CNI

Deep-Dive
Packet-level Debugging of CNI Plugins

Sedef Savas (@ssavas)

Jay Vyas (@jayunit100)

Software Engineers @

August 20, 2020

Agenda

Kubernetes Networking Concepts +

Networking Approaches +

Practical CNI tracing demo =

Learn how to reason about pod networking in your clusters !!!!!!!!!

Kubernetes networking concepts

5 fundamentals in the K8s network stack

Pod-to-pod networking (CNI plugins)

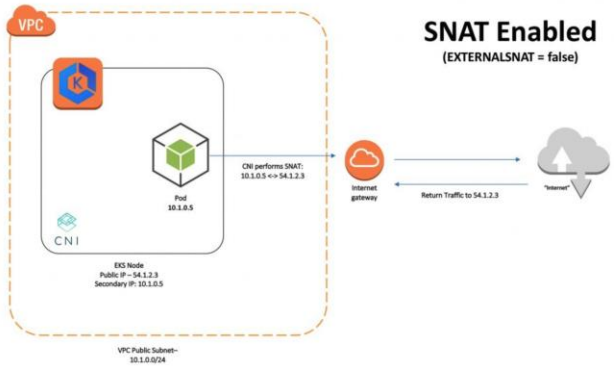
Services (kube-proxy)

Service discovery (kubedns/ coredns)

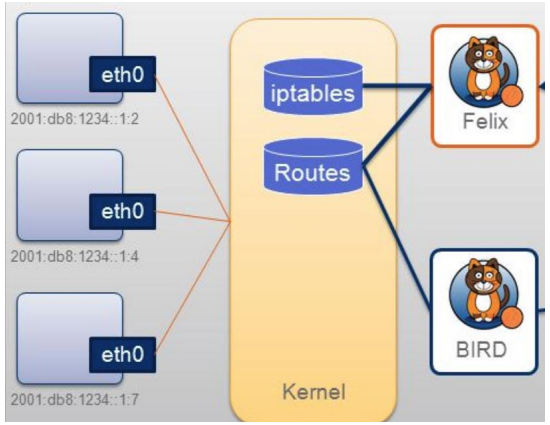
Network policies (CNI plugins)

Load balancers (ingress)

