





## Copyright And Confidentiality

---

All content of this material, including text, images, audio, video, software, programs, tools etc., are collected online.

Visitors may use the content or services provided in this material for personal study, research or appreciation, as well as other non-commercial or non-profit uses, but at the same time comply with the provisions of the Copyright Law and other relevant laws, and may not infringe this information and related the legal rights of the right holder. In addition, any use of any content or service of this material for other purposes requires the written permission of the person and payment.



牧云/罗伟

Network | IaaS | PaaS | ServiceMesh

交流 学习 沉淀 成长 分享

[olaf.luo@foxmail.com](mailto:olaf.luo@foxmail.com)

<https://www.yuque.com/wei.luo>

<https://youdianzhishi.com>

All Rights Reserved.

Rowan Luo

交流

学习

沉淀

成长

分享

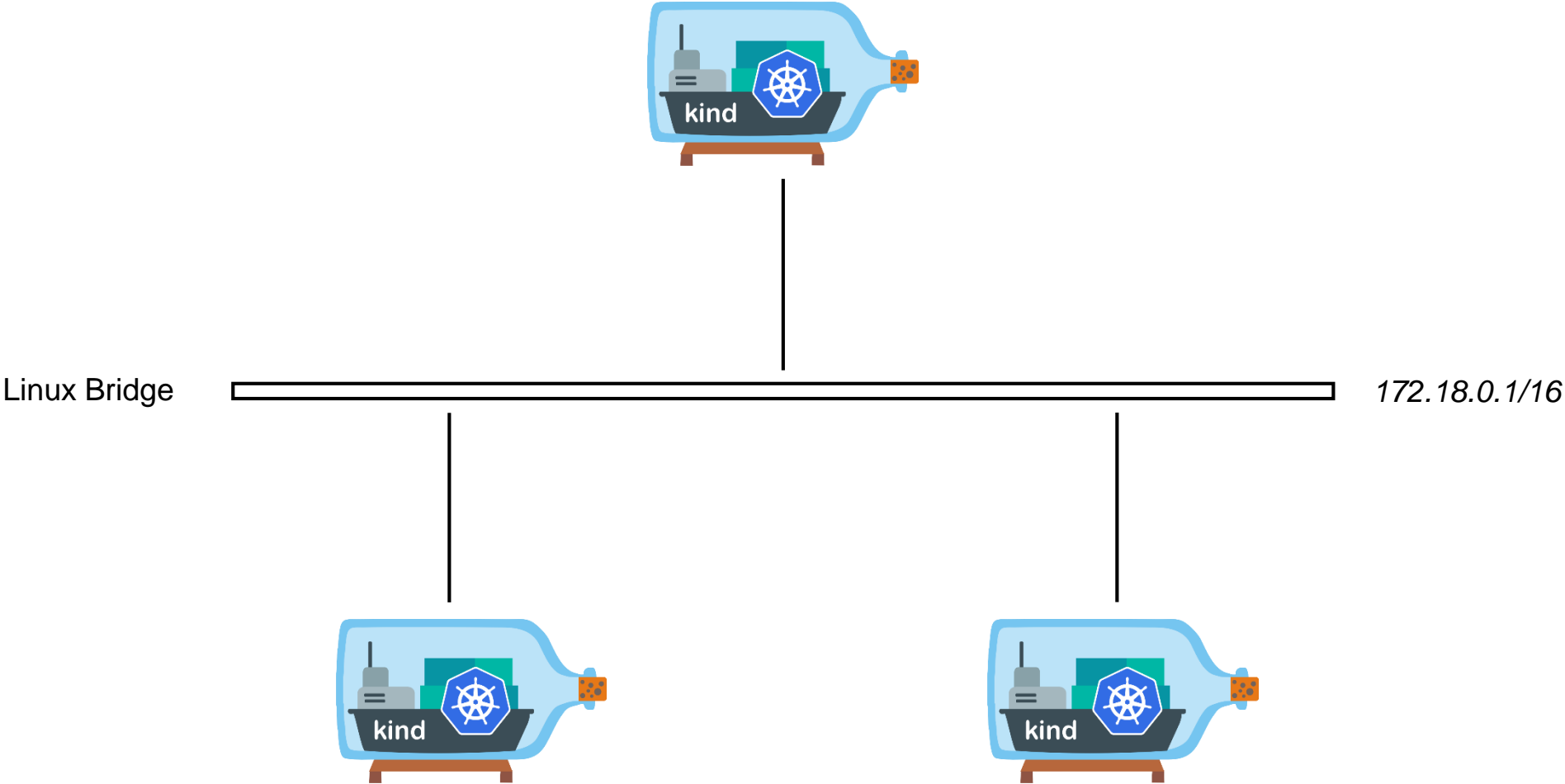
# Kubernetes CNI - Calico IPIP Mode

Revision Date: 2023/02/23

Version: v1.2

DocID: CN00002023XLW

Calico IPIP Mode - Prepare Environment



## Calico IPIP Mode - Prepare Environment

```
# 1.prep noCNI env
cat <<EOF | kind create cluster --name=calico-ipip --image=kindest/node:v1.23.4 --config=-
kind: Cluster
apiVersion: kind.x-k8s.io/v1alpha4
networking:
    disableDefaultCNI: true
nodes:
    - role: control-plane
    - role: worker
    - role: worker
containerdConfigPatches:
- |-
  [plugins."io.containerd.grpc.v1.cri".registry.mirrors."192.168.2.100:5000"]
    endpoint = ["http://192.168.2.100:5000"]
EOF

# 2.remove taints
controller_node=`kubectl get nodes --no-headers -o custom-columns=NAME:.metadata.name| grep control-plane`
kubectl taint nodes $controller_node node-role.kubernetes.io/master:NoSchedule-
