Hard core AKS

Alessandro Melchiori

@amelchiori

Founder & software developer @ CodicePlastico

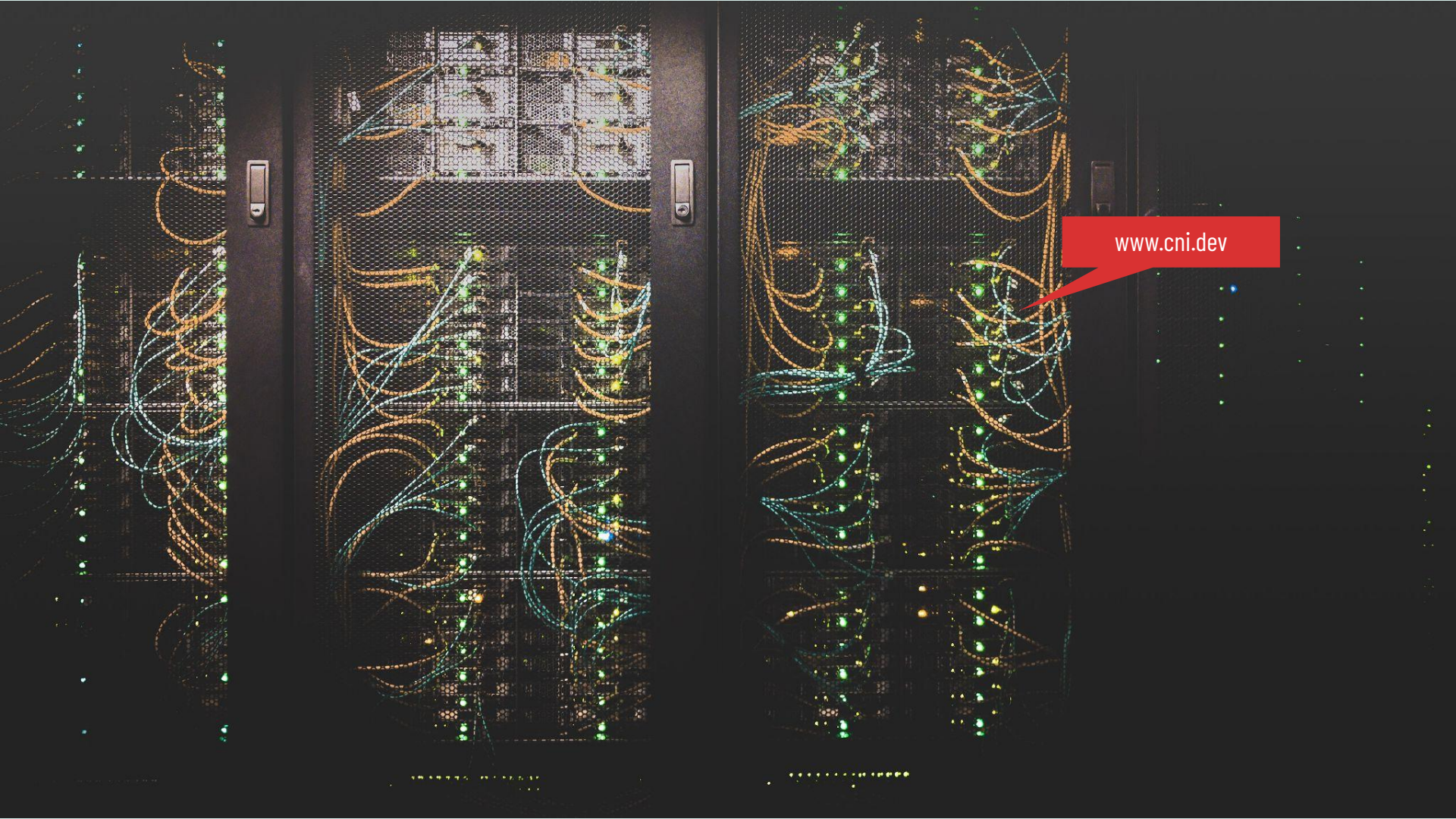
CommunityDays.it

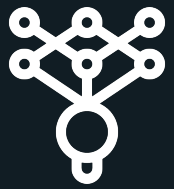
Let's start!

```
az aks create \
   --resource-group demo-rg \
   --name demo-cluster
```

Network Plugin

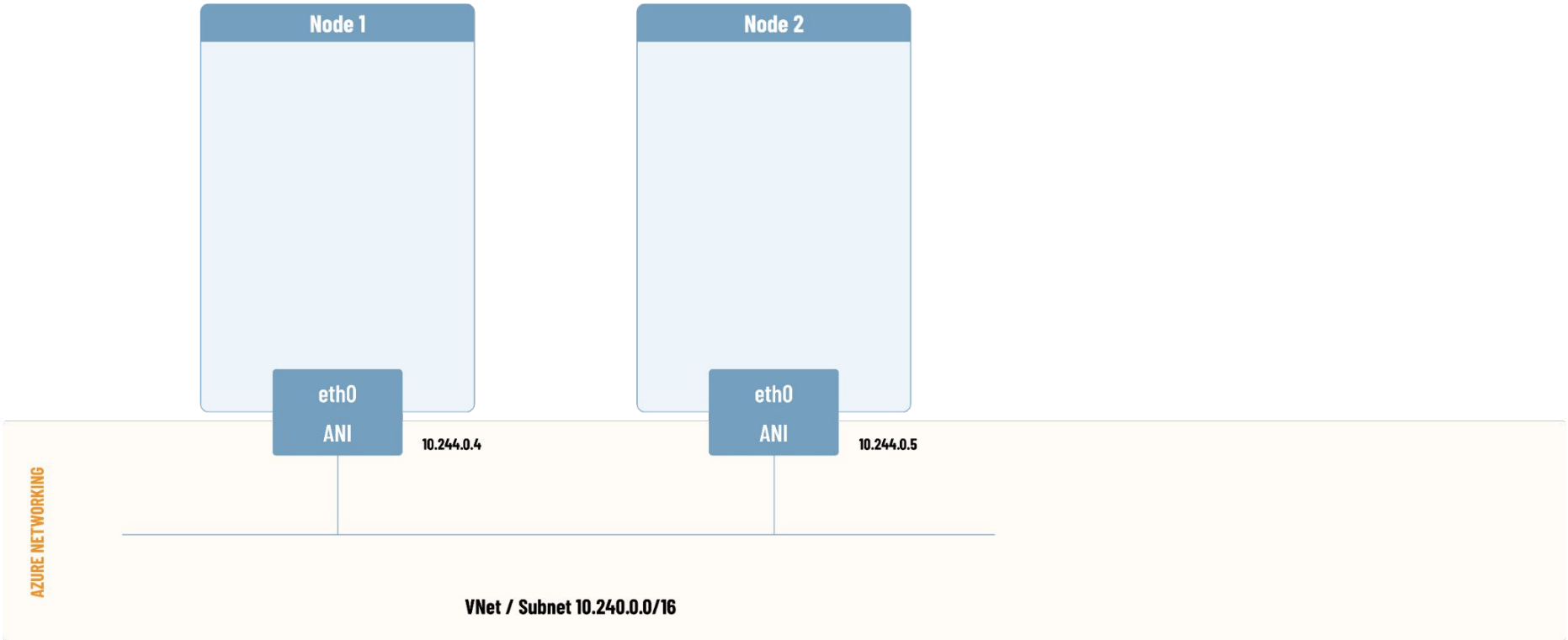
```
az aks create \
   --resource-group demo-rg \
   --name demo-cluster \
   --network-plugin {kubenet, azure}
```