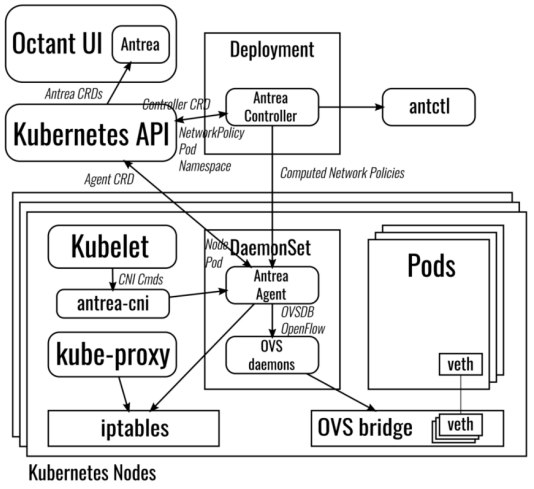
Amazon EKS



PROJECT CALICO

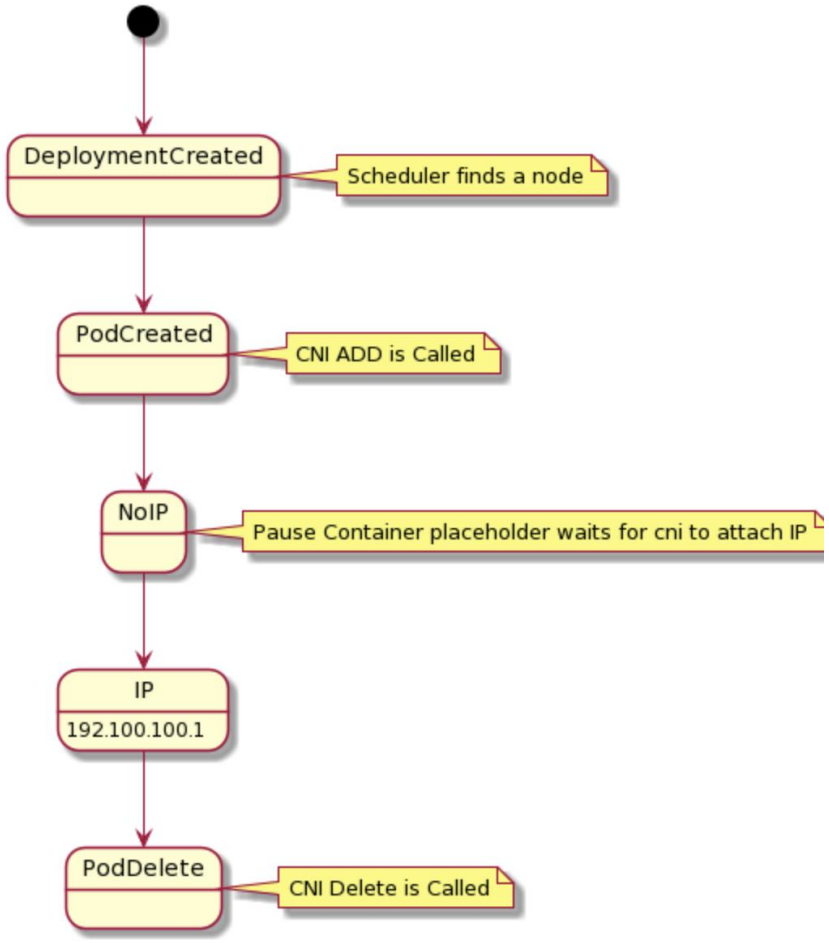


ANTREA



Container Network Interface (CNI)

The life of an IP address...



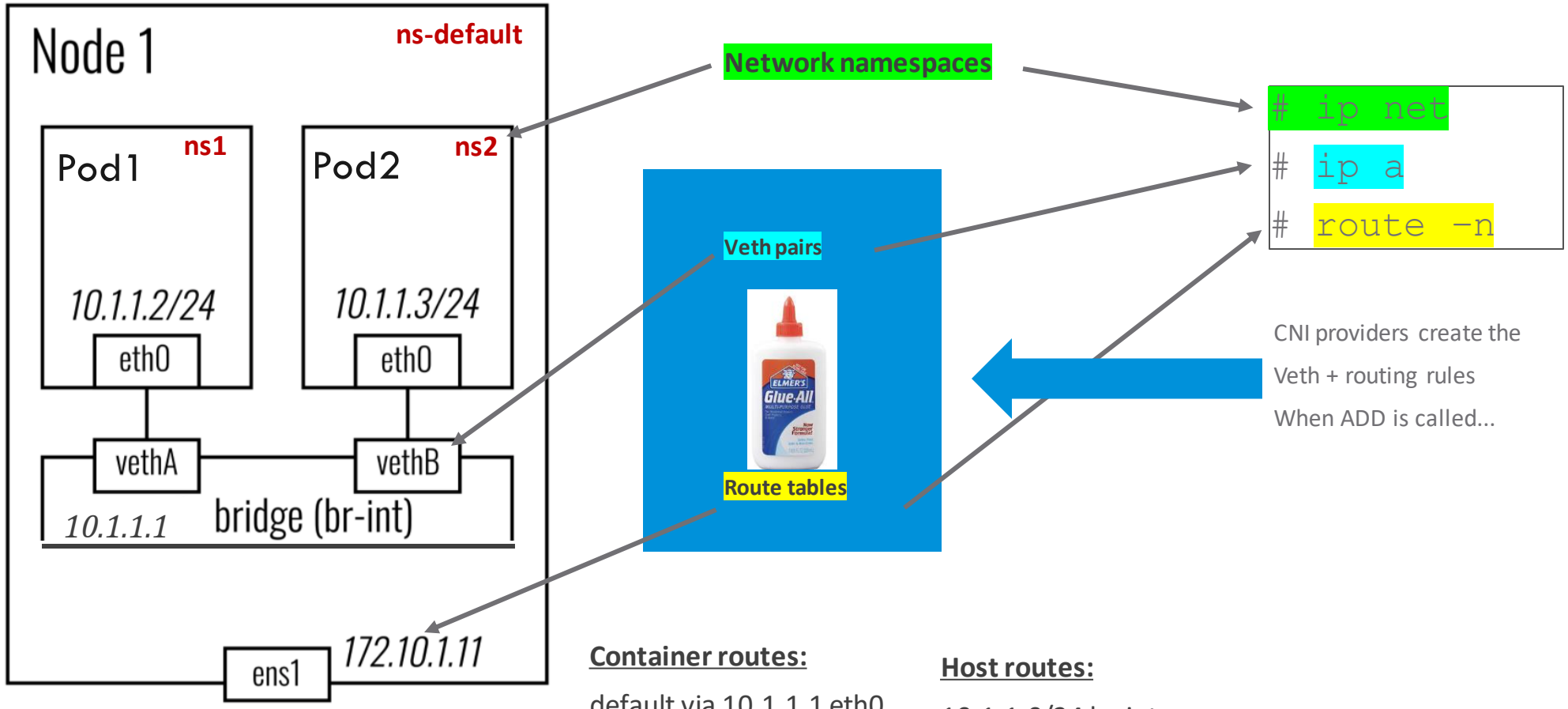
CNI: provides a contract between networks and containers