kubectl get nodes -o wide

# 3. install CNI
kubectl apply -f calico.yaml

# 4. install necessary tools
for i in $(docker ps -a --format "table {{.Names}}" | grep calico-ipip)
do
    echo $i
    docker cp /usr/bin/calicoctl $i:/usr/bin/calicoctl
    docker cp /usr/bin/ping $i:/usr/bin/ping
    docker exec -it $i bash -c "sed -i -e 's/jp.archive.ubuntu.com\|archive.ubuntu.com\|security.ubuntu.com/old-releases.ubuntu.com/g'
/etc/apt/sources.list"
    docker exec -it $i bash -c "apt-get -y update >/dev/null && apt-get -y install net-tools tcpdump lrzsz >/dev/null 2>&1"
done
```

## Calico IPIP Mode - IPIP Basics

In the most general tunneling case we have:

*source ---> encapsulator -----> decapsulator ---> destination*

with the source, encapsulator, decapsulator, and destination being separate nodes. The encapsulator node is considered the "entry point" of the tunnel, and the decapsulator node is considered the "exit point" of the tunnel. There in general may be multiple source-destination pairs using the same tunnel between the encapsulator and decapsulator.

An IP tunnel is an Internet Protocol (IP) network communications channel between two networks. It is used to transport another network protocol by encapsulation of its packets.

IP tunnels are often used for connecting two disjoint IP networks that don't have a native routing path to each other, via an underlying routable protocol across an intermediate transport network. In conjunction with the IPsec protocol they may be used to create a virtual private network between two or more private networks across a public network such as the Internet. Another prominent use is to connect islands of IPv6 installations across the IPv4 Internet.

