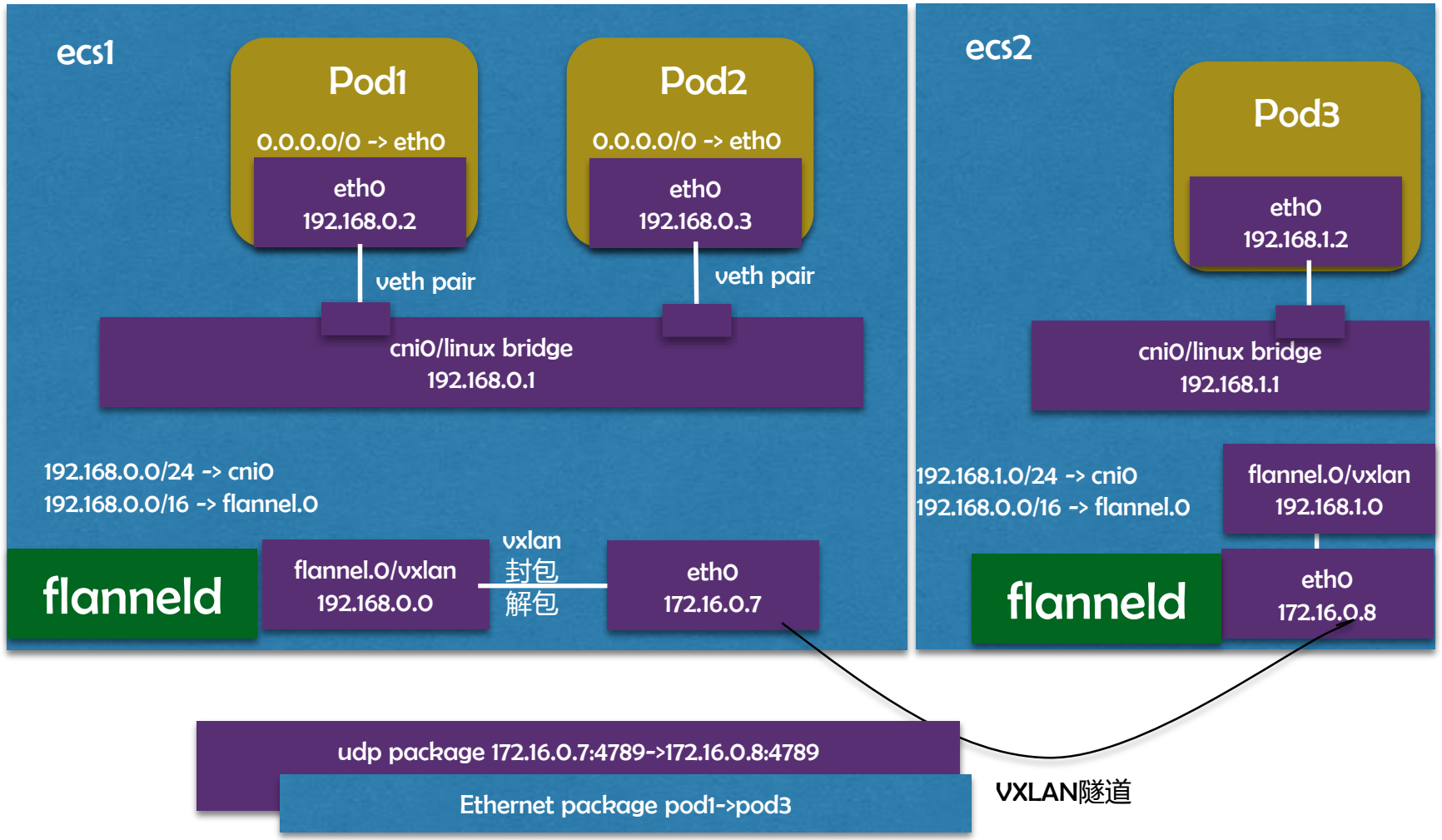
Kubernetes网络

- Kubernetes使用CNI接口调用网络插件
- 网络插件(CNI)能分配且唯一的IP地址
- 网络插件(CNI)可以配置Pod的网络和打通Pods之间的访问