CNI plugins: are responsible for IPAM, connectivity between pods, security

When Pods are initialized or removed, CNI plug-in (ADD/DELETE) is called

CNI glues Linux infrastructure into your pod

2 pods on the same node



Used By

Antrea,

GKE (cbr0)

Container routes:

default via 10.1.1.1 eth0

10.1.1.0/24 eth0

Host routes:

10.1.1.0/24 br-int

172.10.1.0/24 ens1

Communication across pods/services in different nodes

Non-overlay Networking (No encapsulation)

- **Layer 2 networking (same subnet)** - via arp/broadcast/route entry
- **Layer 3 networking** - via BGP or SDN
  - E.g., Calico, private datacenter CNIs

Overlay Networking (Encapsulation)

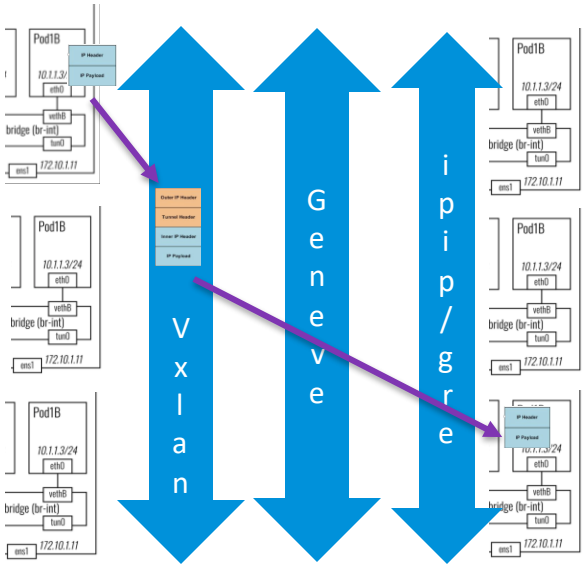
- **Encapsulation with VXLAN/GENEVE**

Hybrid Mode (Non-overlay + Overlay)

- Many CNIs can make decisions based on the network topology on *when* to encapsulate, and when not to.