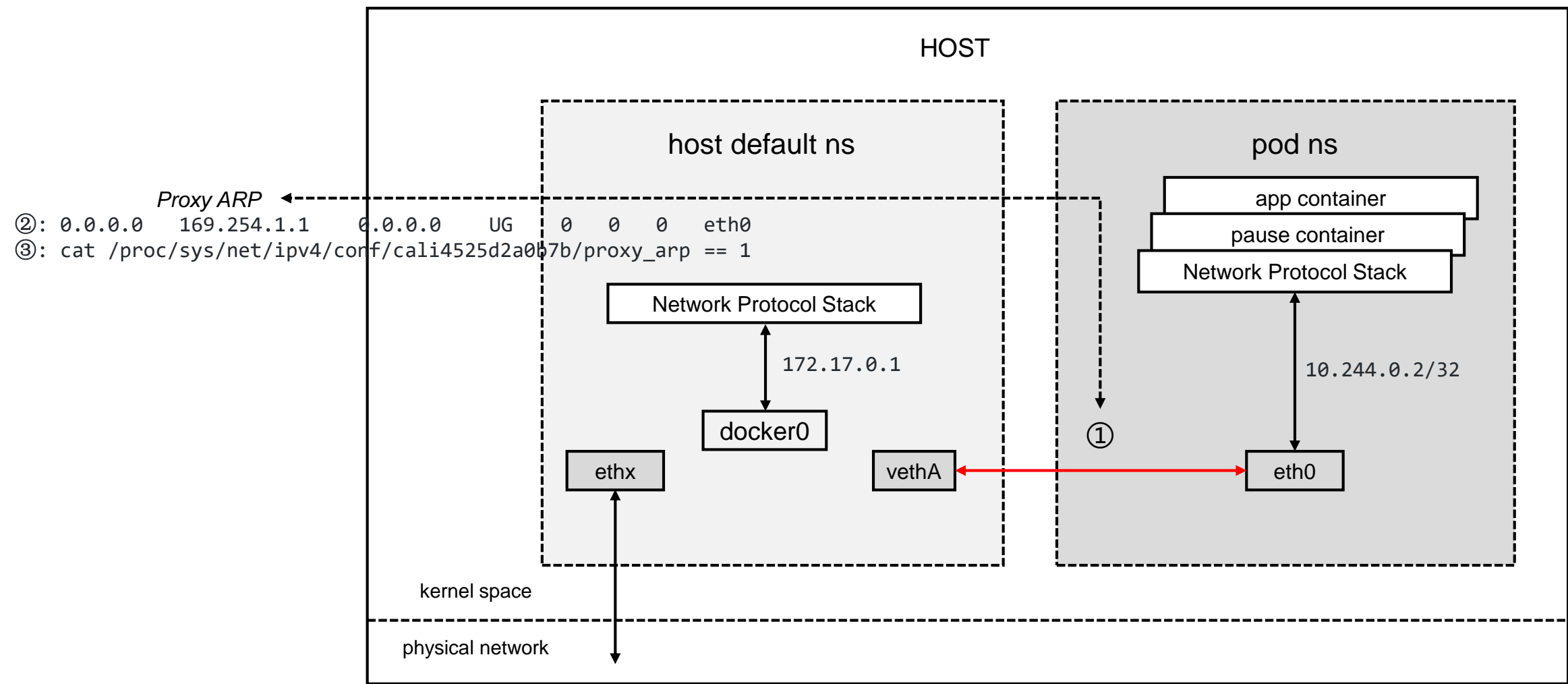
In IP tunnelling, every IP packet, including addressing information of its source and destination IP networks, is encapsulated within another packet format native to the transit network.

At the borders between the source network and the transit network, as well as the transit network and the destination network, gateways are used that establish the end-points of the IP tunnel across the transit network. Thus, the IP tunnel endpoints become native IP routers that establish a standard IP route between the source and destination networks. Packets traversing these end-points from the transit network are stripped from their transit frame format headers and trailers used in the tunnelling protocol and thus converted into native IP format and injected into the IP stack of the tunnel endpoints. In addition, any other protocol encapsulations used during transit, such as IPsec or Transport Layer Security, are removed.

IP in IP, sometimes called ipencap, is an example of IP encapsulation within IP and is described in RFC 2003. Other variants of the IP-in-IP variety are IPv6-in-IPv4 (6in4) and IPv4-in-IPv6 (4in6).