Flannel-vxlan的实现方式

- 不同主机上的pod在不同网段，每个节点独立分配，保证IP唯一
- 跨主机的容器通信通过vxlan封装

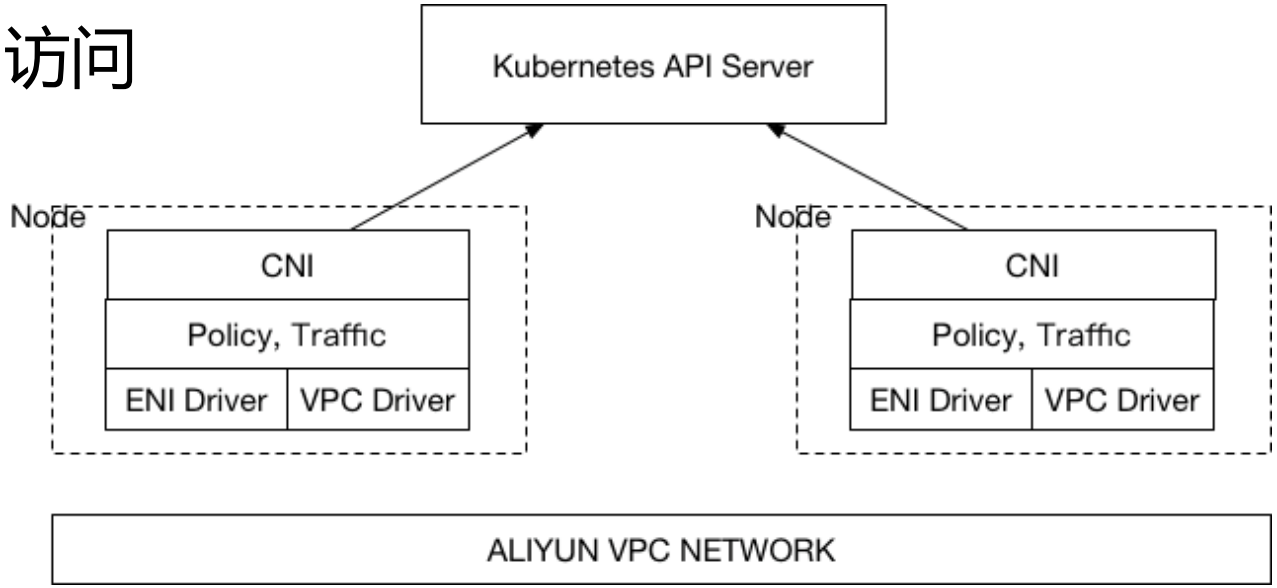


我们遇到的挑战

- 游戏行业/机器学习/基因计算 等场景中对性能有极致要求
  - 封包解包会带来大量性能损失
  - Linux网络栈处理能力有限
- 客户生产集群的应用要做网络的访问控制
  - 生产/预发/测试 环境做网络隔离
  - 多租户隔离
- 应用网络的流量需要做控制
  - 应用占用过多节点带宽导致节点离线
  - 不同应用避免相互影响

高性能Terway网络组件

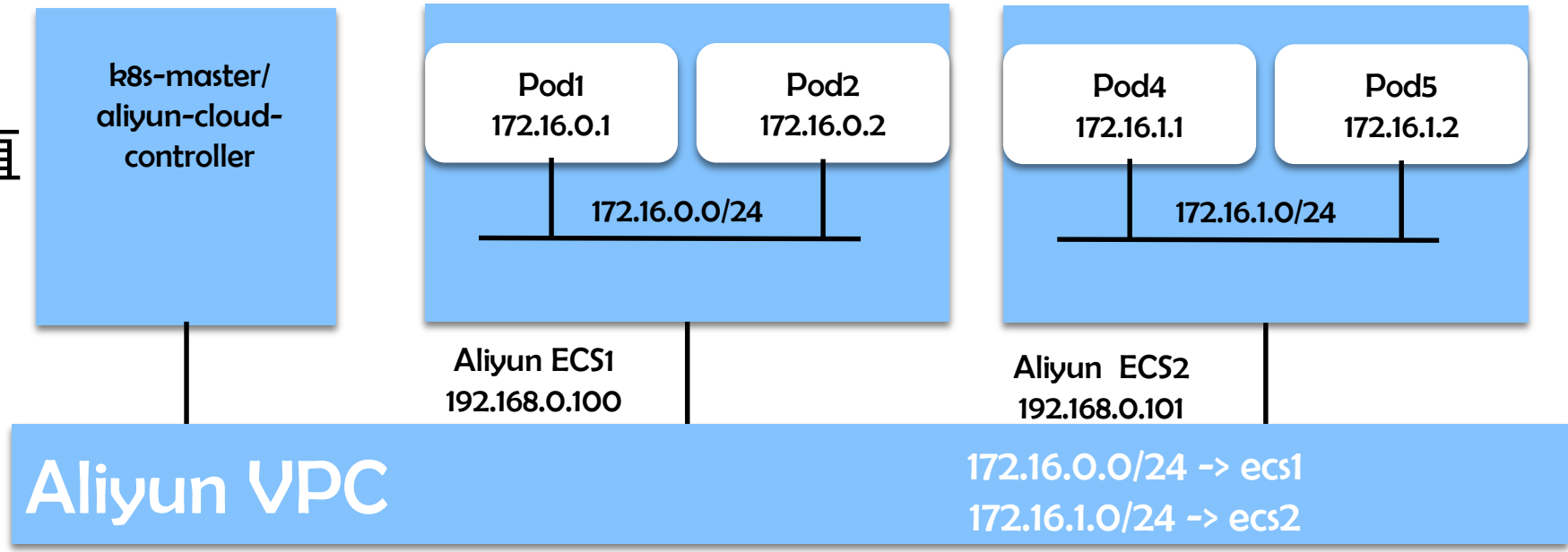
- 同时支持VPC和ENI两种方式打通容器网络
- 支持Network Policy来控制应用访问
- 支持Qos流控来限制应用带宽