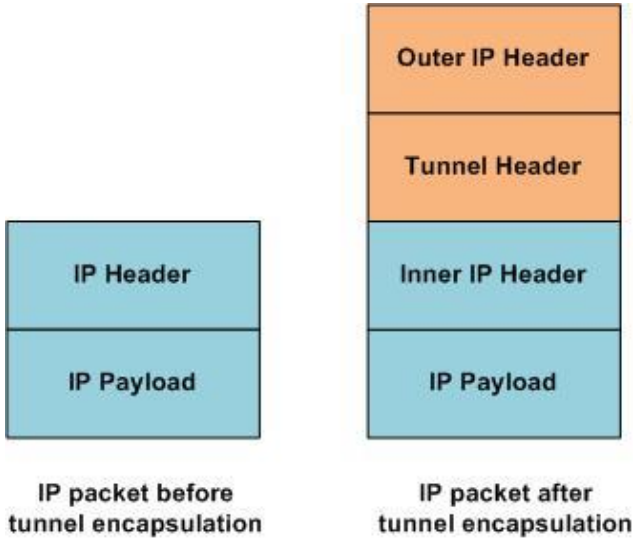
Antrea/Calico both support encap and no encap modes.

How encapsulation works…



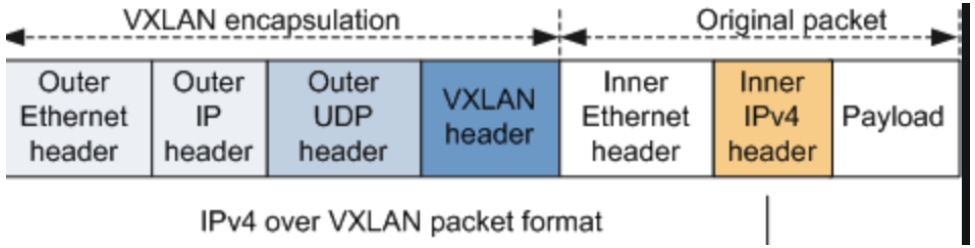
IP/IP Tunneling

(2 IP headers)



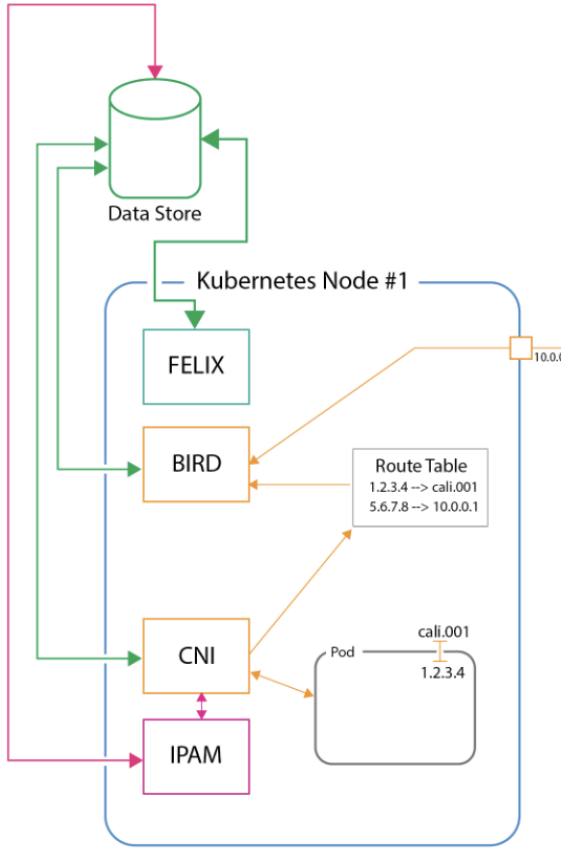

```
Frame 81: 94 bytes on wire (752 bits), 94 bytes captured (752 bits) on
Ethernet II, Src: 06:ec:51:d6:80:ea (06:ec:51:d6:80:ea), Dst: 06:be:26
Internet Protocol Version 4, Src: 10.30.0.206, Dst: 10.30.0.56
Internet Protocol Version 4, Src: 192.168.226.69, Dst: 192.168.133.194
```

VXLAN Tunneling (Ethernet frames to UDP packets)




```
Frame 36: 124 bytes on wire (992 bits), 124 bytes captured (992 bits) on interface /var/folders/0d/
Ethernet II, Src: 06:2f:22:b7:04:68 (06:2f:22:b7:04:68), Dst: 06:23:08:e1:16:18 (06:23:08:e1:16:18)
Internet Protocol Version 4, Src: 10.30.0.105, Dst: 10.30.1.131
User Datagram Protocol, Src Port: 37489, Dst Port: 4789
Virtual eXtensible Local Area Network
Ethernet II, Src: 66:93:7f:54:d5:53 (66:93:7f:54:d5:53), Dst: 66:1c:4e:98:15:45 (66:1c:4e:98:15:45)
```