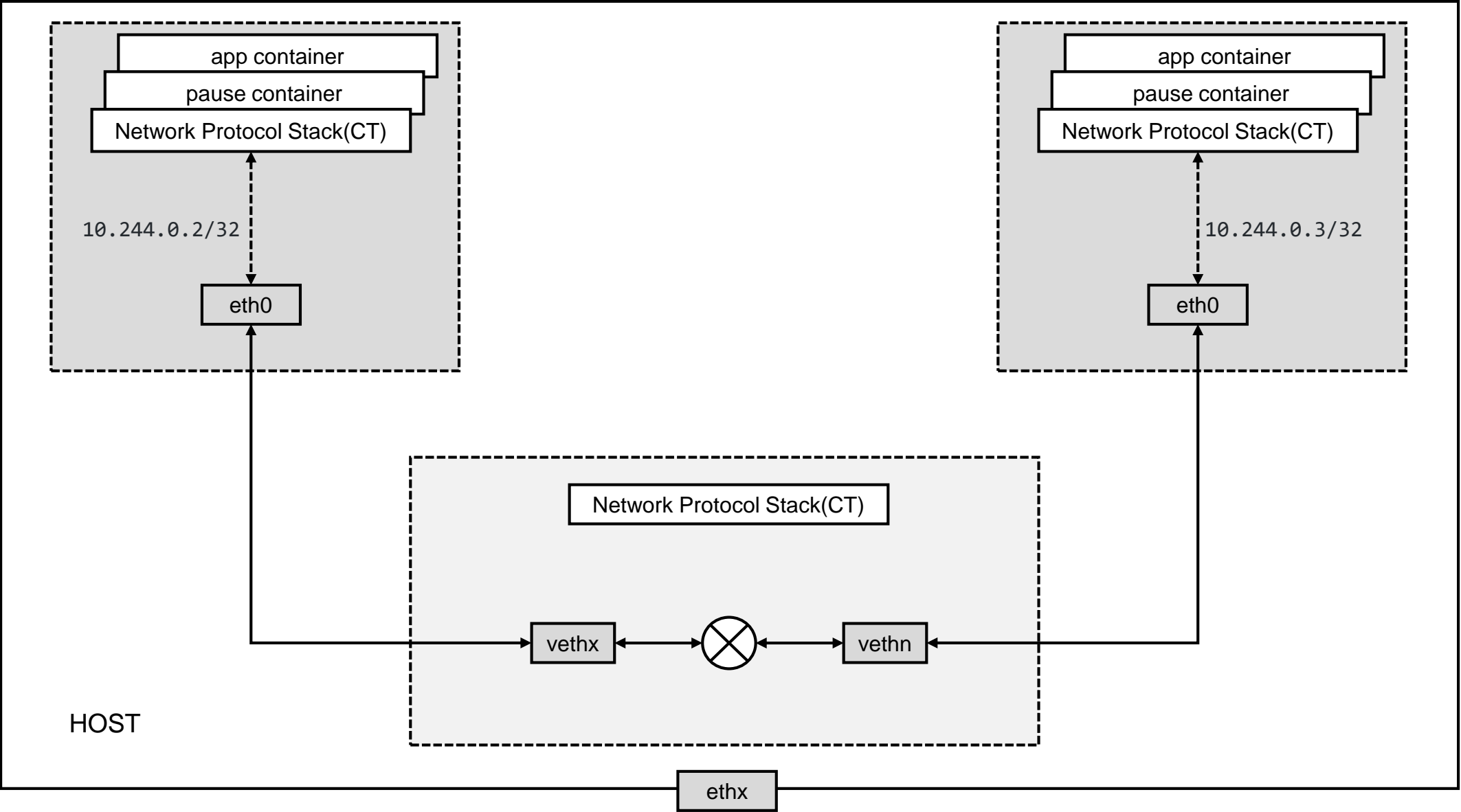
IP tunneling often bypasses simple firewall rules transparently since the specific nature and addressing of the original datagrams are hidden. Content-control software is usually required to block IP tunnels.