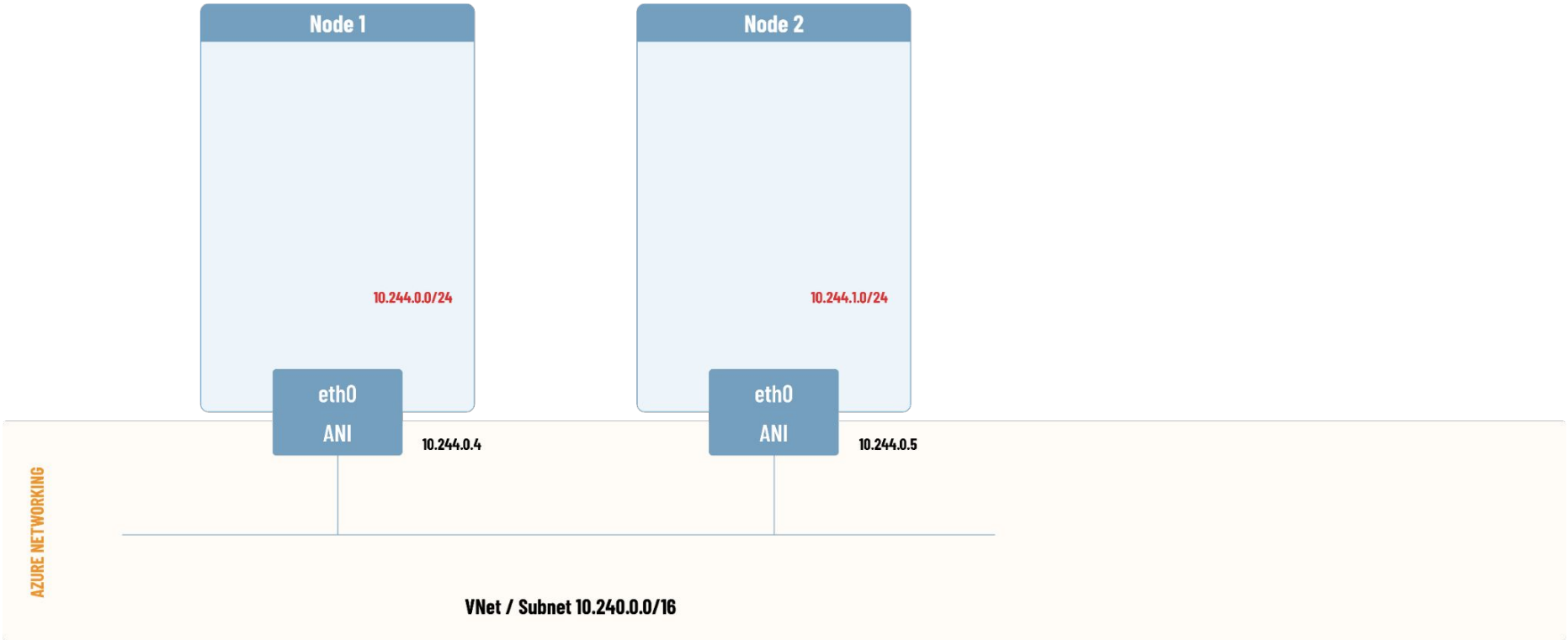

www.cni.dev

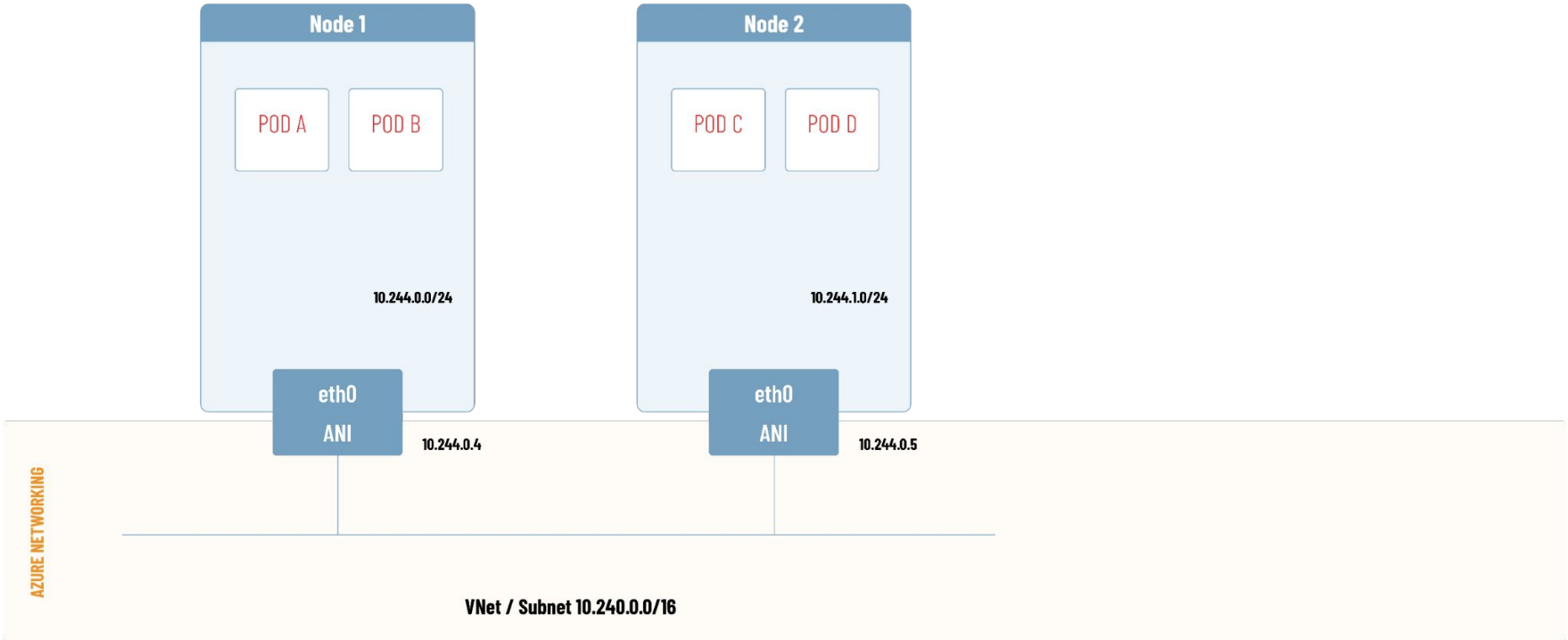


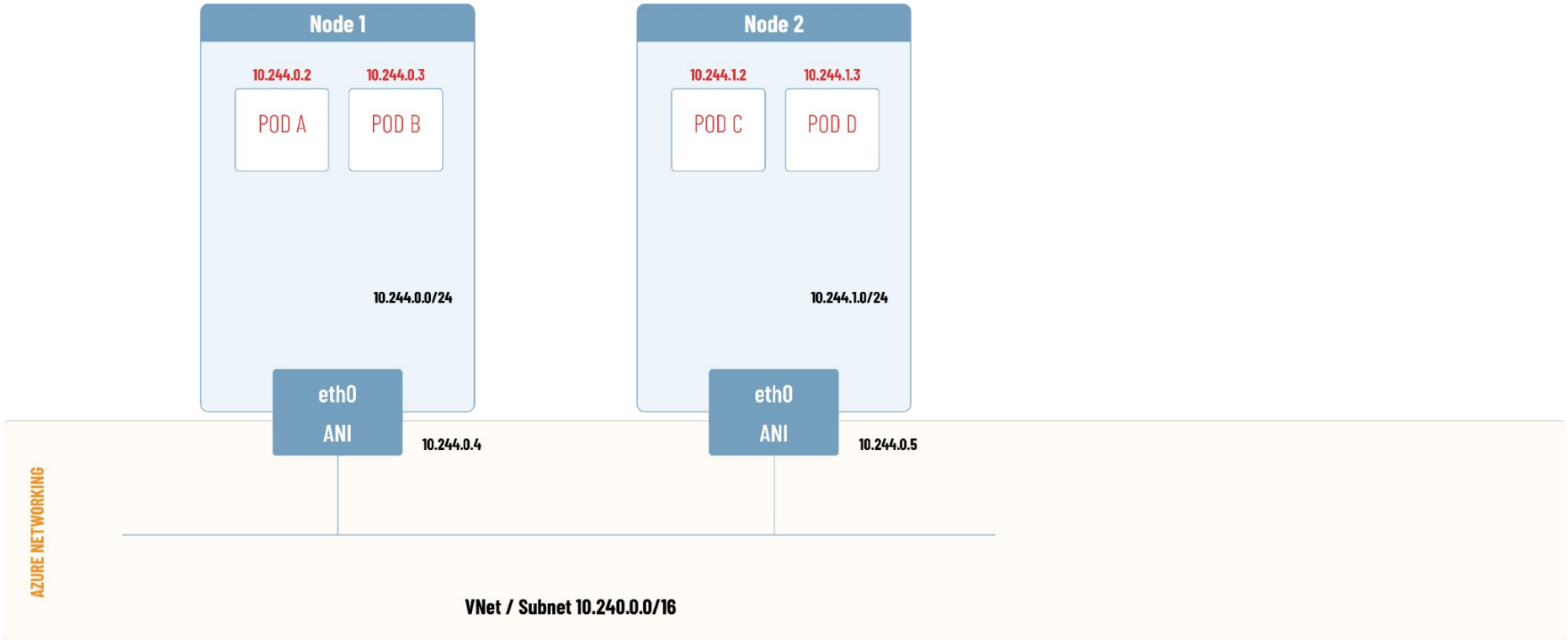
Basic networking

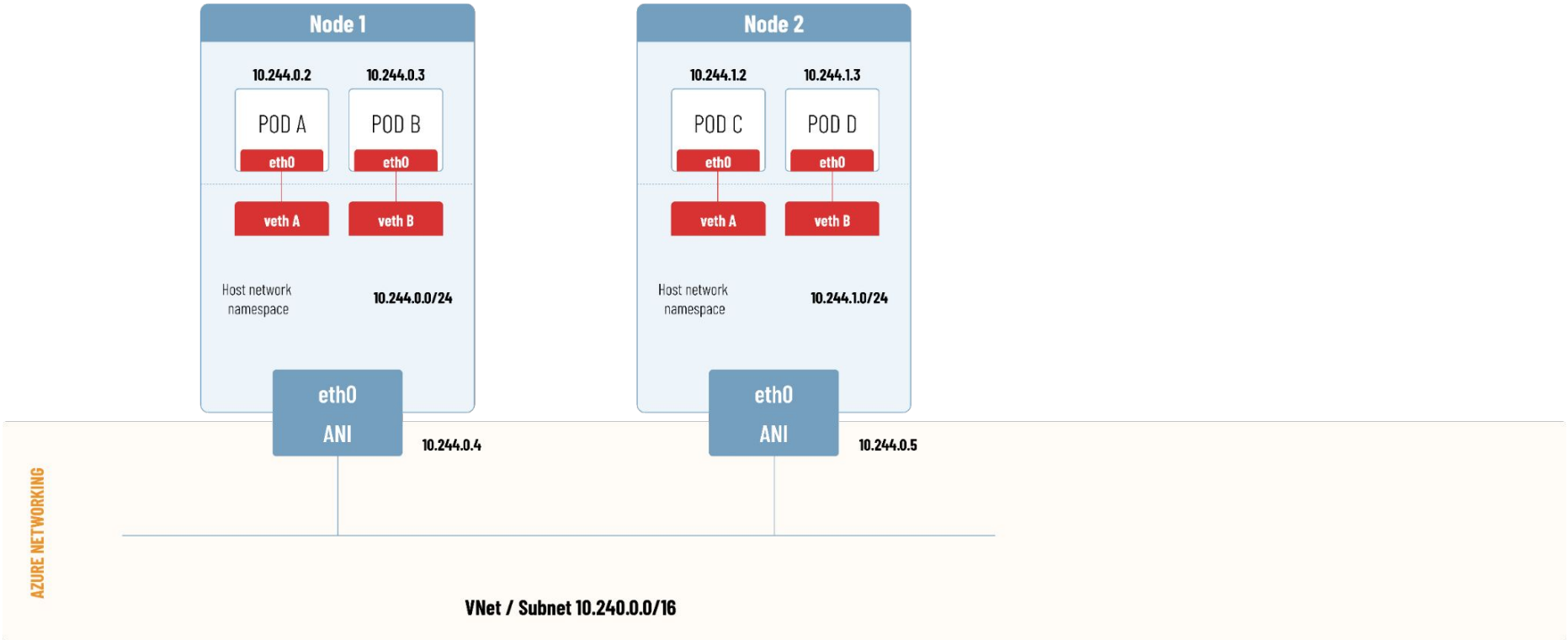
kubenet

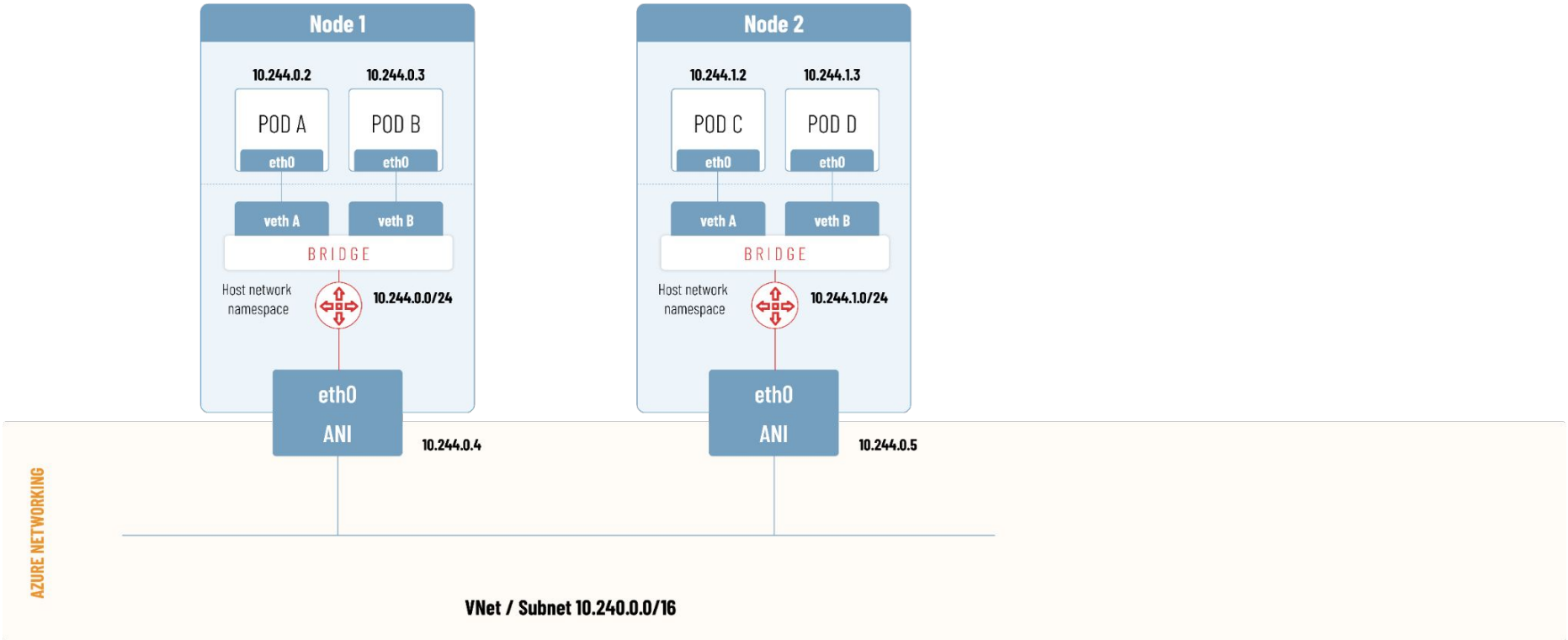

Node 1
Node 2
eth0
ANI
10.244.0.4
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

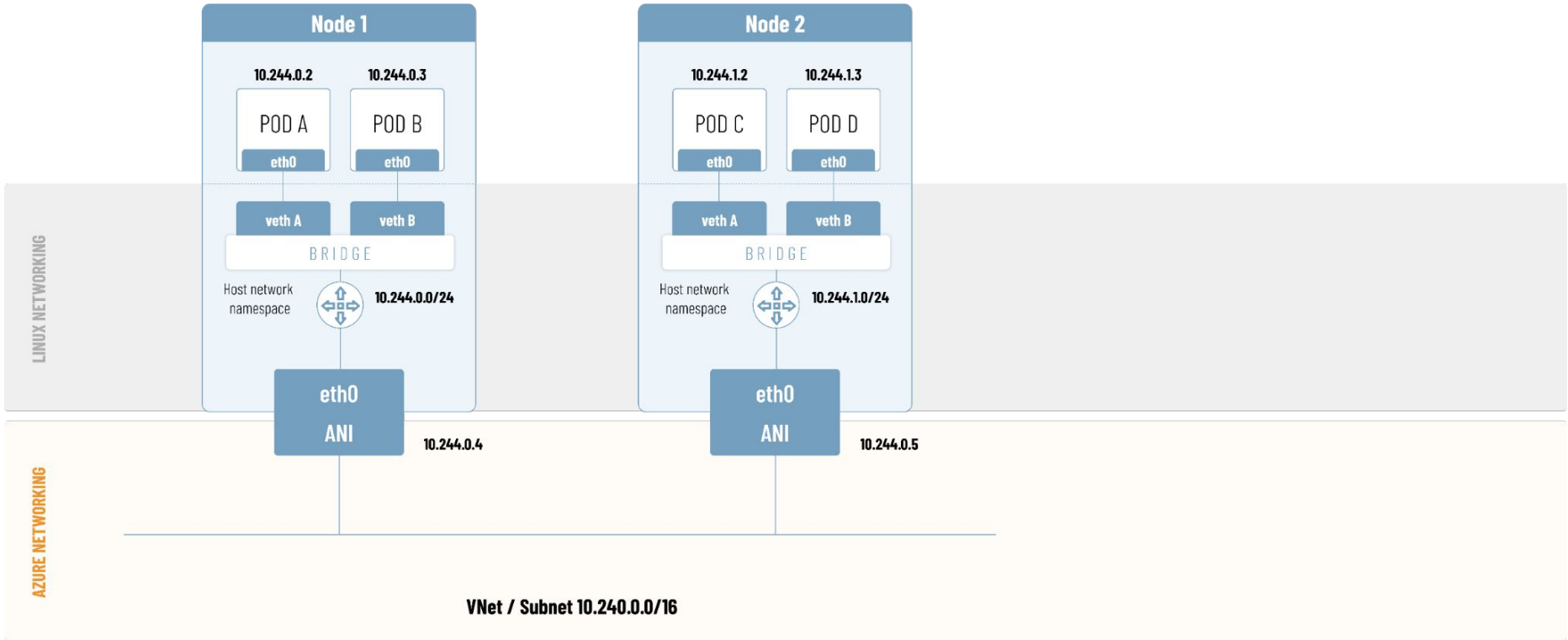

Node 1
Node 2
10.244.0.0/24
10.244.1.0/24
eth0
ANI
10.244.0.4
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

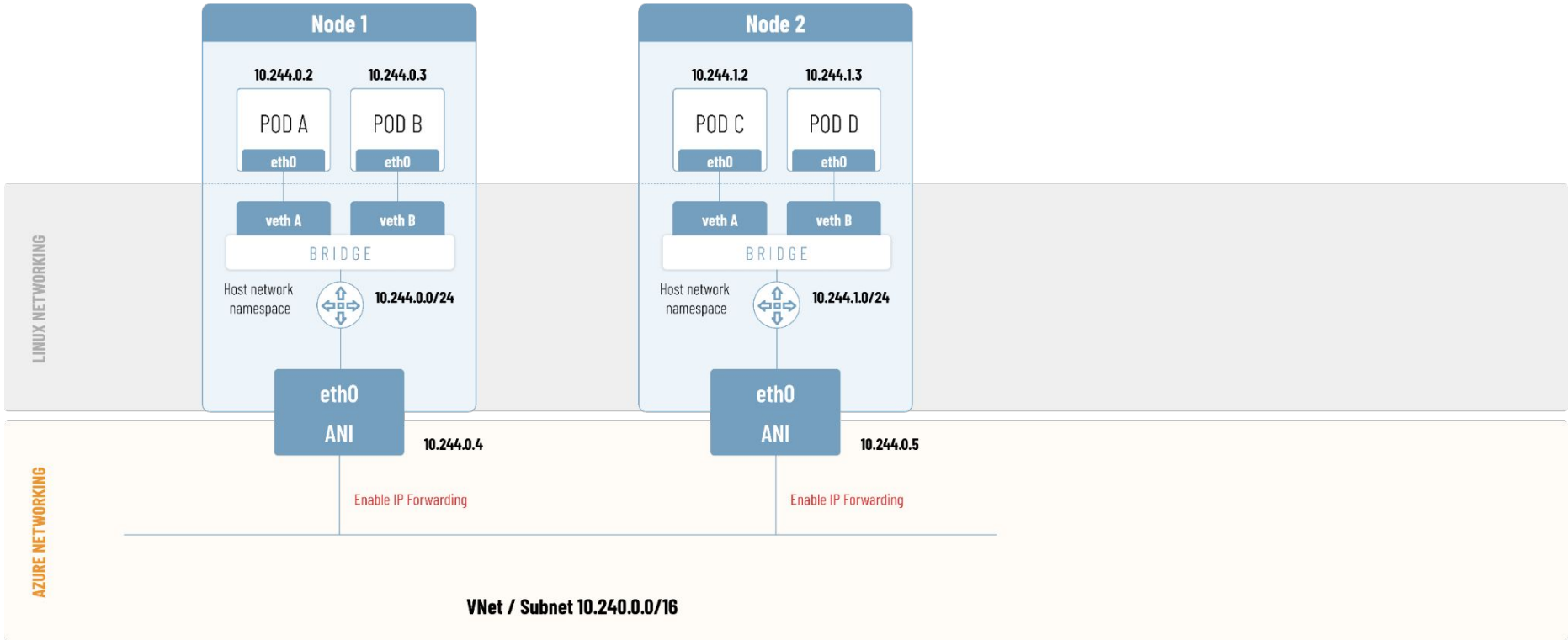

Node 1
POD A
POD B
10.244.0.0/24
eth0
ANI
10.244.0.4
Node 2
POD C
POD D
10.244.1.0/24
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

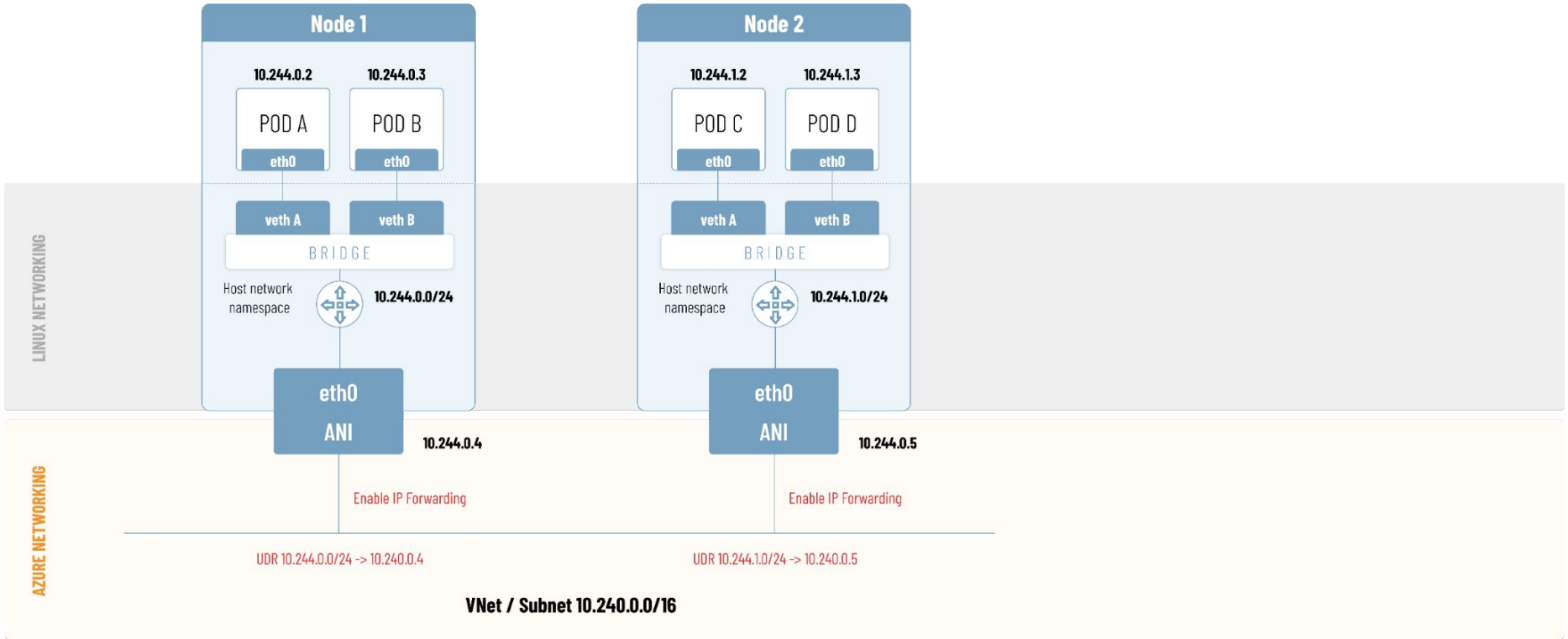

Node 1
10.244.0.2
10.244.0.3
POD A
POD B
10.244.0.0/24
eth0
ANI
10.244.0.4
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
10.244.1.0/24
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