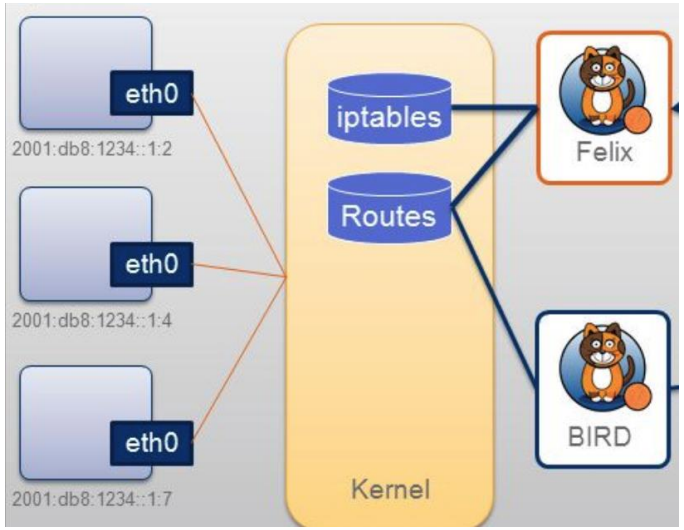
Calico (a non-bridged CNI plugin)



https://medium.com/@k.grundy/project-calico-kubernetes-integration-overview-a3a860cd974e

Supported networking modes:

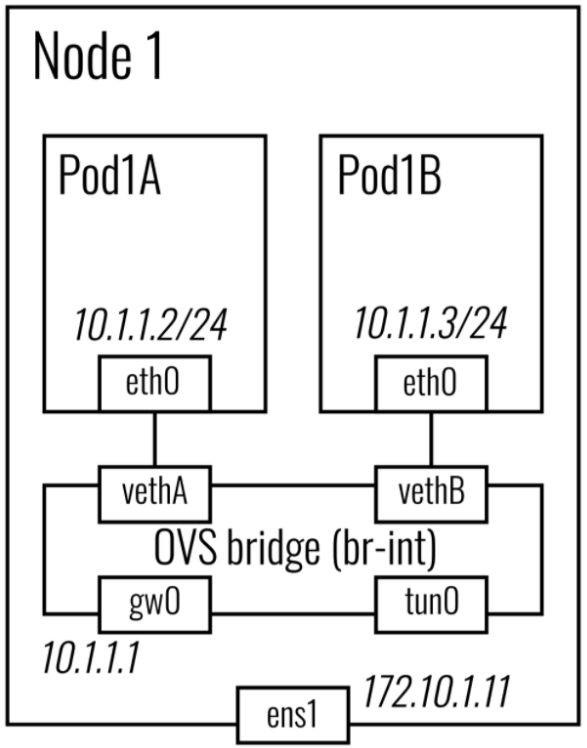
- **L3 network** for same/different subnets via BGP route sharing
- **Overlay mode** for same/different subnets using IP-in-IP/VXLAN encapsulations
- **Hybrid mode:** L2 for intra-network (uses next-hop routing) + overlay for inter-network