# Calico IPIP Mode - DataPath



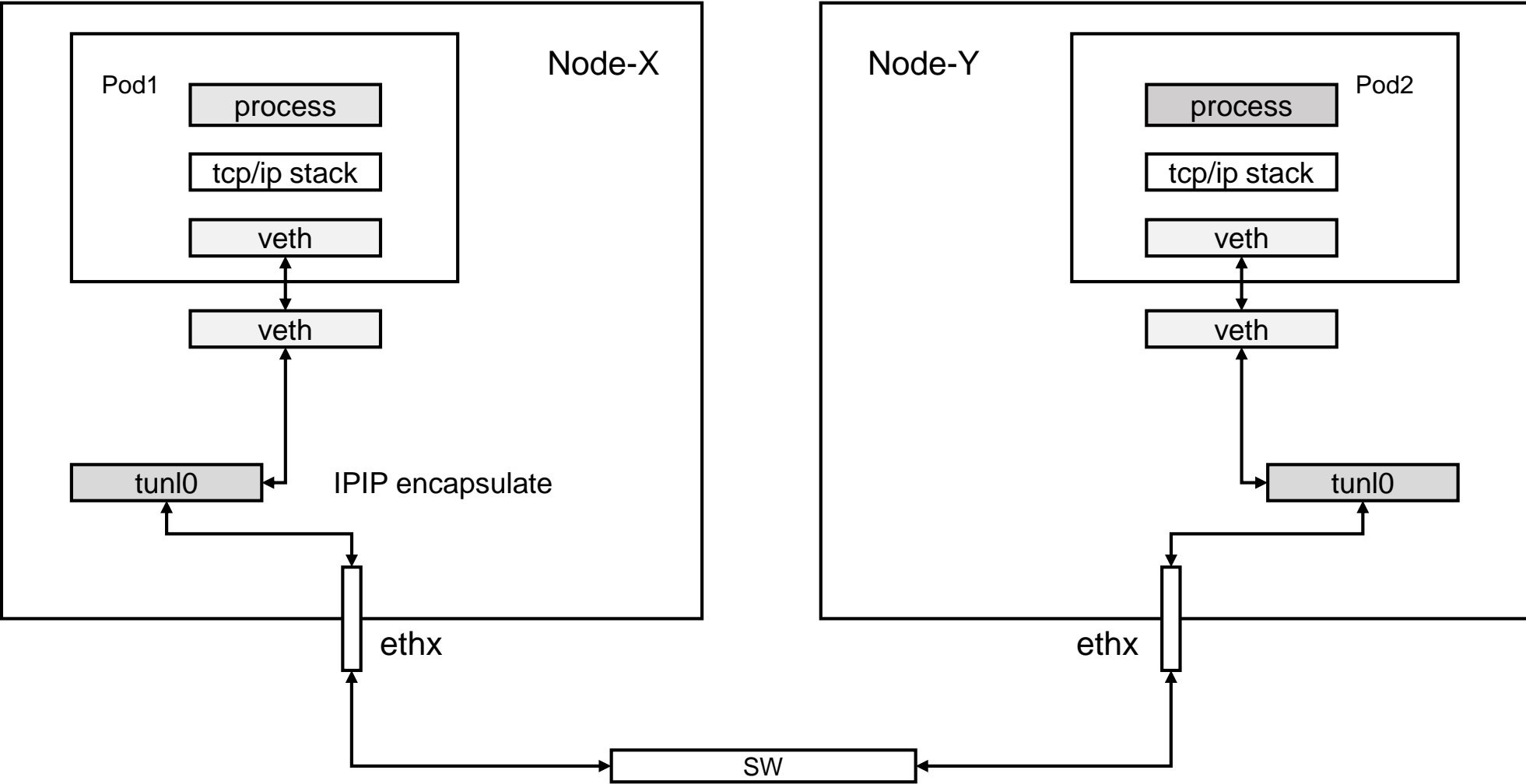
Docker spawns a container in the containers own network namespace (use the `CLONE_NEWNET` flag defined in `sched.h` when calling the `clone` system call to create a new network namespace) and later on runs a veth pair (a cable with two ends) between the container namespace and the host network stack.

Kubernetes CNI - Bridge Mode





Calico IPIP Mode - DataPath





## Copyright And Confidentiality

---

All content of this material, including text, images, audio, video, software, programs, tools etc., are collected online.

Visitors may use the content or services provided in this material for personal study, research or appreciation, as well as other non-commercial or non-profit uses, but at the same time comply with the provisions of the Copyright Law and other relevant laws, and may not infringe this information and related the legal rights of the right holder. In addition, any use of any content or service of this material for other purposes requires the written permission of the person and payment.



牧云/罗伟

Network | IaaS | PaaS | ServiceMesh

交流 学习 沉淀 成长 分享

[olaf.luo@foxmail.com](mailto:olaf.luo@foxmail.com)

<https://www.yuque.com/wei.luo>

<https://youdianzhishi.com>

All Rights Reserved.