https://github.com/AliyunContainerService/terway

Terway网络连通

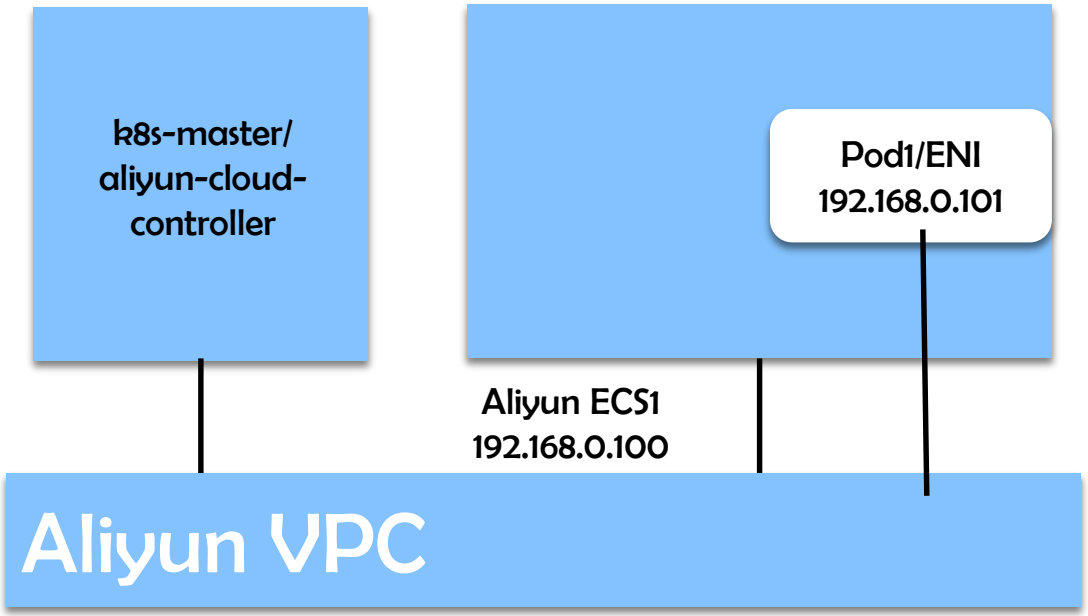
- 基于阿里云VPC的路由转发能力，不再需要封包解包
- 支持与VPC中其他资源直通
- 性能提升20%(相对flannel vxlan)