No bridges! Uses route tables for forwarding

iptables used for network policy enforcement

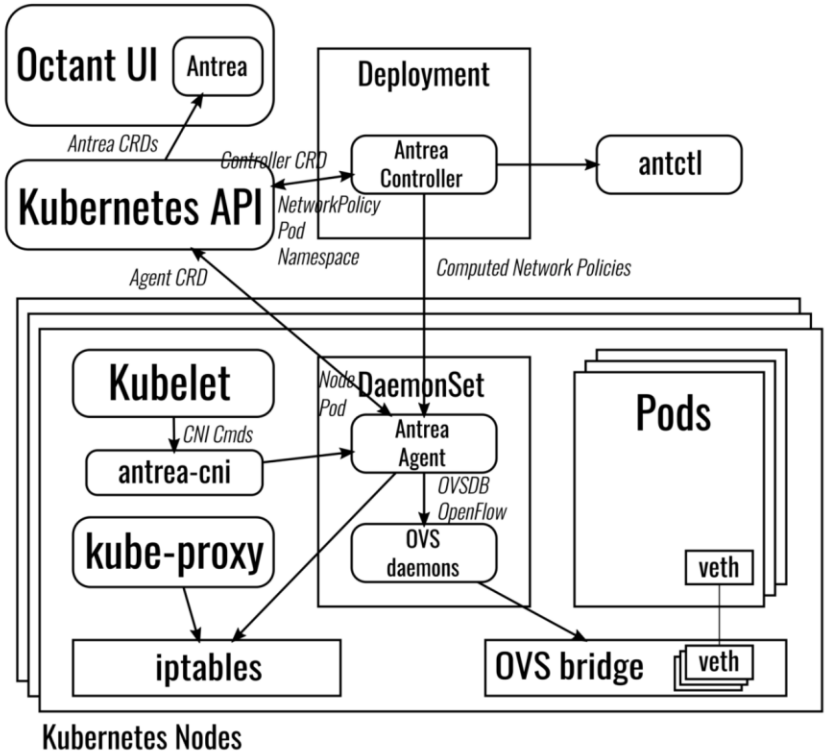
Antrea (a bridged CNI plugin)



Data plane: Open vSwitch (OVS)

Supported networking modes:

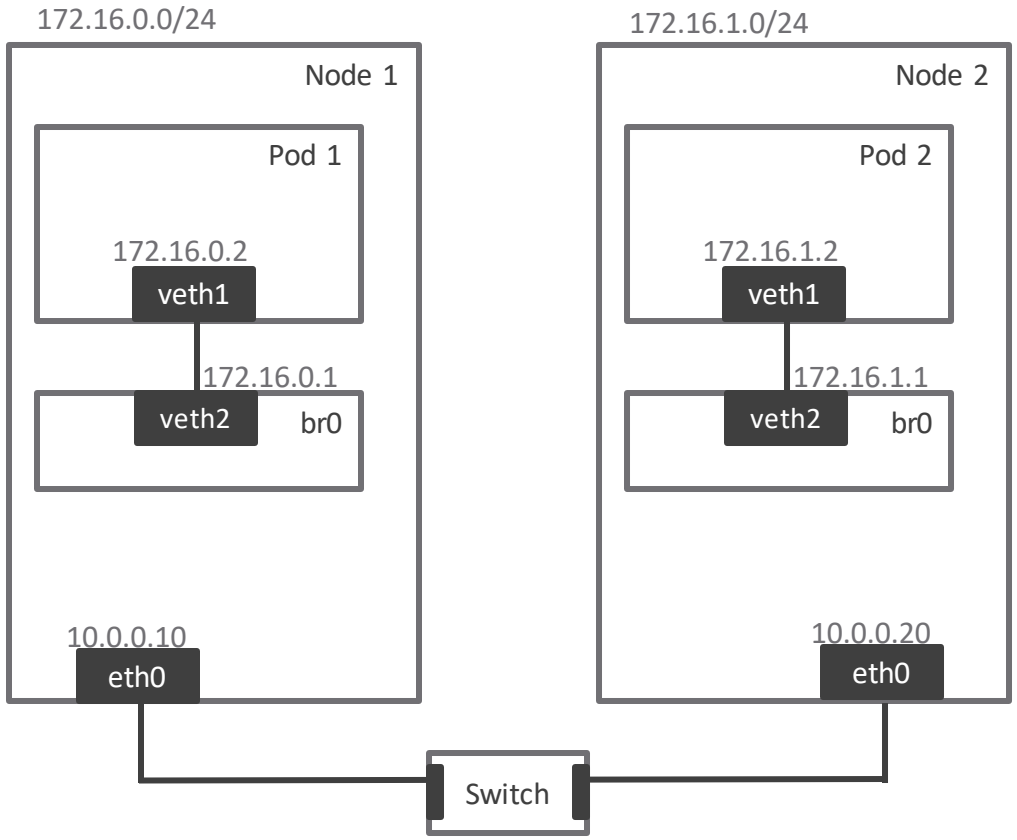
- **Overlay mode** for same/different subnets using Geneve / VXLAN / GRE / STT)
- **Hybrid mode:** L2 for intra-network + overlay for inter-network