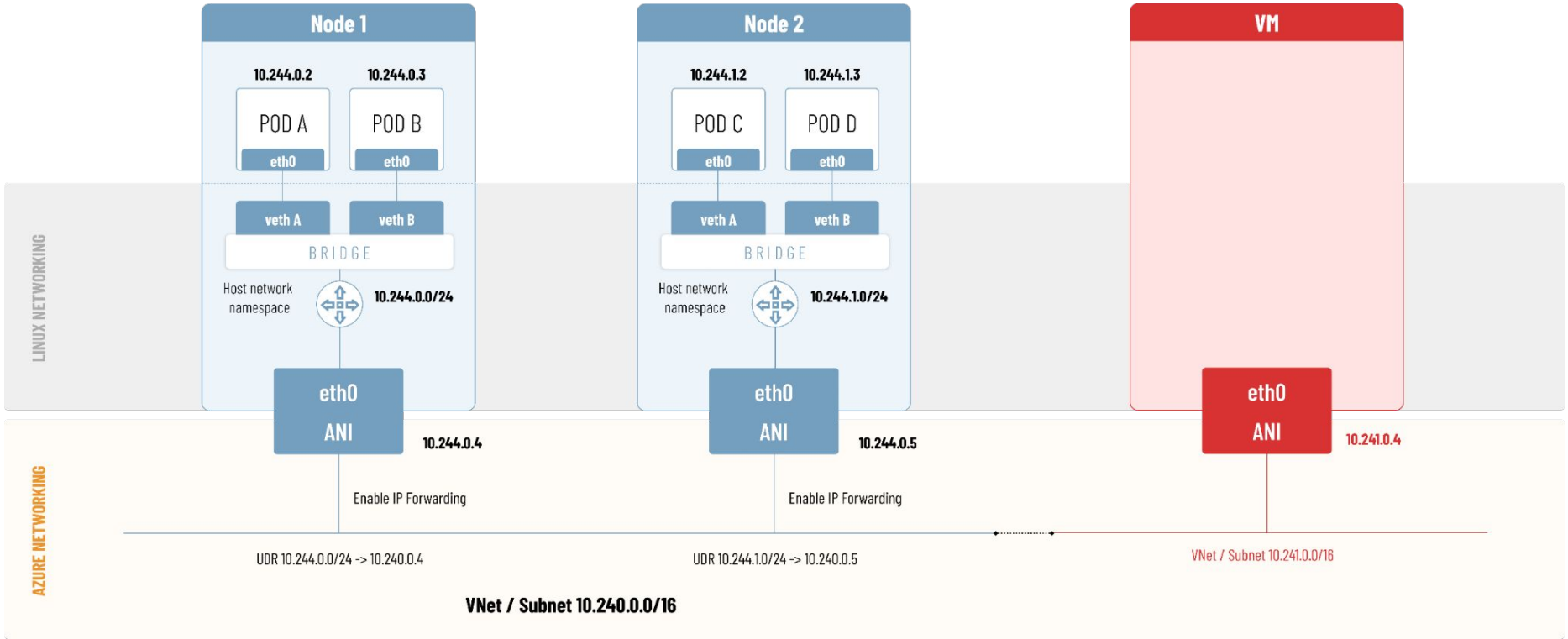

Node 1
10.244.0.2
10.244.0.3
POD A
POD B
eth0
eth0
veth A
veth B
Host network namespace
10.244.0.0/24
eth0
ANI
10.244.0.4
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
eth0
eth0
veth A
veth B
Host network namespace
10.244.1.0/24
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

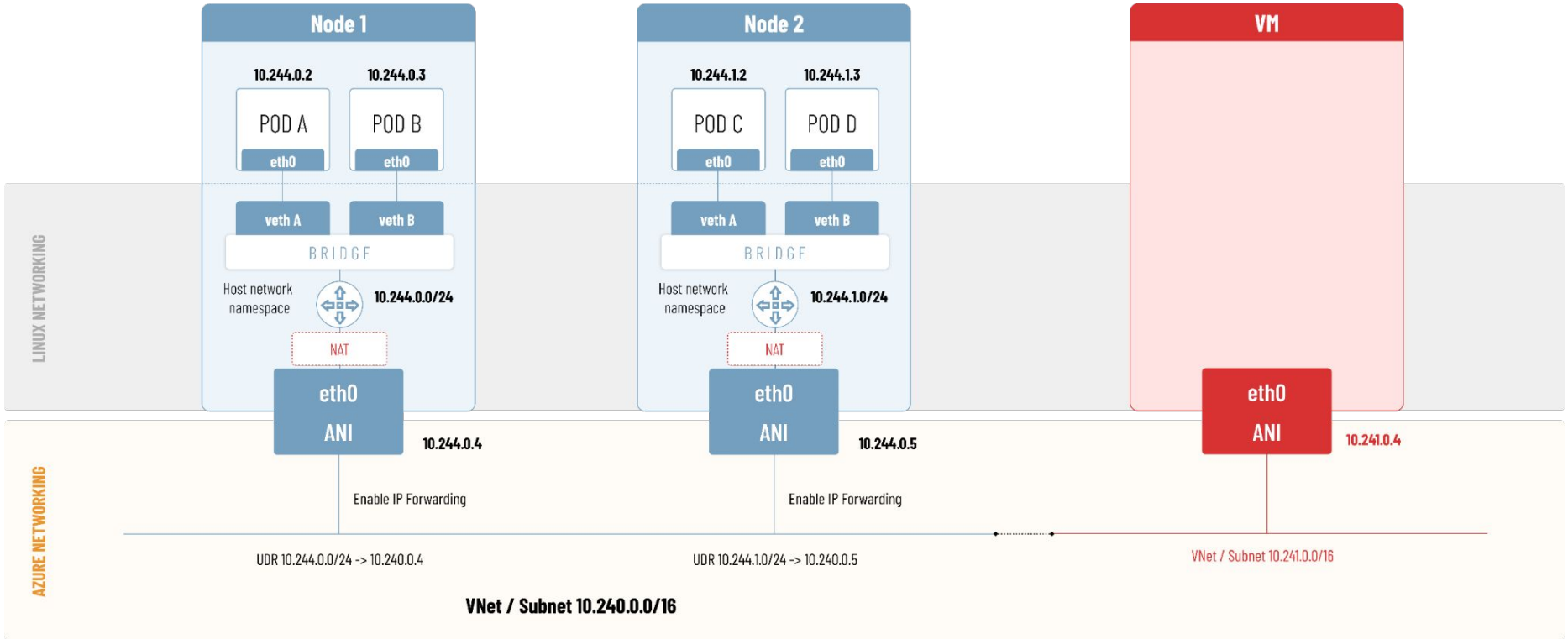

Node 1
10.244.0.2
10.244.0.3
POD A
POD B
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.0.0/24
eth0
ANI
10.244.0.4
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.1.0/24
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

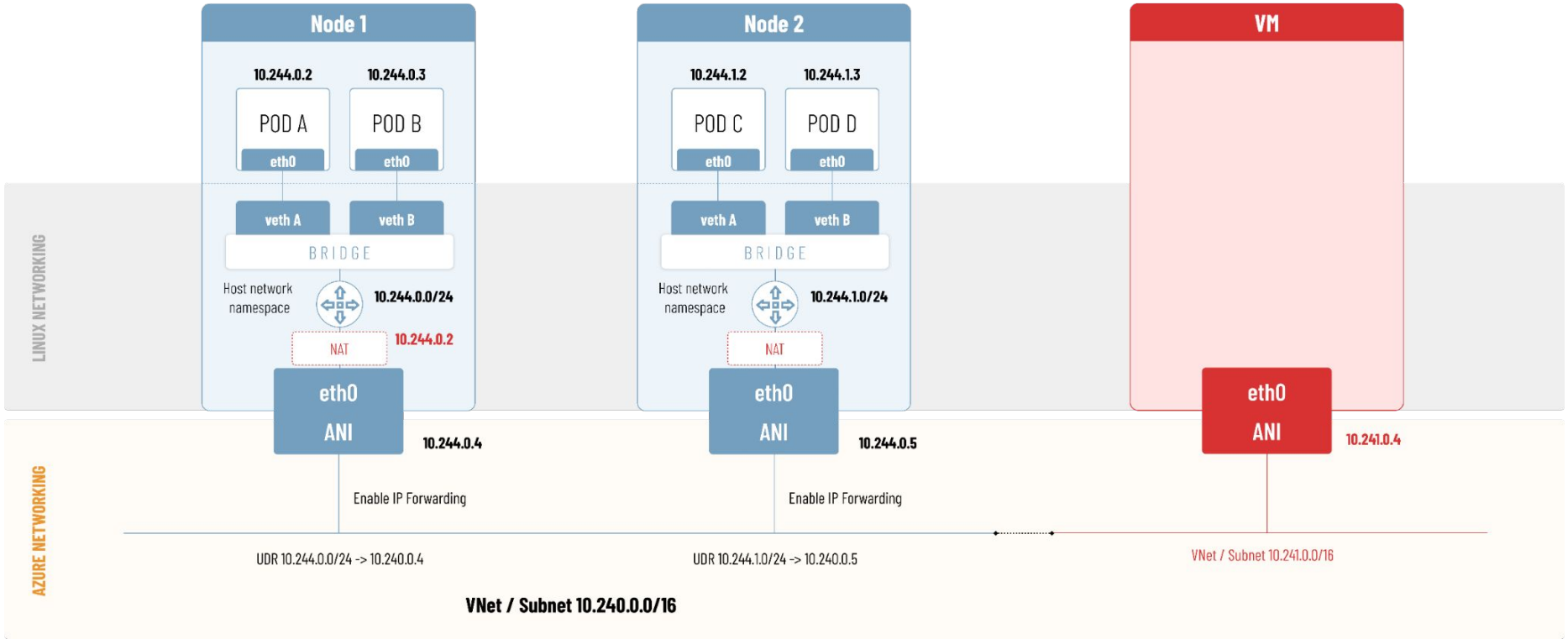

Node 1
10.244.0.2
10.244.0.3
POD A
POD B
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.0.0/24
eth0
ANI
10.244.0.4
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.1.0/24
eth0
ANI
10.244.0.5
LINUX NETWORKING
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

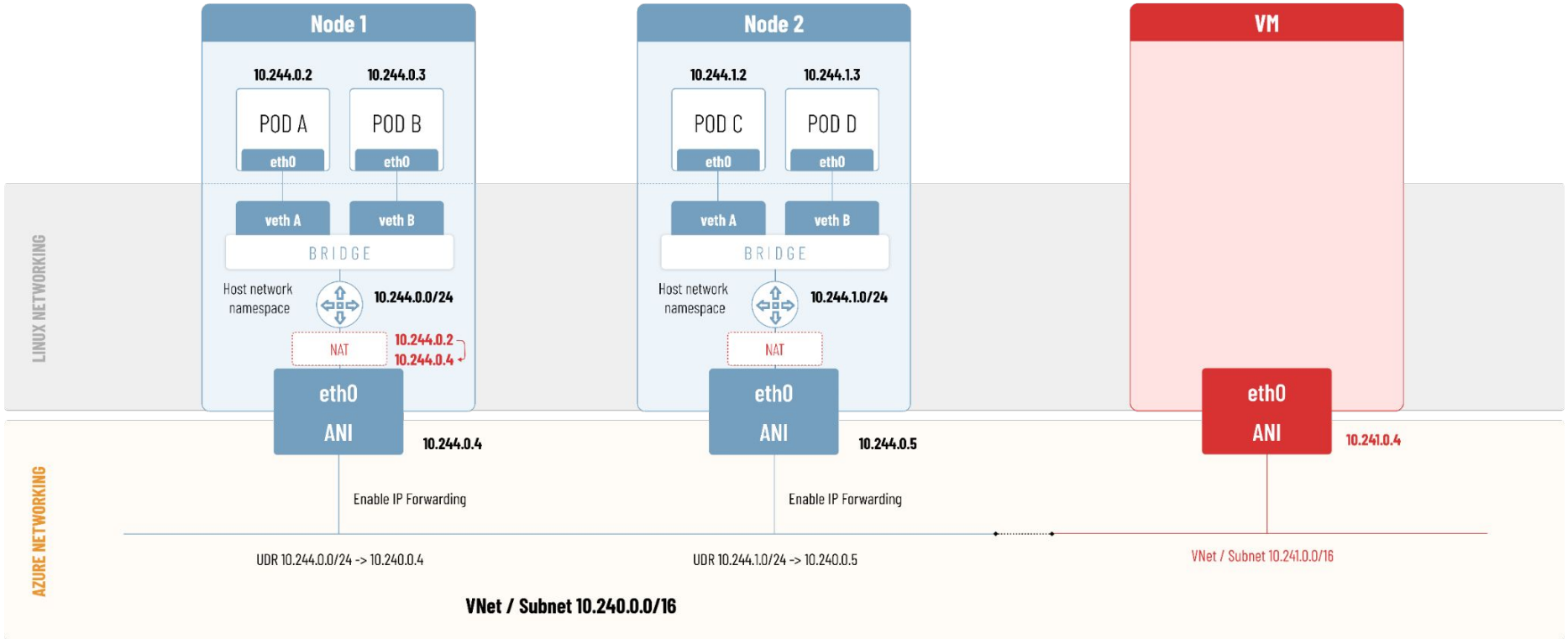

Node 1
10.244.0.2
10.244.0.3
POD A
POD B
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.0.0/24
eth0
ANI
10.244.0.4
Enable IP Forwarding
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.1.0/24
eth0
ANI
10.244.0.5
Enable IP Forwarding
LINUX NETWORKING
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16


Node 1
10.244.0.2
10.244.0.3
POD A
POD B
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.0.0/24
eth0
ANI
10.244.0.4
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.1.0/24
eth0
ANI
10.244.0.5
LINUX NETWORKING
AZURE NETWORKING
Enable IP Forwarding
Enable IP Forwarding
UDR 10.244.0.0/24 -> 10.240.0.4
UDR 10.244.1.0/24 -> 10.240.0.5
VNet / Subnet 10.240.0.0/16


Node 1
10.244.0.2
10.244.0.3
POD A
POD B
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.0.0/24
eth0 ANI
10.244.0.4
Enable IP Forwarding
UDR 10.244.0.0/24 -> 10.240.0.4
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.1.0/24
eth0 ANI
10.244.0.5
Enable IP Forwarding
UDR 10.244.1.0/24 -> 10.240.0.5
VM
eth0 ANI
10.241.0.4
VNet / Subnet 10.241.0.0/16
VNet / Subnet 10.240.0.0/16
LINUX NETWORKING
AZURE NETWORKING

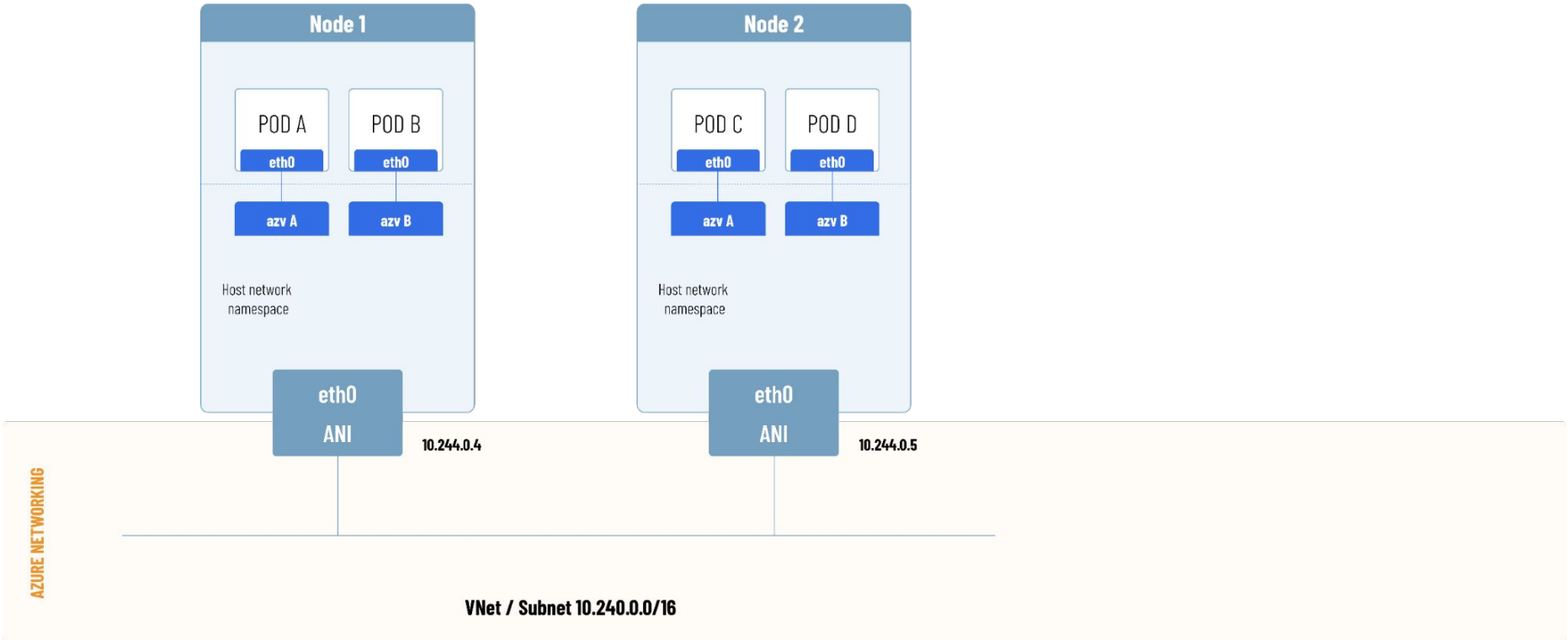

Node 1
10.244.0.2
10.244.0.3
POD A
POD B
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.0.0/24
NAT
eth0 ANI
10.244.0.4
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.1.0/24
NAT
eth0 ANI
10.244.0.5
VM
eth0 ANI
10.241.0.4
LINUX NETWORKING
AZURE NETWORKING
Enable IP Forwarding
Enable IP Forwarding
UDR 10.244.0.0/24 -> 10.240.0.4
UDR 10.244.1.0/24 -> 10.240.0.5
VNet / Subnet 10.241.0.0/16
VNet / Subnet 10.240.0.0/16