Terway网络连通

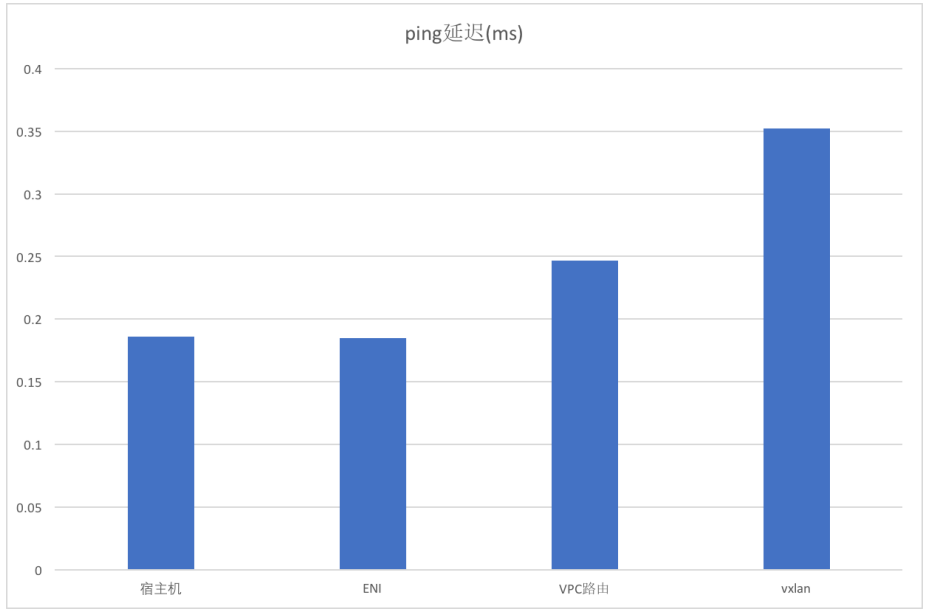
- 直接使用ECS的弹性网卡，不依赖于linux的虚拟网络设备
- 热插拔弹性网卡直接绑定给容器
- 通过k8s Device plugin方式跟k8s调度结合
- 性能相对宿主机无任何损失
- 结合神龙服务器可达到万兆容器网络

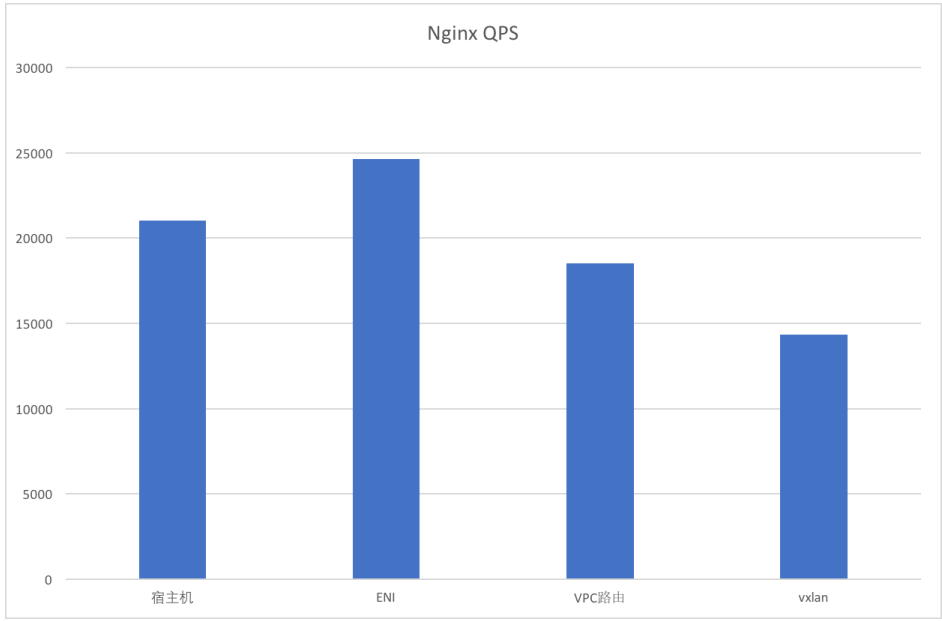
```
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image: nginx
    resources:
      limits:
        aliyun/eni: 1
```