Network policies are enforced by installing OVS flows

Non-overlay (Calico, Antrea)

No encapsulation: via ARP, route entry

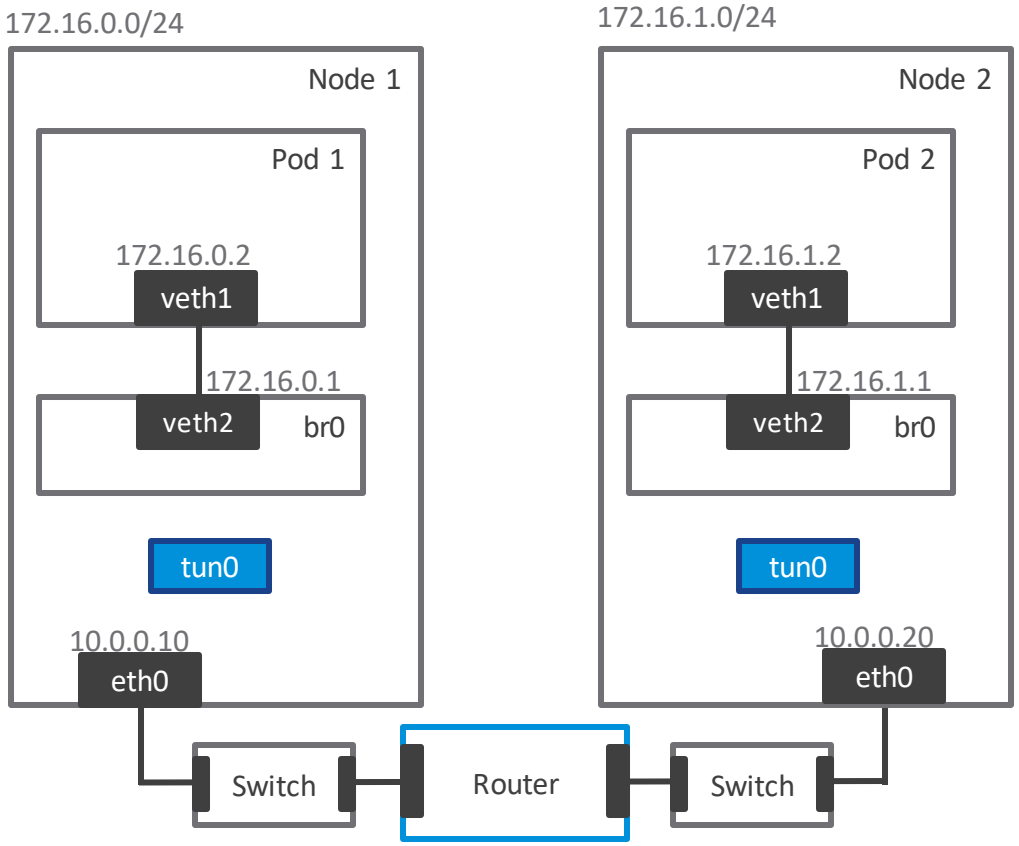


Routes
- 172.16.0.0/24 br0
- 172.16.1.0/24 via 10.0.0.20 eth0
- 10.0.0.0/16 eth0

Routes
- 172.16.1.0/24 br0
- 172.16.0.0/24 via 10.0.0.10 eth0