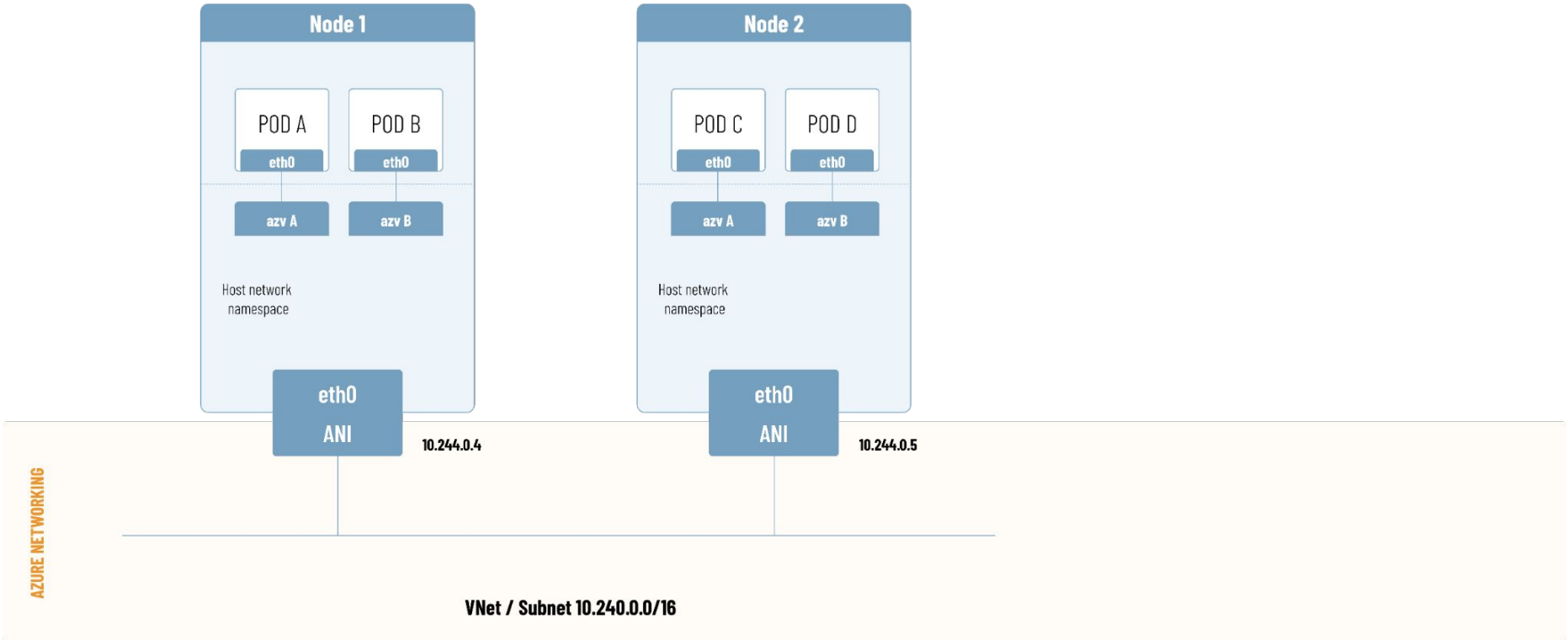

Node 1
10.244.0.2
10.244.0.3
POD A
POD B
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.0.0/24
NAT
10.244.0.2
eth0
ANI
10.244.0.4
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.1.0/24
NAT
eth0
ANI
10.244.0.5
VM
eth0
ANI
10.241.0.4
LINUX NETWORKING
AZURE NETWORKING
Enable IP Forwarding
Enable IP Forwarding
UDR 10.244.0.0/24 -> 10.240.0.4
UDR 10.244.1.0/24 -> 10.240.0.5
VNet / Subnet 10.241.0.0/16
VNet / Subnet 10.240.0.0/16

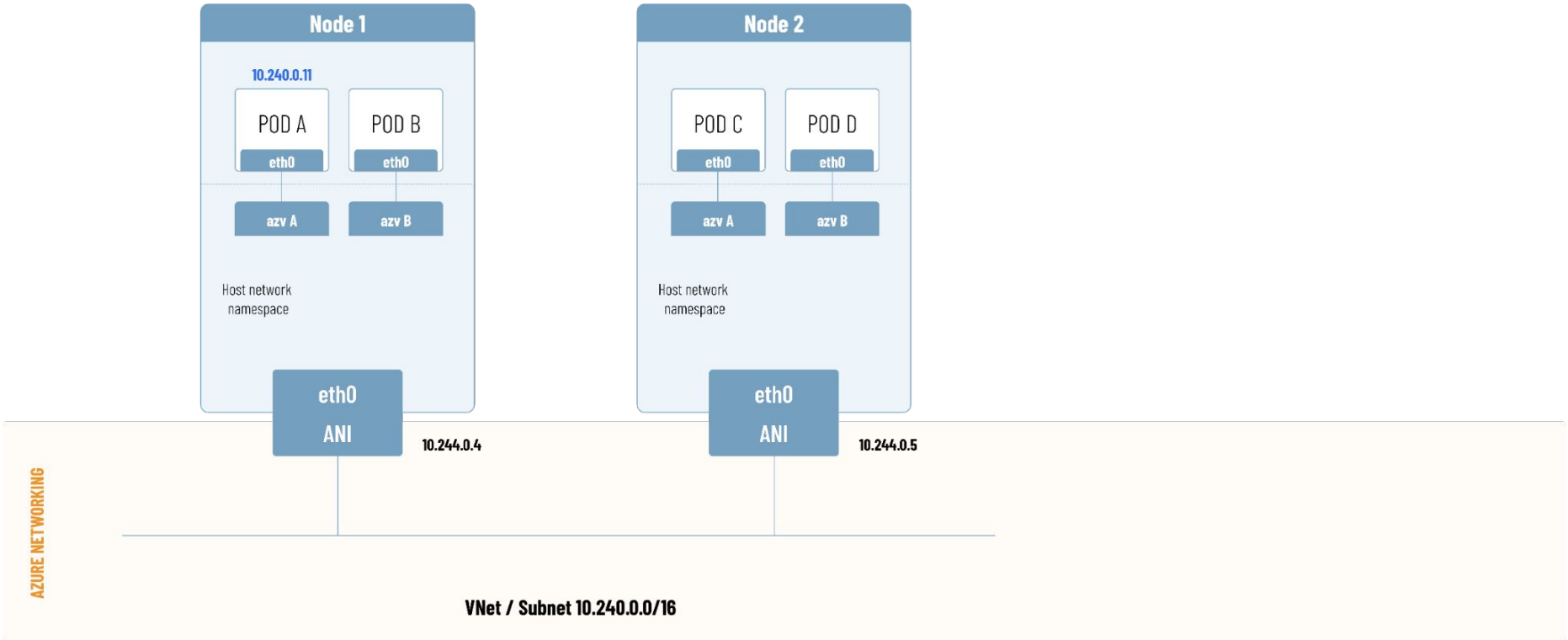

Node 1
10.244.0.2
10.244.0.3
POD A
POD B
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.0.0/24
NAT
10.244.0.2
10.244.0.4
eth0
ANI
10.244.0.4
Node 2
10.244.1.2
10.244.1.3
POD C
POD D
eth0
eth0
veth A
veth B
BRIDGE
Host network namespace
10.244.1.0/24
NAT
eth0
ANI
10.244.0.5
VM
eth0
ANI
10.241.0.4
LINUX NETWORKING
AZURE NETWORKING
Enable IP Forwarding
Enable IP Forwarding
UDR 10.244.0.0/24 -> 10.240.0.4
UDR 10.244.1.0/24 -> 10.240.0.5
VNet / Subnet 10.241.0.0/16
VNet / Subnet 10.240.0.0/16

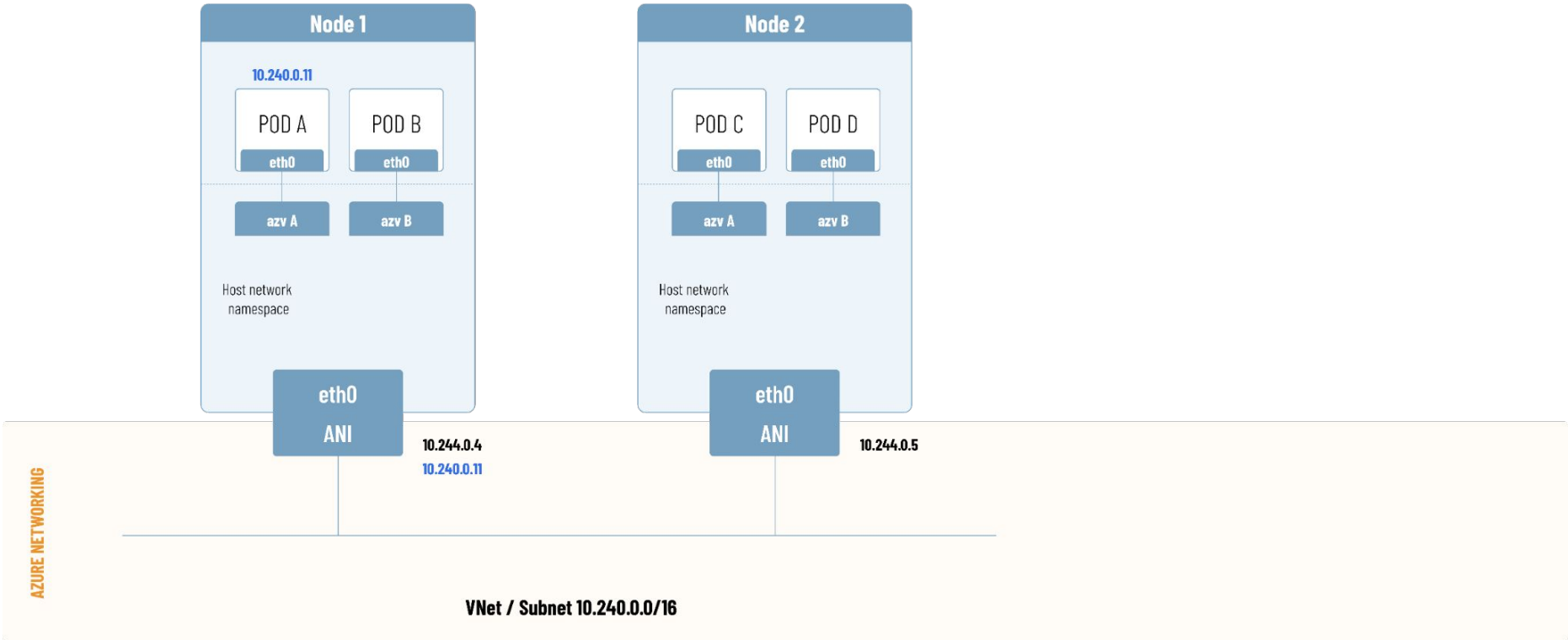
Advanced networking

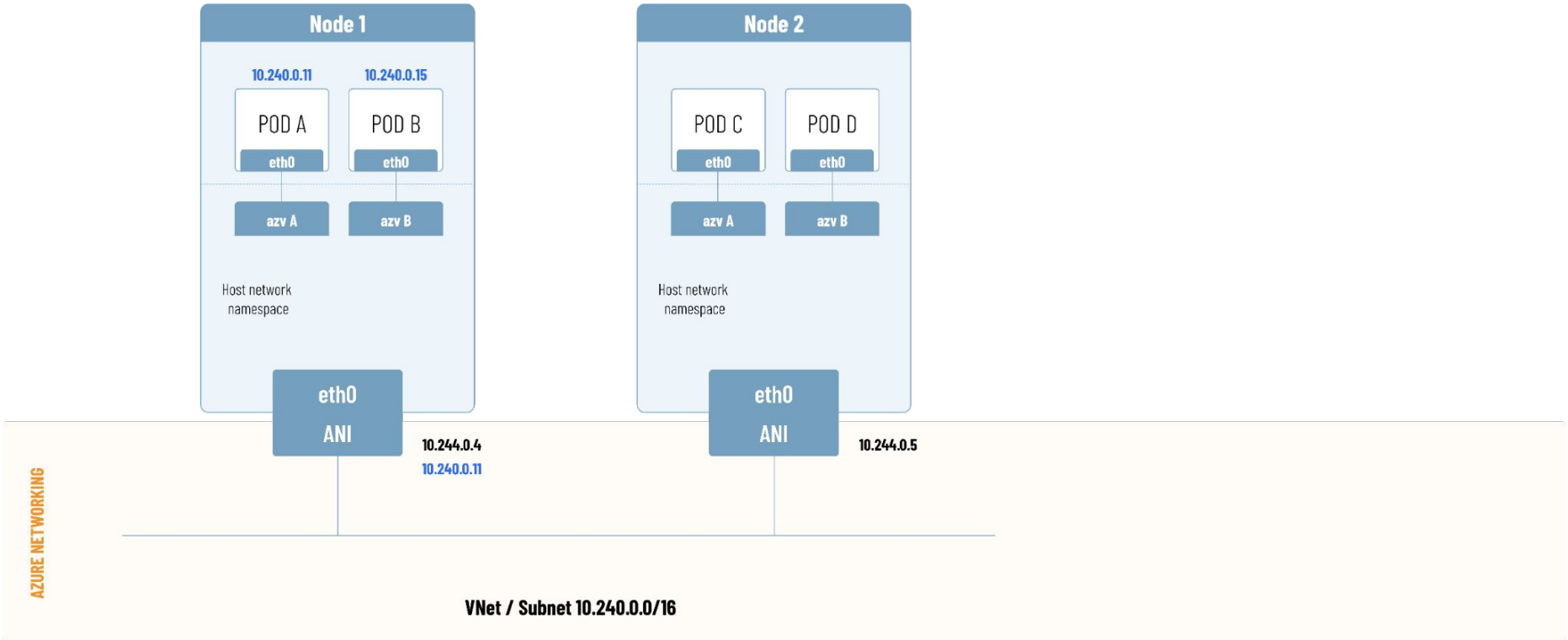
Azure CNI

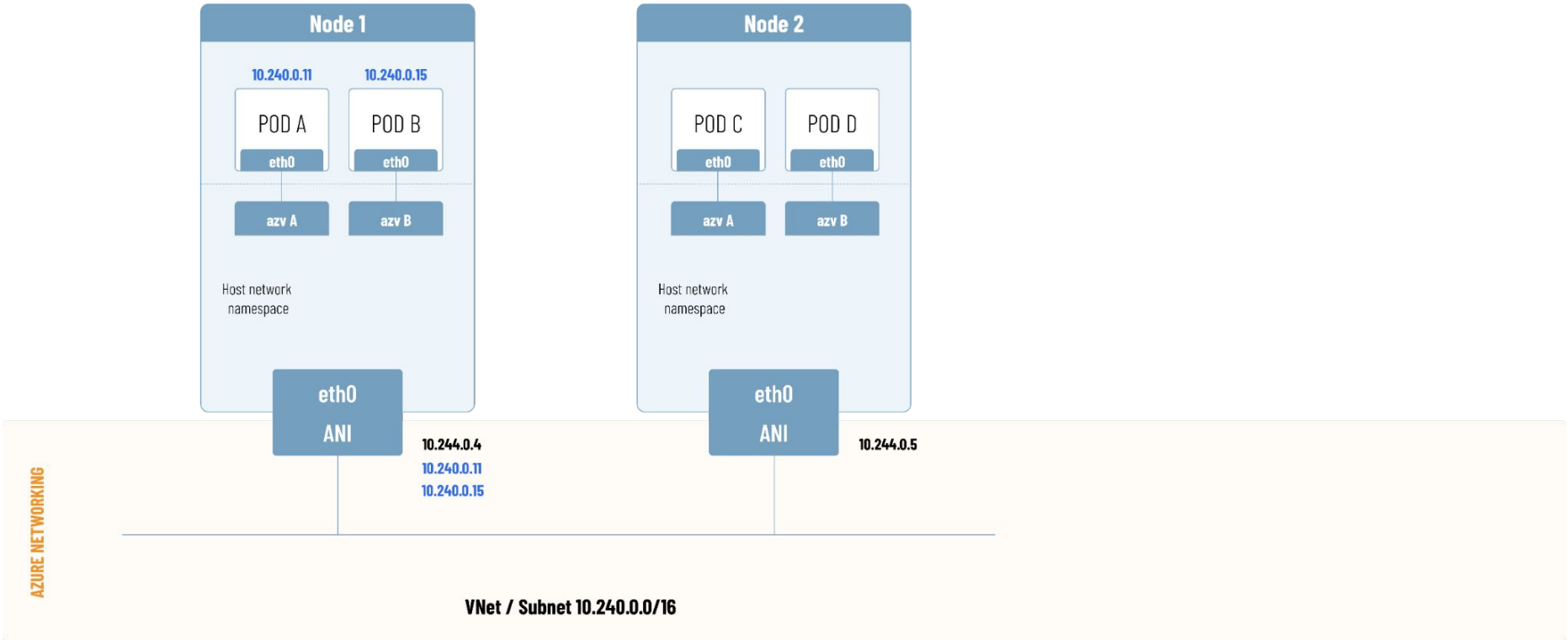

Node 1
POD A
POD B
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.4
Node 2
POD C
POD D
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