Terway性能数据

- 性能对比

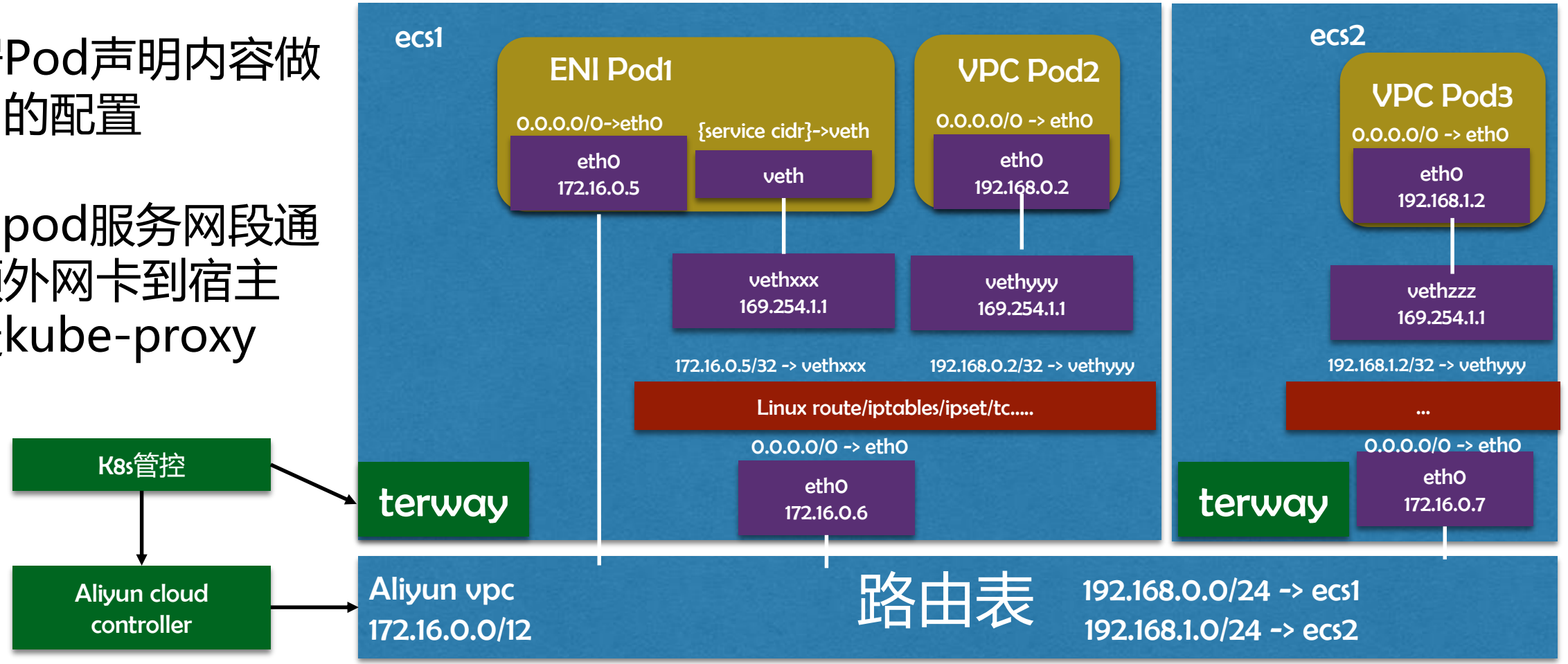




- 支持裸金属和RDMA的高速网络

Terway网络原理

- 根据Pod声明内容做不同的配置
- ENI pod服务网段通过额外网卡到宿主机走kube-proxy



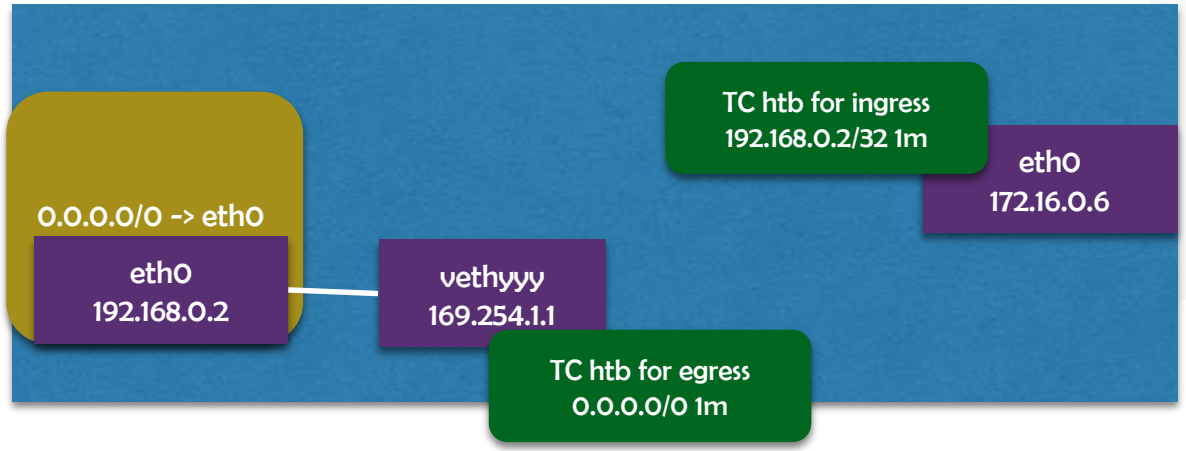
Terway Network Policy

- 使用kubernetes标准的Network Policy配置
- 用来控制容器网络的访问策略
  - 出入流量
  - 网段/pod label/namespace
- Terway中实现：
  - 监听apiserver network policy
  - 动态更新宿主机iptables+ipset

```yaml
kind: NetworkPolicy
apiVersion: networking.k8s.io/v1
metadata:
  name: access-nginx
  namespace: default
spec:
  podSelector:
    matchLabels:
      run: nginx
  ingress:
  - from:
    - podSelector:
        matchLabels:
          access: "true"
```