- 10.0.0.0/16 eth0

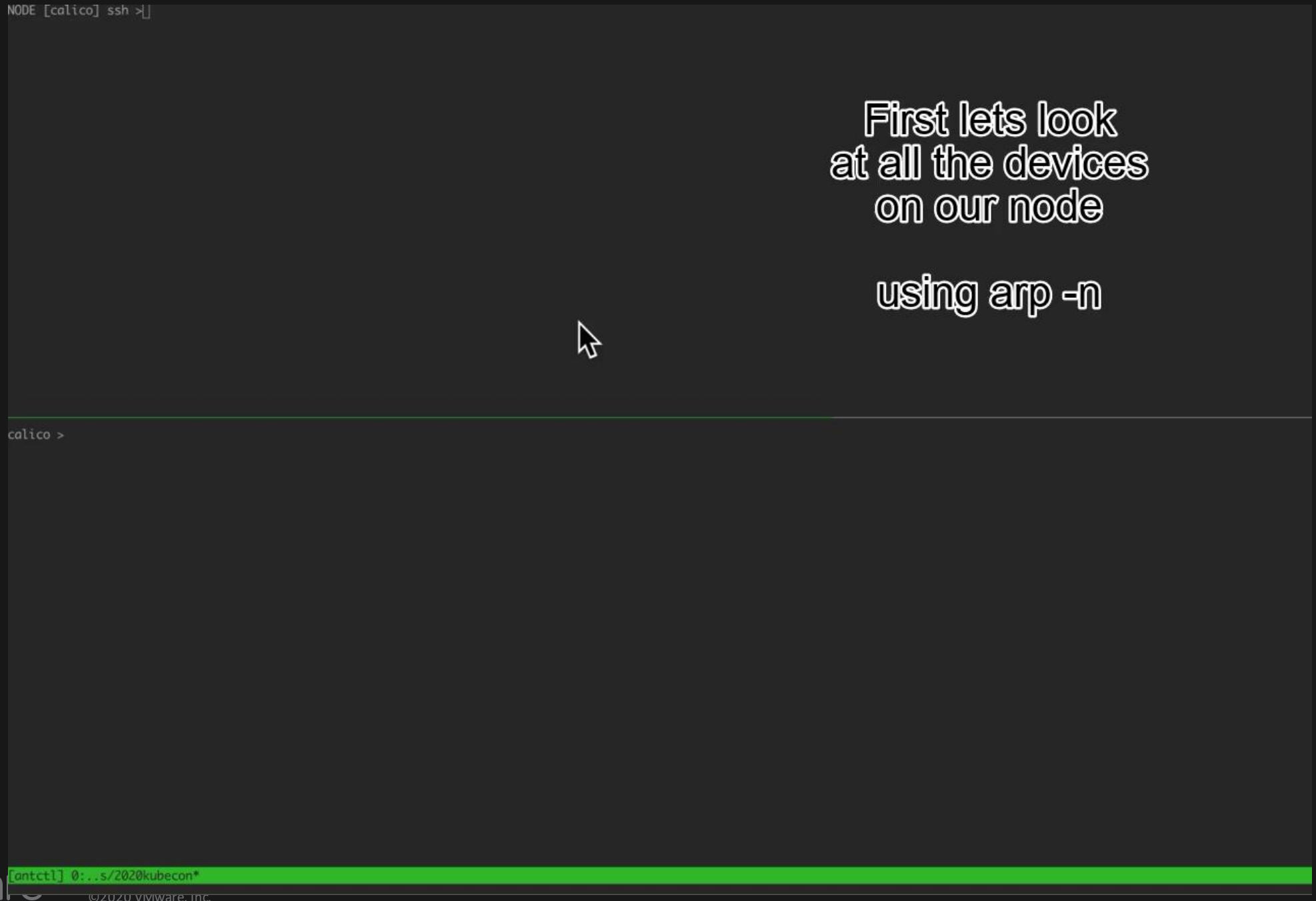
Overlay approach (Calico, Antrea)

Encapsulation: VXLAN/IP-in-IP/Geneve



Routes
- 172.16.0.0/24 br0
- 172.16.1.0/24 **tun0**
- 10.0.0.0/16 eth0

Routes
- 172.16.1.0/24 br0
- 172.16.0.0/24 **tun0**
- 10.0.0.0/16 eth0



Q&A

Thank You

Link to this talk

vmware