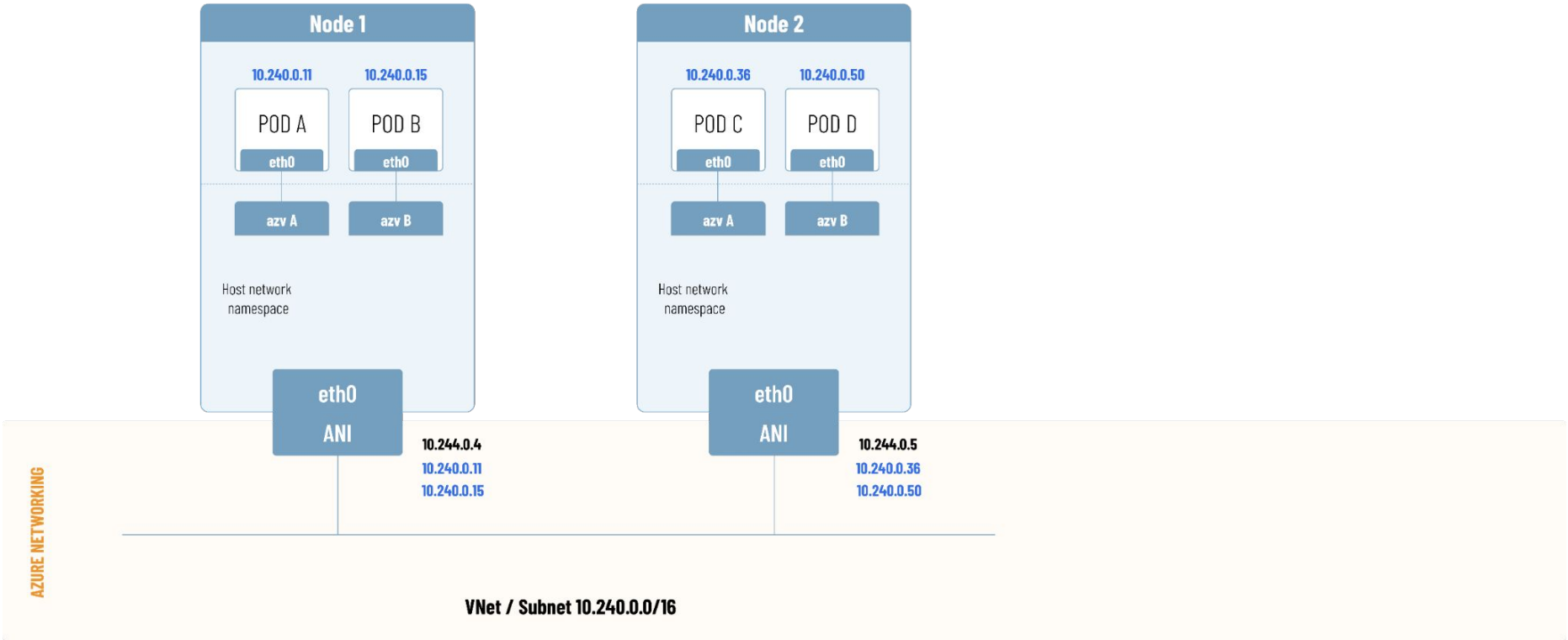

Node 1
POD A
POD B
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.4
Node 2
POD C
POD D
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

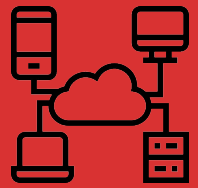

Node 1
10.240.0.11
POD A
POD B
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.4
Node 2
POD C
POD D
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16


Node 1
10.240.0.11
POD A
POD B
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.4
10.240.0.11
Node 2
POD C
POD D
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16


Node 1
10.240.0.11
10.240.0.15
POD A
POD B
eth0
eth0
azv A
azv B
Host network namespace
eth0 ANI
10.244.0.4
10.240.0.11
Node 2
POD C
POD D
eth0
eth0
azv A
azv B
Host network namespace
eth0 ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16


Node 1
10.240.0.11
10.240.0.15
POD A
POD B
eth0
eth0
azv A
azv B
Host network namespace
eth0 ANI
10.244.0.4
10.240.0.11
10.240.0.15
Node 2
POD C
POD D
eth0
eth0
azv A
azv B
Host network namespace
eth0 ANI
10.244.0.5
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

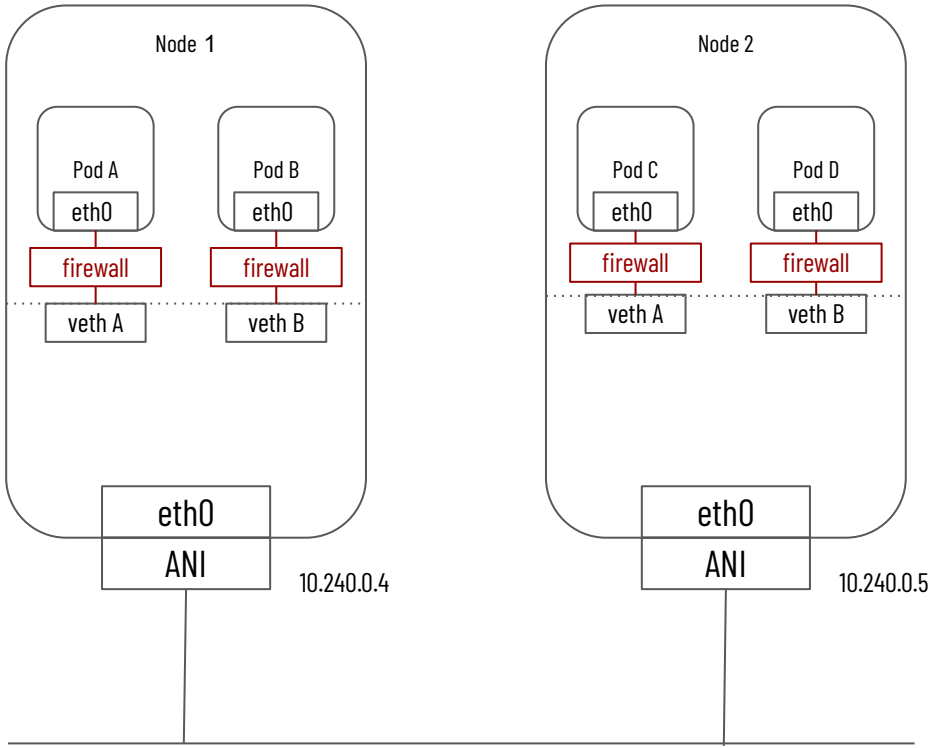

Node 1
10.240.0.11
10.240.0.15
POD A
POD B
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.4
10.240.0.11
10.240.0.15
Node 2
10.240.0.36
10.240.0.50
POD C
POD D
eth0
eth0
azv A
azv B
Host network namespace
eth0
ANI
10.244.0.5
10.240.0.36
10.240.0.50
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16



Network policies

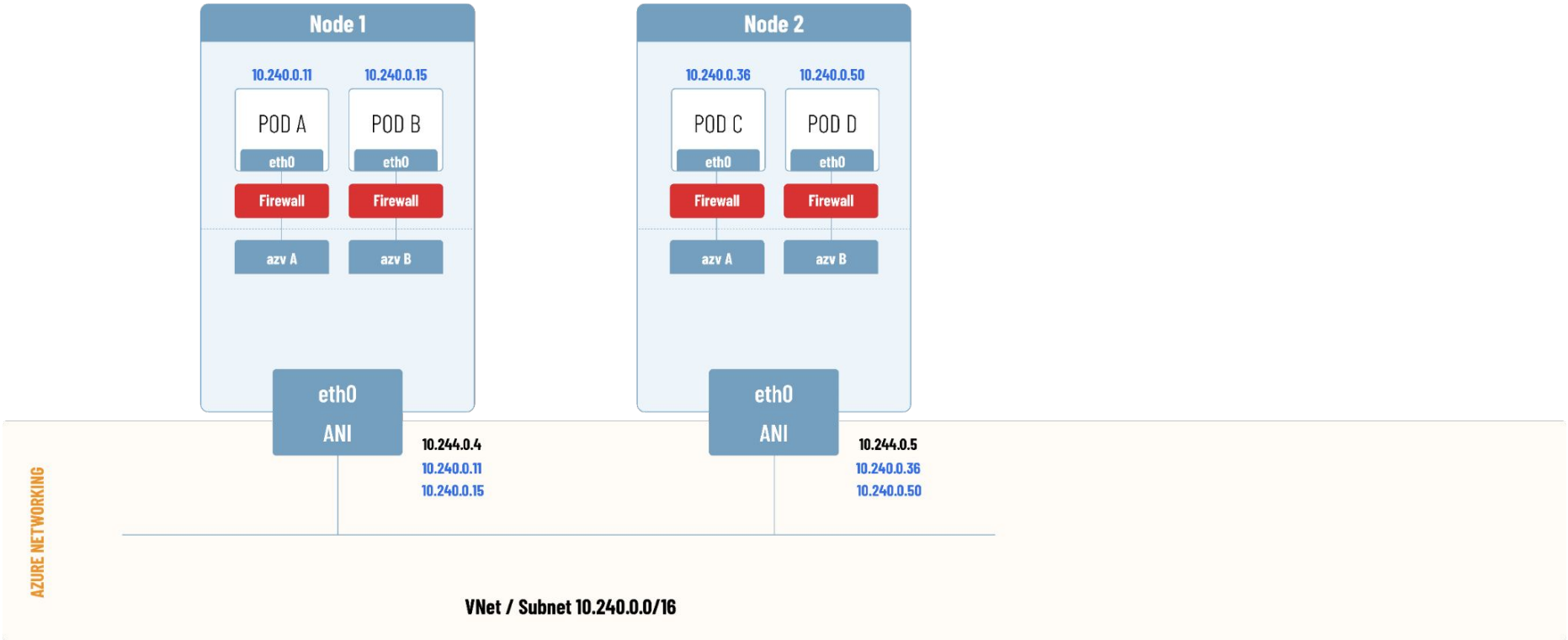
Network Policy

```
az aks create \
    --resource-group demo-rg \
    --name demo-cluster \
    --network-plugin {kubenet, azure} \
    --network-policy {azure, calico}
```



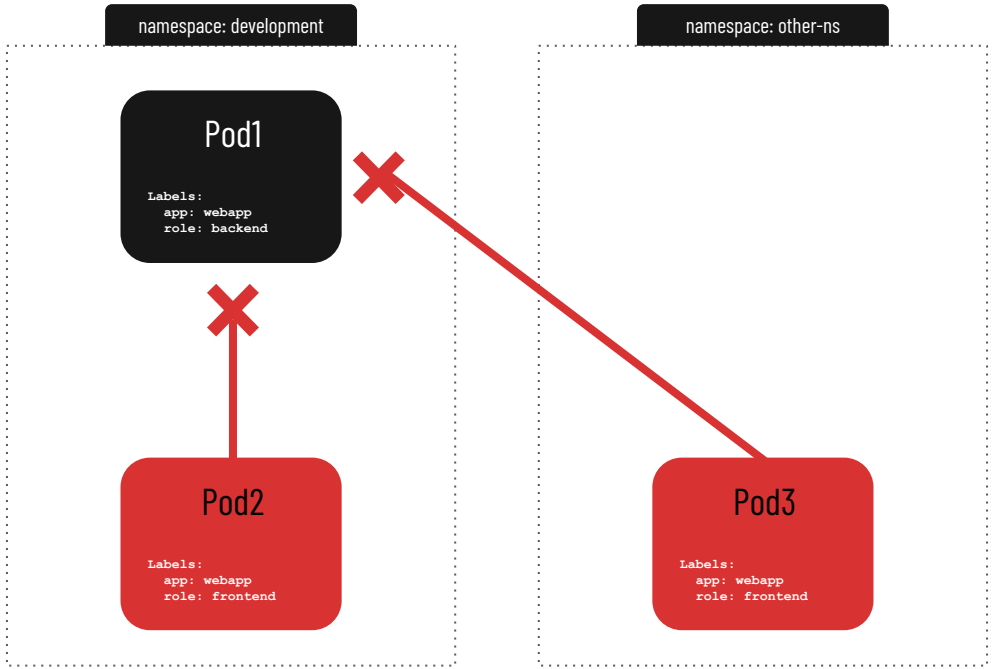
VNet / Subnet 10.240.0.0/16

Azure Networking


Node 1
10.240.0.11
10.240.0.15
POD A
POD B
eth0
eth0
Firewall
Firewall
azv A
azv B
eth0
ANI
10.244.0.4
10.240.0.11
10.240.0.15
Node 2
10.240.0.36
10.240.0.50
POD C
POD D
eth0
eth0
Firewall
Firewall
azv A
azv B
eth0
ANI
10.244.0.5
10.240.0.36
10.240.0.50
AZURE NETWORKING
VNet / Subnet 10.240.0.0/16

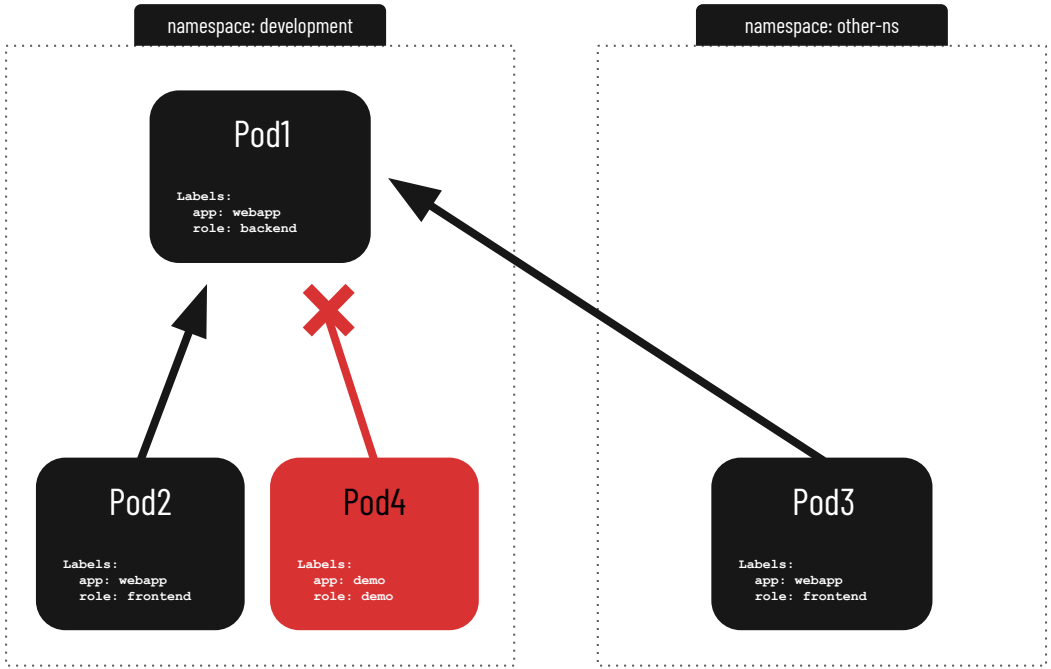
Deny all inbound traffic to a pod

```yaml
kind: NetworkPolicy
apiVersion: networking.k8s.io/v1
metadata:
  name: backend-policy
  namespace: development
spec:
  podSelector:
    matchLabels:
      app: webapp
      role: backend
  ingress: []
```