Terway Pod Qos

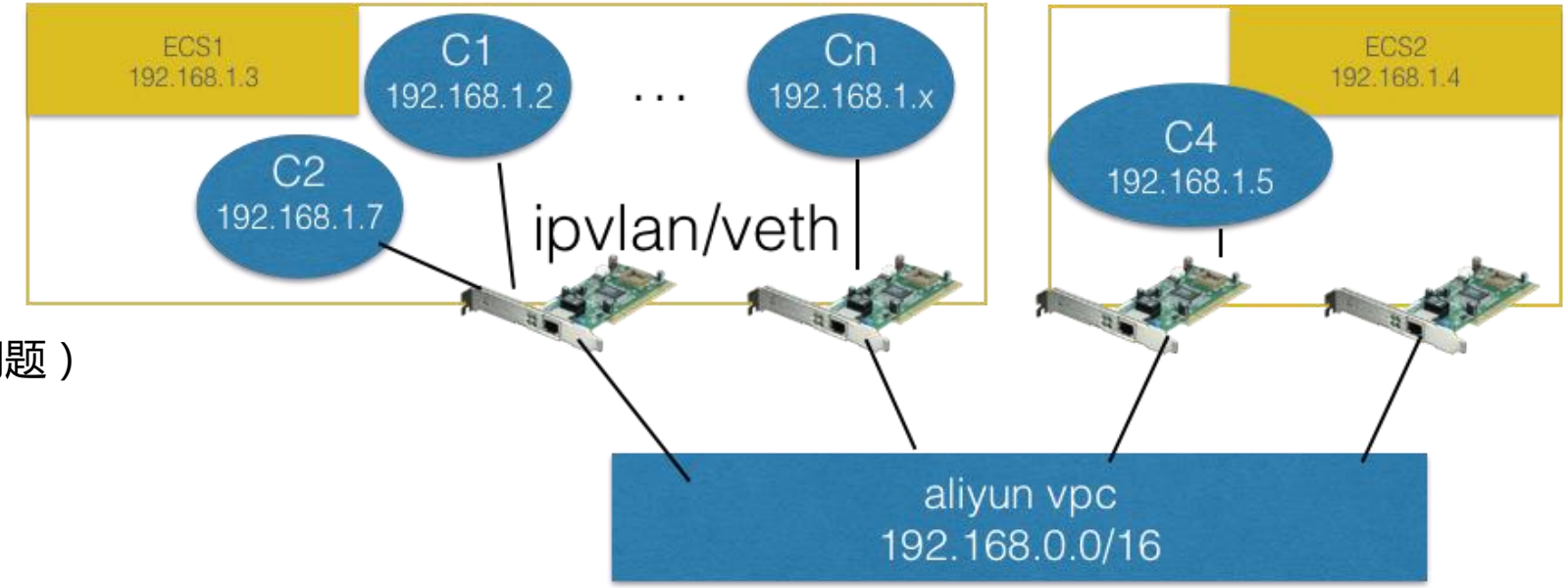
- 同一个节点Pod共用节点带宽
  - 避免Pod耗尽带宽导致节点离线
  - 避免Pod间互相影响




```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  annotations:
    k8s.aliyun.com/ingress-bandwidth: 1m
    k8s.aliyun.com/egress-bandwidth: 1m
spec:
  nodeSelector:
    kubernetes.io/hostname: cn-shanghai.i-uf63p6s96kf4jfh8wpwn
  containers:
  - name: nginx
    image: nginx:1.7.9
    ports:
    - containerPort: 80
```

Terway后续工作

- 结合弹性网卡即将支持的多IP能力
  - 把弹性网卡上的IP地址分配给容器
  - 不在依赖 VPC路由方式（性能提升）
  - VPC路由表受限（路由模式规模问题）
  - 解决弹性网卡数量有限（ENI模式密度问题）



谢谢！
Q & A

K8S 社区钉钉大群

容器服务团队博客

terway项目地址：
github.com/AliyunContainerService/terway


MORE THAN JUST CLOUD
Alibaba Cloud