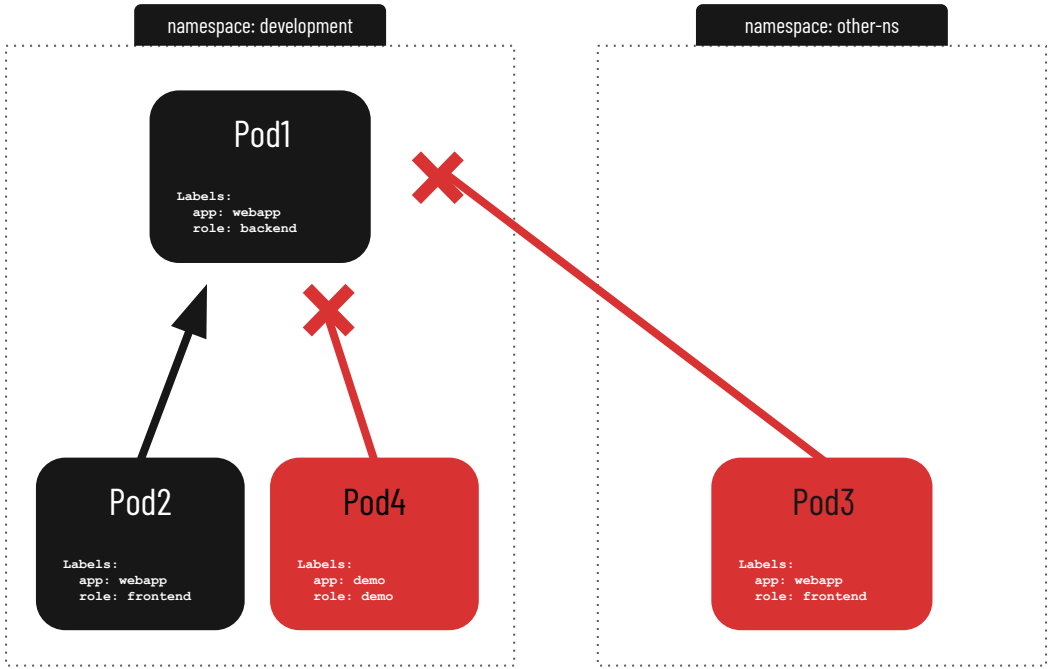
Allow inbound traffic based on a pod label

```yaml
kind: NetworkPolicy
apiVersion: networking.k8s.io/v1
metadata:
  name: backend-policy
  namespace: development
spec:
  podSelector:
    matchLabels:
      app: webapp
      role: backend
  ingress:
  - from:
    - namespaceSelector: {}
      podSelector:
        matchLabels:
          app: webapp
          role: frontend
```



Allow traffic only from within a defined namespace

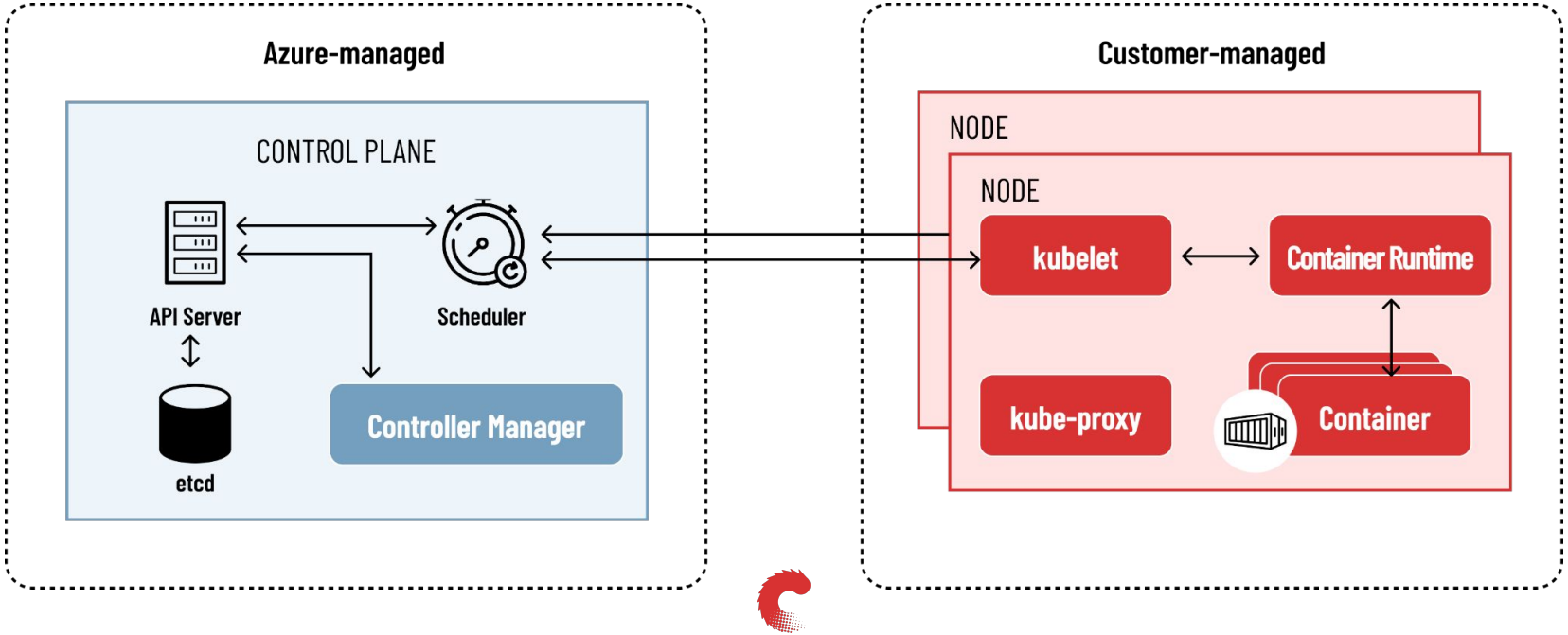
```yaml
kind: NetworkPolicy
apiVersion: networking.k8s.io/v1
metadata:
  name: backend-policy
  namespace: development
spec:
  podSelector:
    matchLabels:
      app: webapp
      role: backend
  ingress:
  - from:
    - namespaceSelector:
        matchLabels:
          purpose: development
      podSelector:
        matchLabels:
          app: webapp
          role: frontend
```



Business continuity

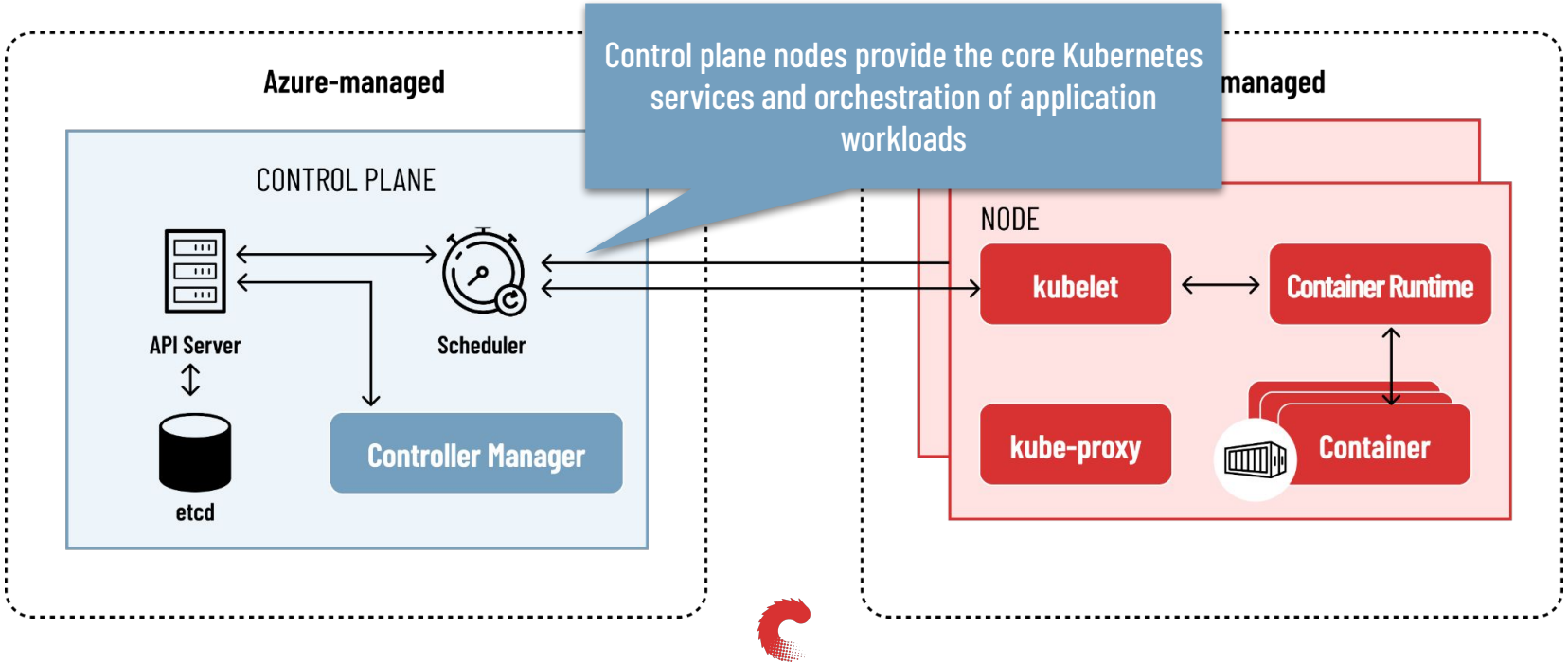
Kubernetes cluster architecture

A Kubernetes cluster is divided into two components:



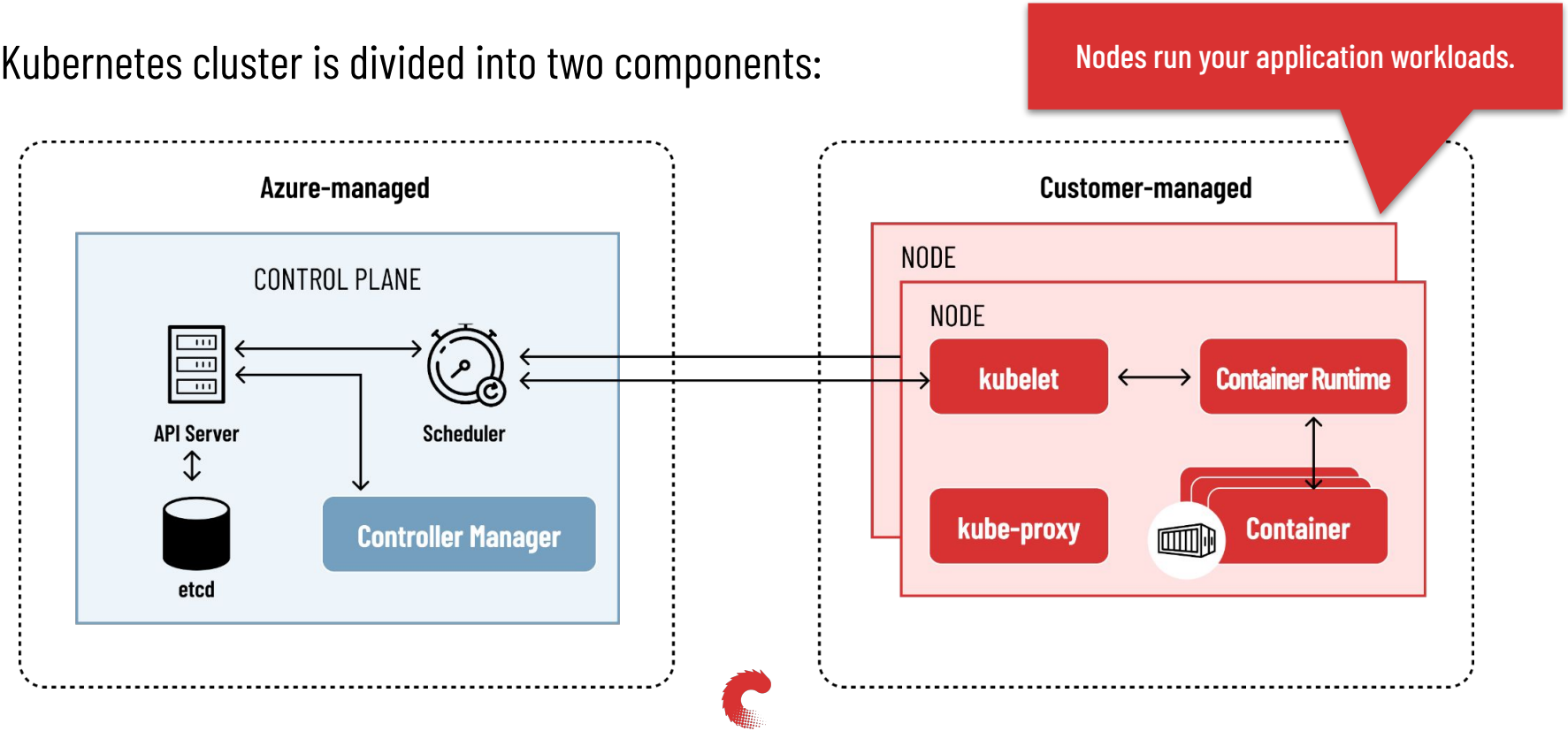
Kubernetes cluster architecture

A Kubernetes cluster is divided into two components:

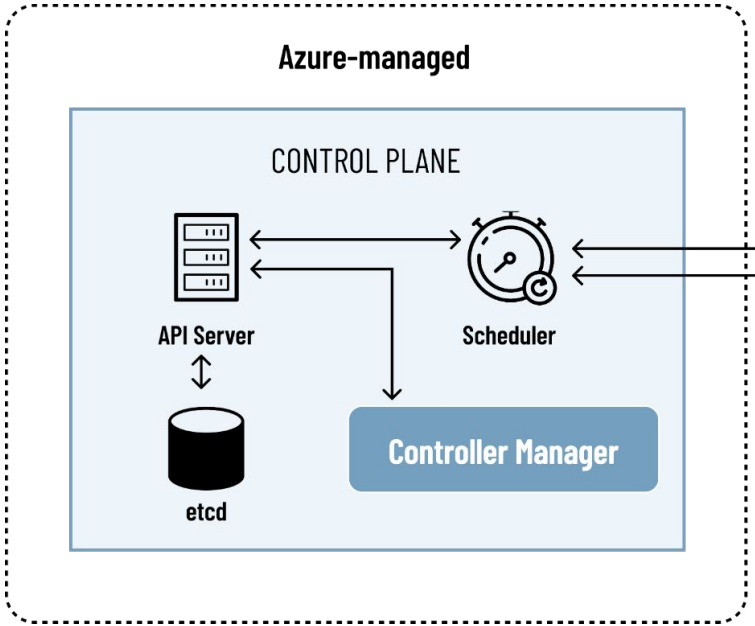


Kubernetes cluster architecture

A Kubernetes cluster is divided into two components:



Control plane



When you create an AKS cluster, a control plane is automatically created and configured.

This control plane is provided as a managed Azure resource abstracted from the user.

There's no cost for the control plane, only the nodes that are part of the AKS cluster.

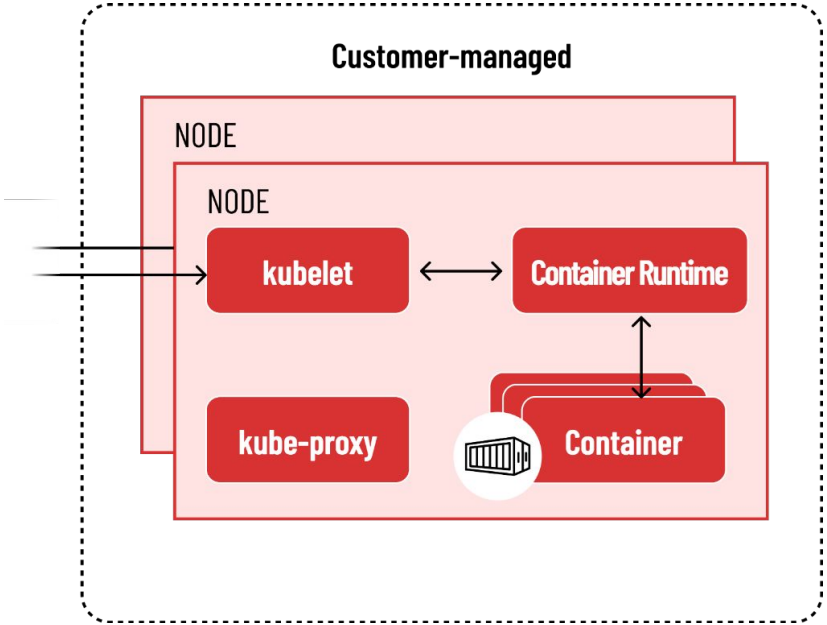
API server business SLA

```
az aks create \
   --resource-group demo-rg \
   --name demo-cluster \
   --uptime-sla
```

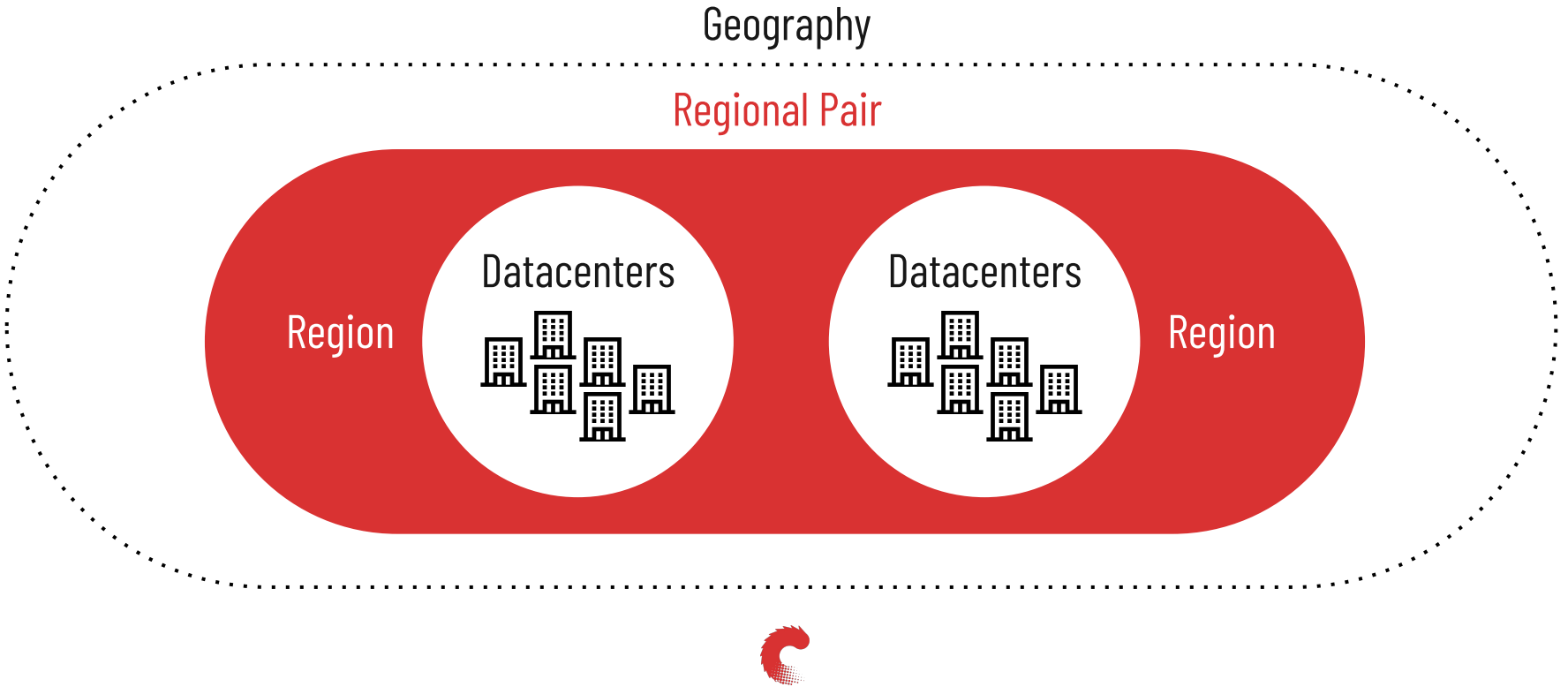
Nodes and node pools

To run your applications and supporting services, you need a Kubernetes node.

An AKS cluster has one or more nodes, which is an Azure virtual machine (VM) that runs the Kubernetes node components and container runtime



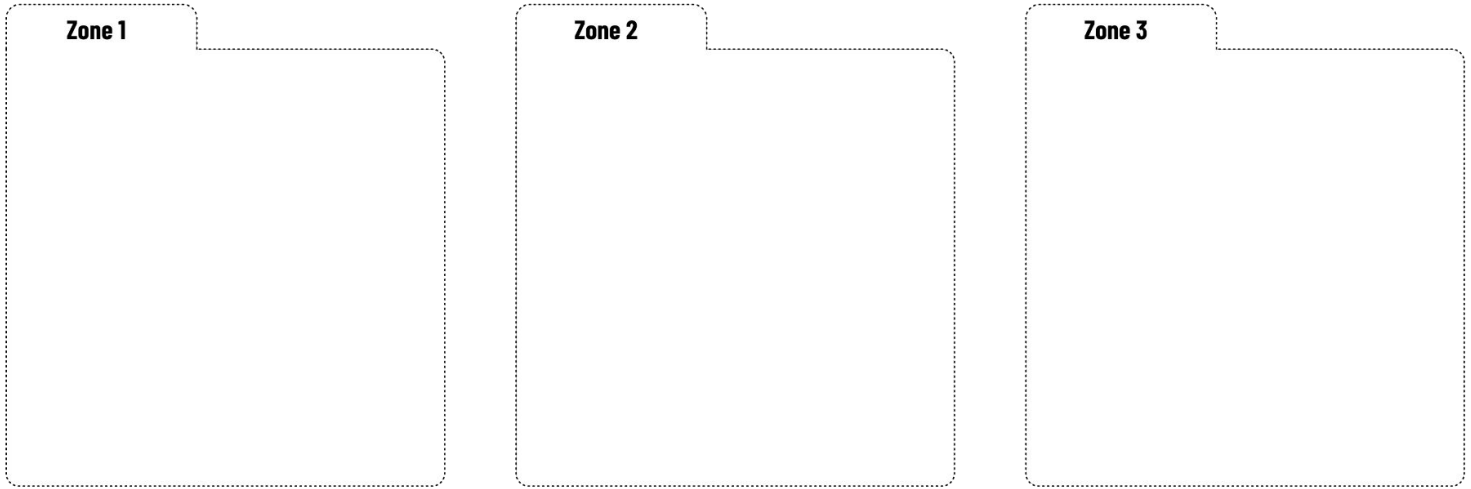
Allow traffic only from within a defined namespace



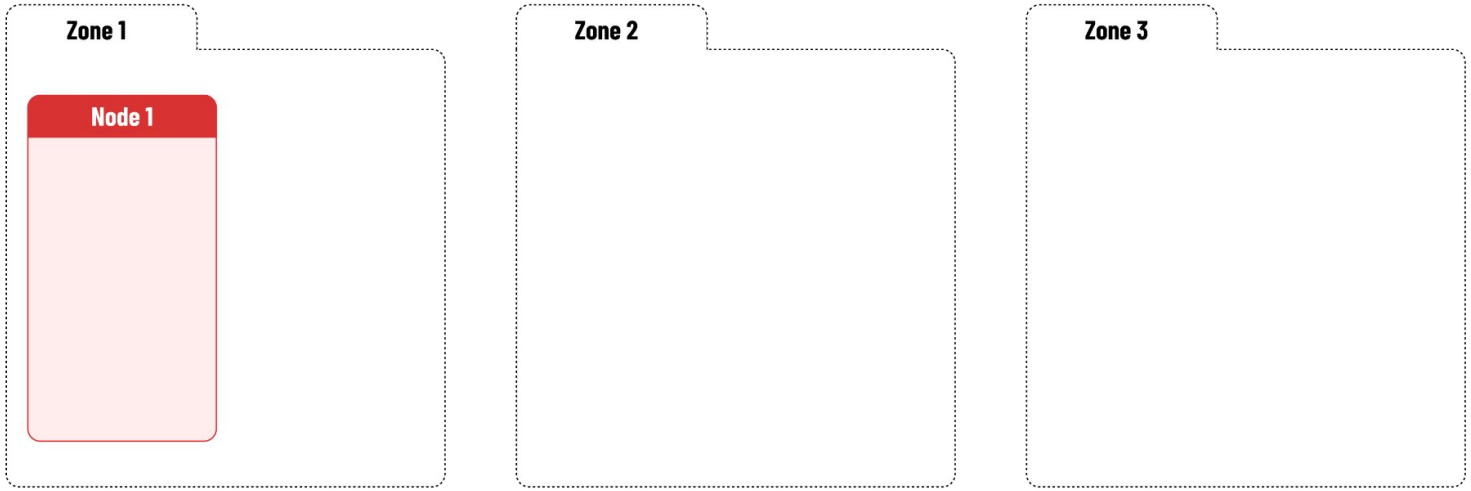
Availability zones

```
az aks create \
   --resource-group demo-rg \
   --name demo-cluster \
   --zones {1, 2, 3}
```

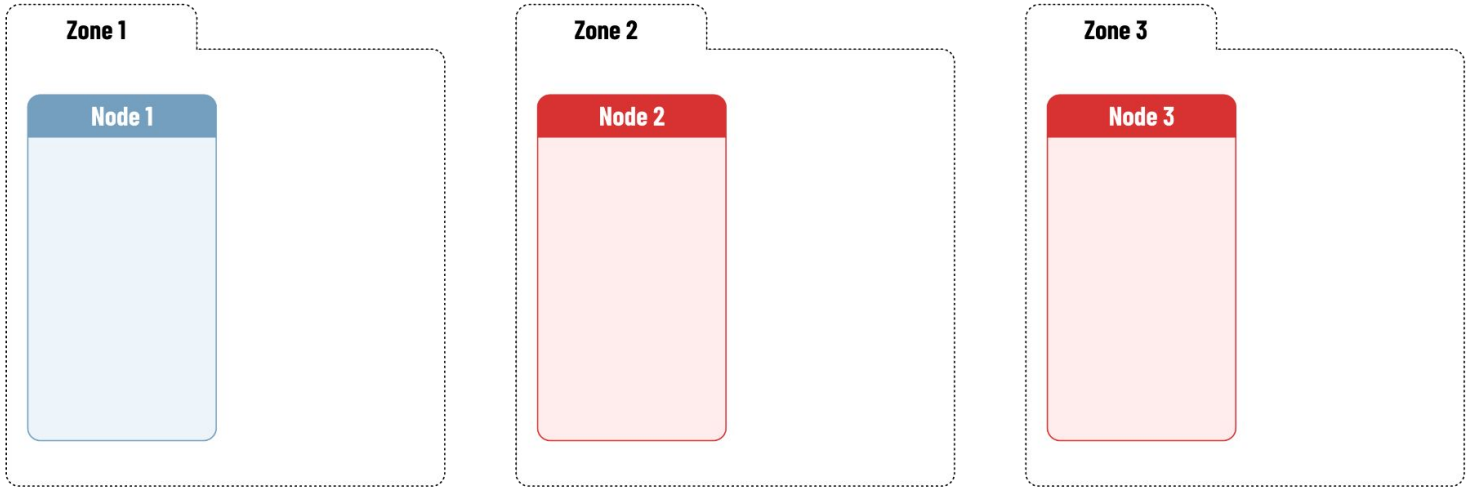
AKS and Availability Zones



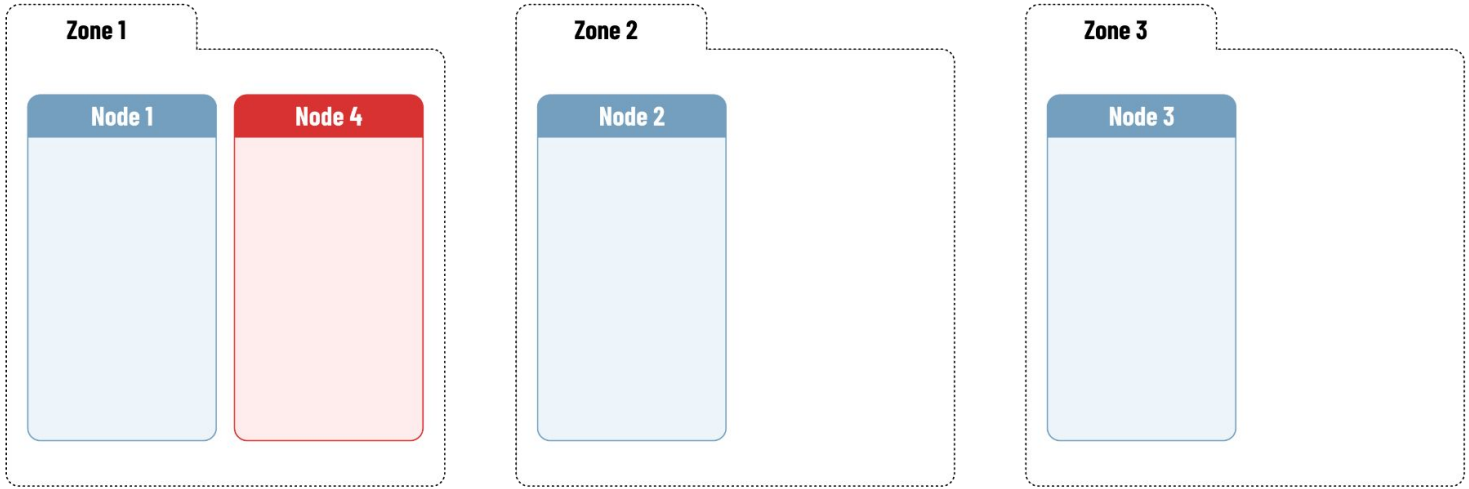
AKS and Availability Zones



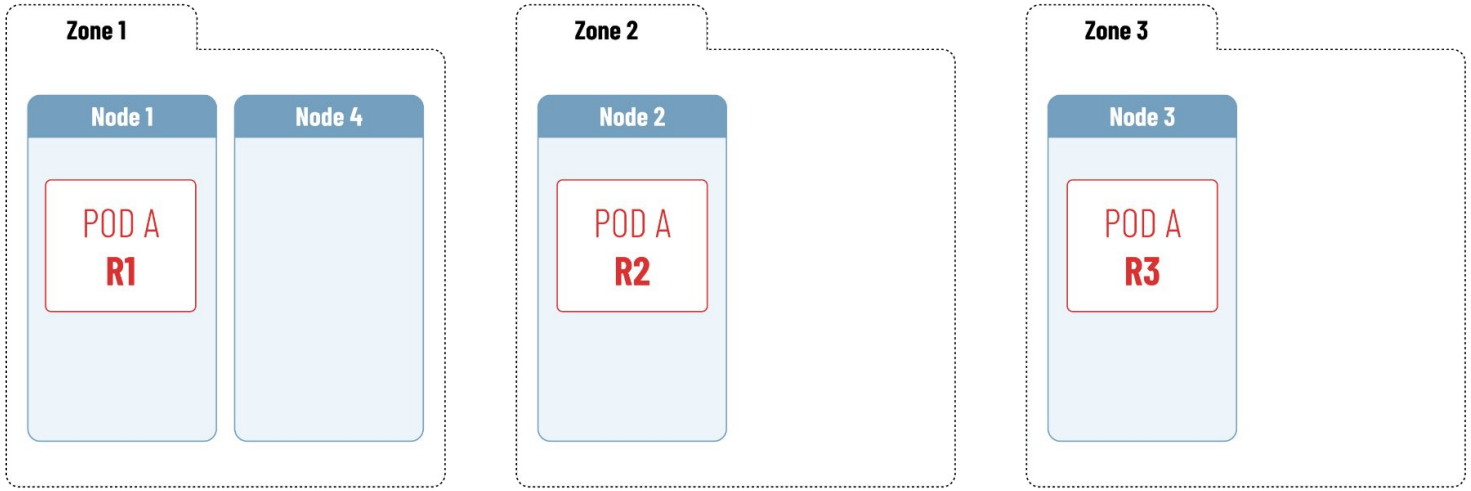
AKS and Availability Zones



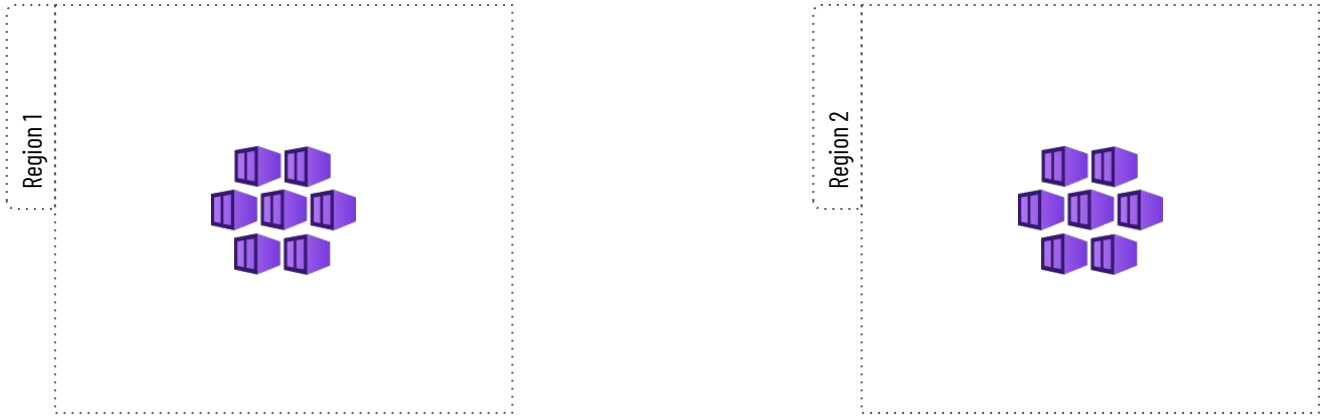
AKS and Availability Zones



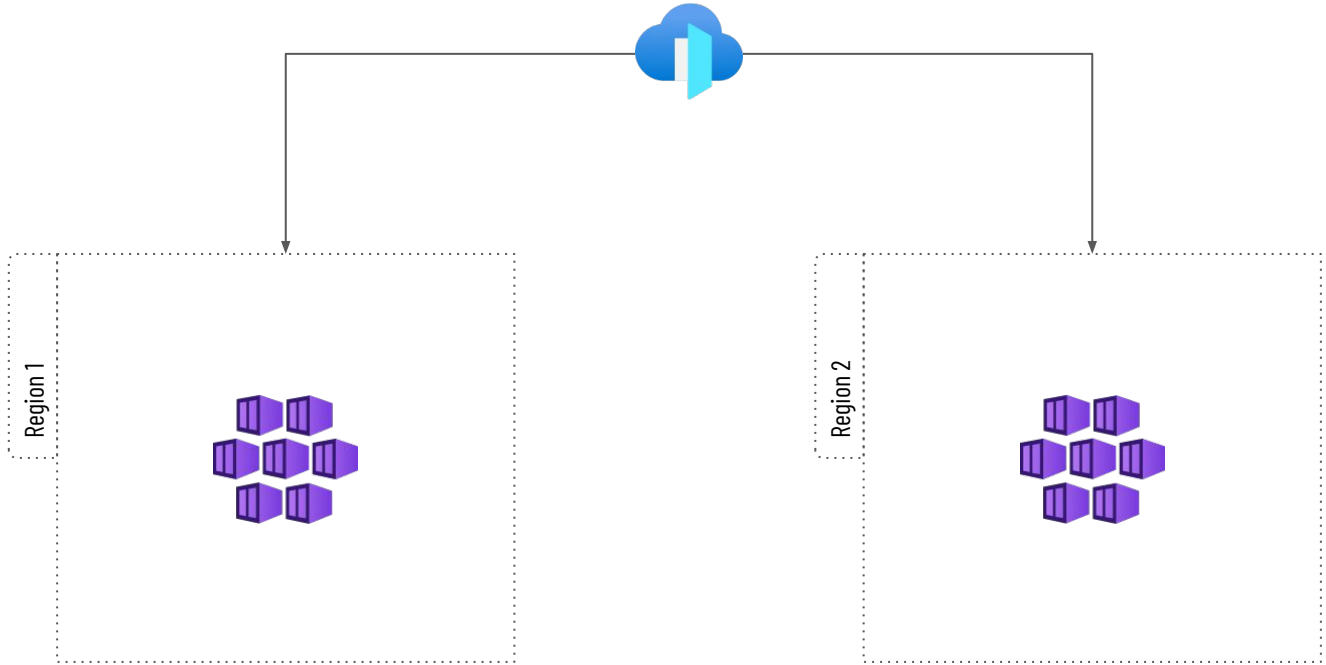
AKS and Availability Zones



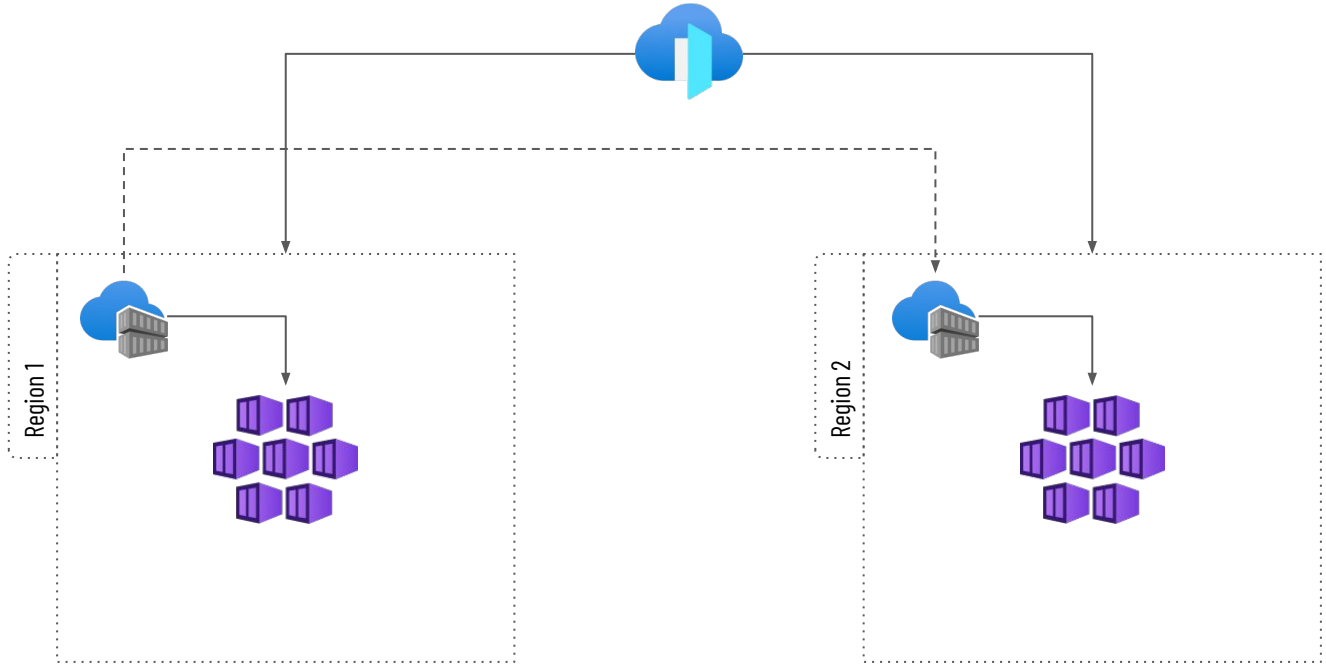
AKS and cross-region HA



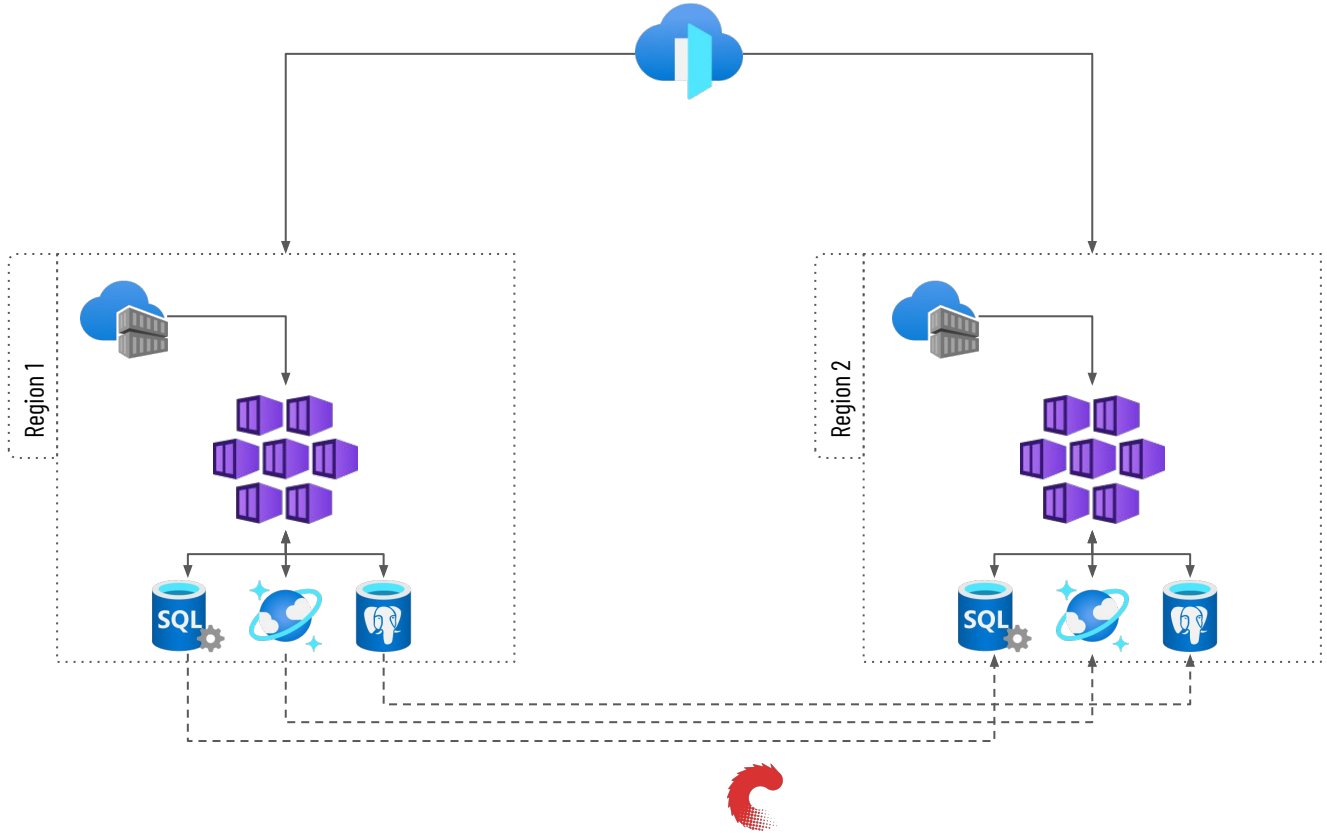
AKS and cross-region HA



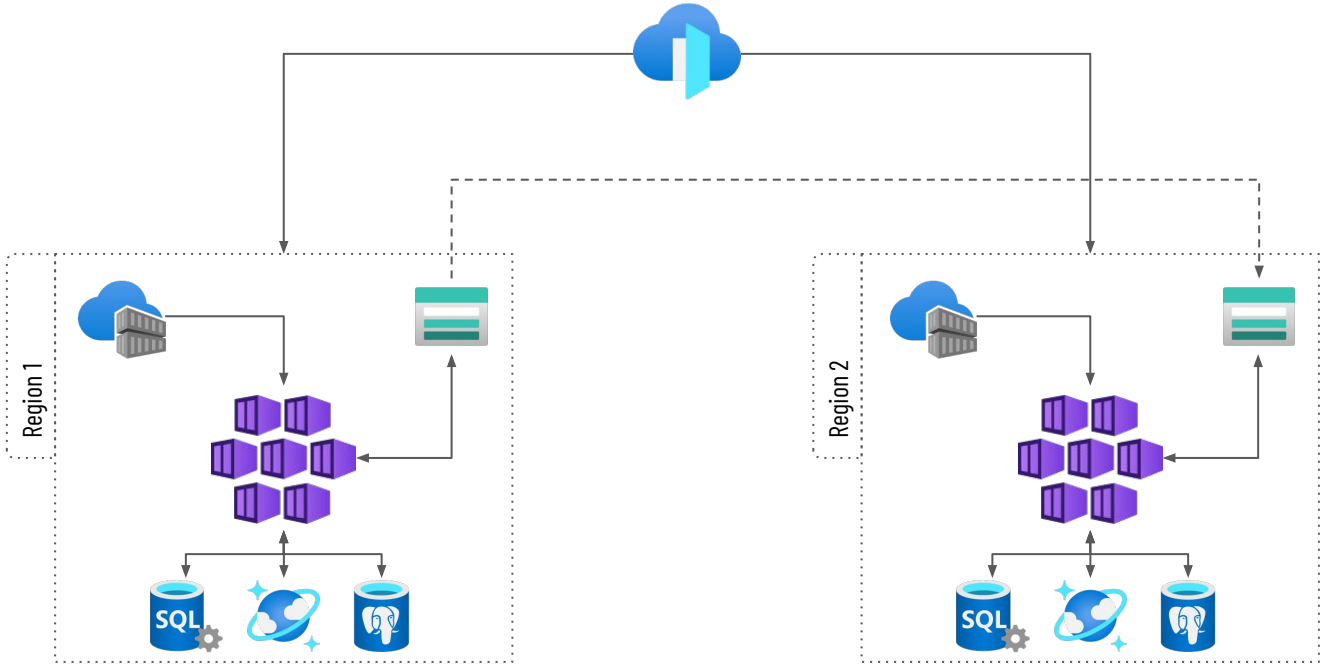
AKS and cross-region HA



AKS and cross-region HA



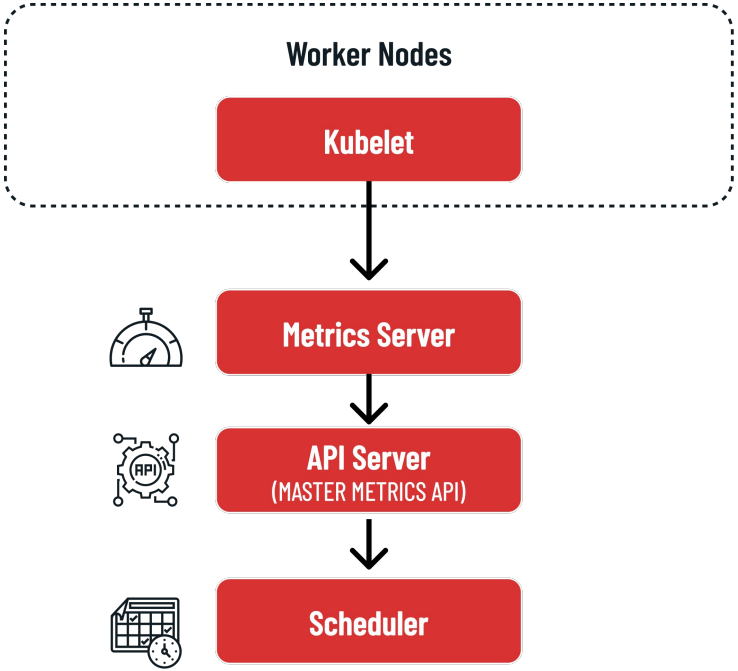
AKS and cross-region HA



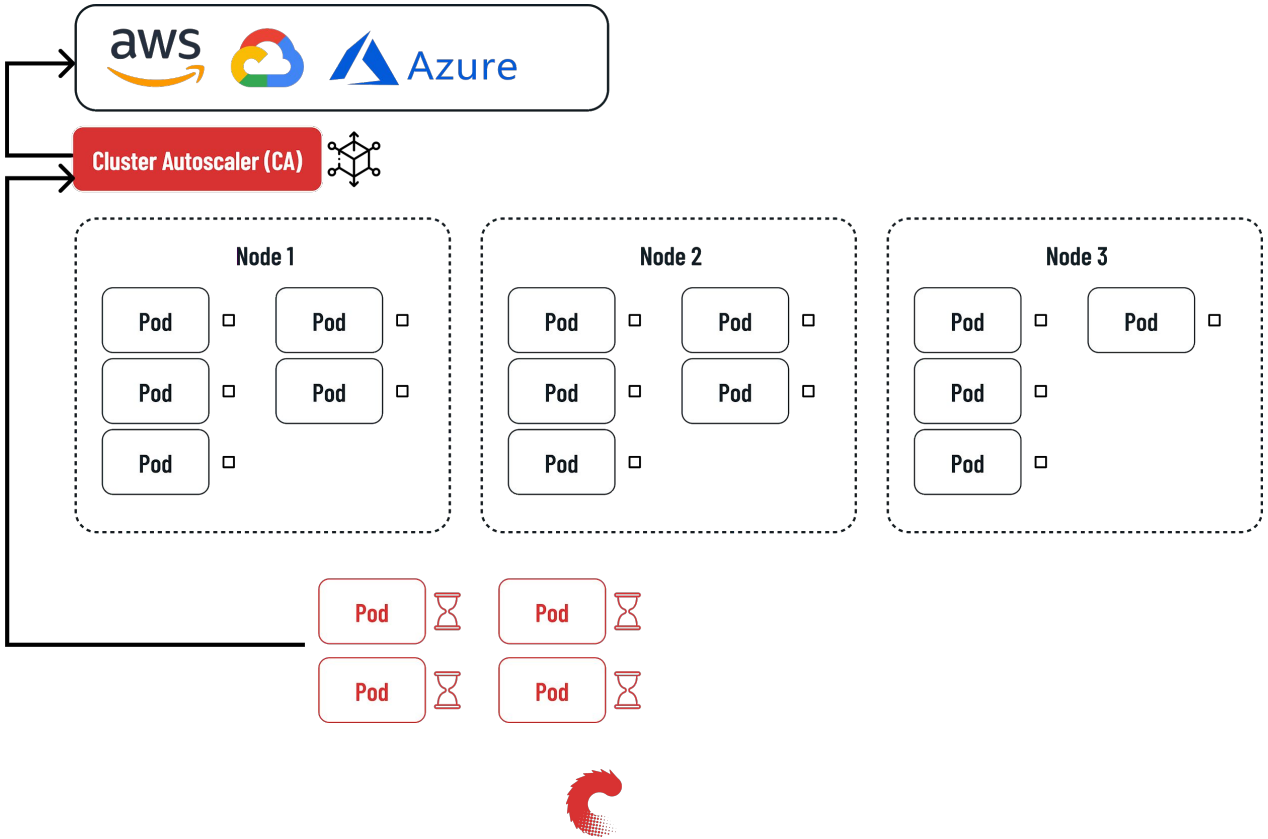
Cluster autoscaler

```
az aks create \
   --resource-group demo-rg \
   --name demo-cluster \
   --enable-cluster-autoscaler \
   --min-count 1 \
   --max-count 3
```

Scheduling pod



Cluster autoscaler




That's all folks!

Grazie!

- Il materiale sarà online nei prossimi giorni su http://www.communitydays.it

ALESSANDRO MELCHIORI

Founder & Software developer @ CodicePlastico
alessandro@codiceplastico.com
@amelchiori

Grazie!

FOLLOW US