ons
NORTH AMERICA
OPEN NETWORKING //
Enabling Collaborative
Development & Innovation

# Secondary Network Interfaces for Containers, its Types and Use-cases

Tomofumi Hayashi, Red Hat

Hosted By
THE LINUX FOUNDATION | LF NETWORKING

# Disclaimer

The content set forth herein is Red Hat confidential information and does not constitute in any way a binding or legal agreement or impose any legal obligation or duty on Red Hat.

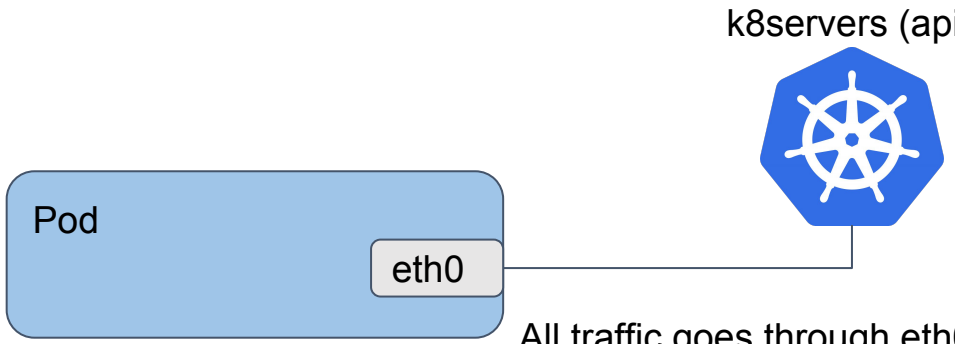
This information is provided for discussion purposes only and is subject to change for any or no reason.

# Agenda

# Secondary Network Interfaces?

- Network attachment definition (net-attach-def) CRD in Kubernetes Network Plumbing Working Group (= NPWG)
  - Spec: https://github.com/K8sNetworkPlumbingWG/multi-net-spec/
- NPWG provides multus-cni as reference plugin for network-attachment-definition
  - https://github.com/intel/multus-cni
  - Multus-cni is meta CNI plugin and it uses other CNI plugin for net-attach-def

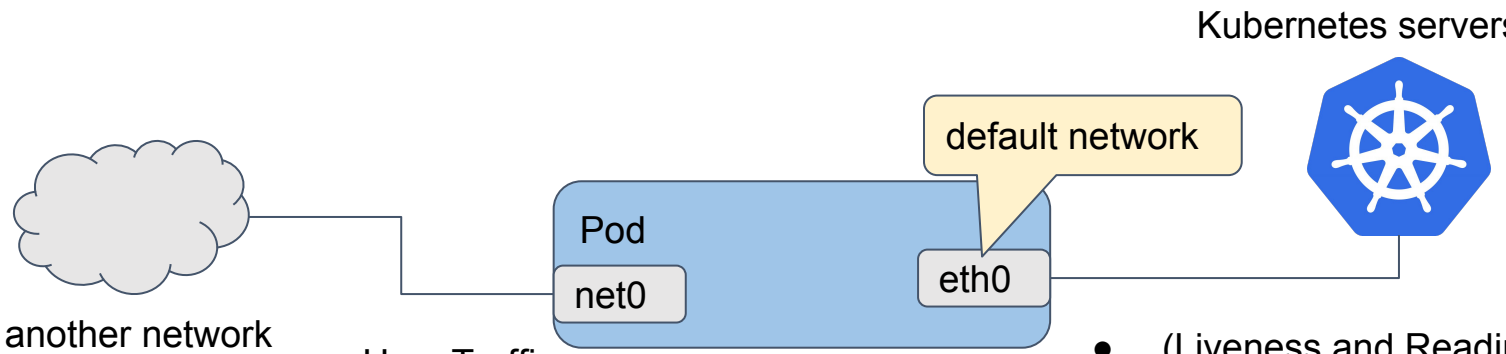
# Secondary Network Interfaces? (Cont'd)




All traffic goes through eth0

- (Liveness and Readiness) Probes
- Communication between API and Pod
- User Traffic

# Secondary Network Interfaces? (Cont'd)

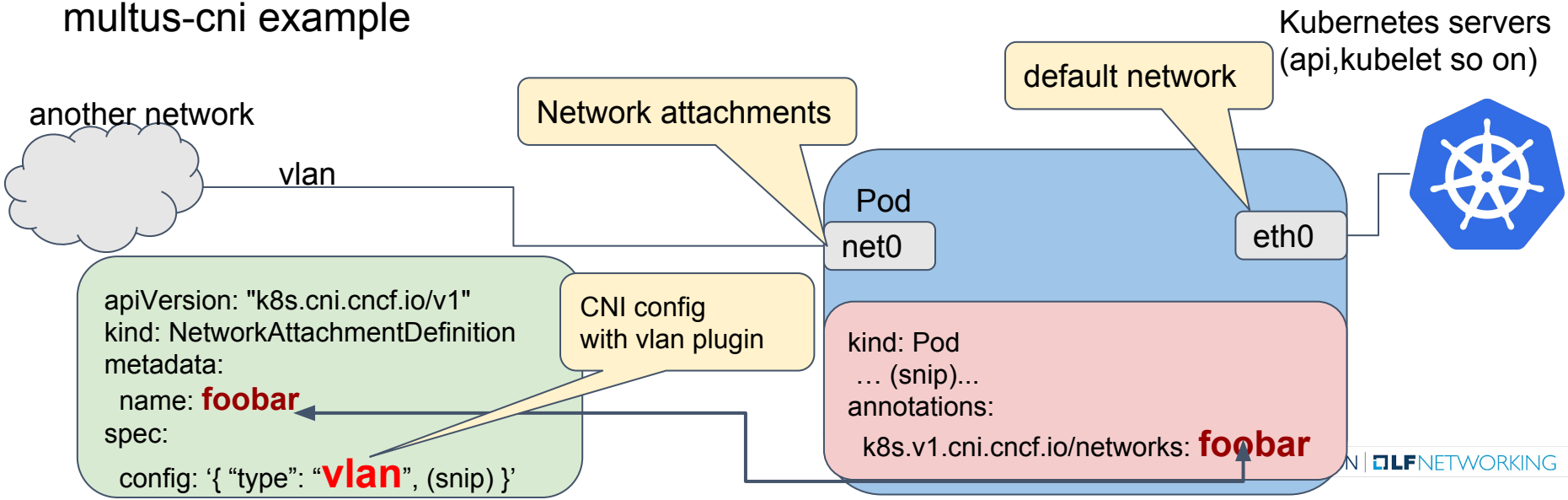



- User Traffic

- (Liveness and Readiness) Probes
- Communication between API and Pod

# Multiple Interface in Kubernetes (cont'd)

## multus-cni example



# CNI Plugins?

- CNCF CNI Reference Plugins
  - https://github.com/containernetworking/plugins/: 16 Plugins

- Plugins Type:
  - Interface Plugins: create interfaces to container
  - IPAM Plugins (IP Address Management):
    assign IP address to container interface
  - Meta Plugins:
    do something to container interface (MTU, bandwidth, so on)

# So Which Plugin is Good for Net-attach-def?

Interface Plugin
- bridge
- ptp
- host-device
- ipvlan
- macvlan
- vlan
- (flannel)
- (loopback)

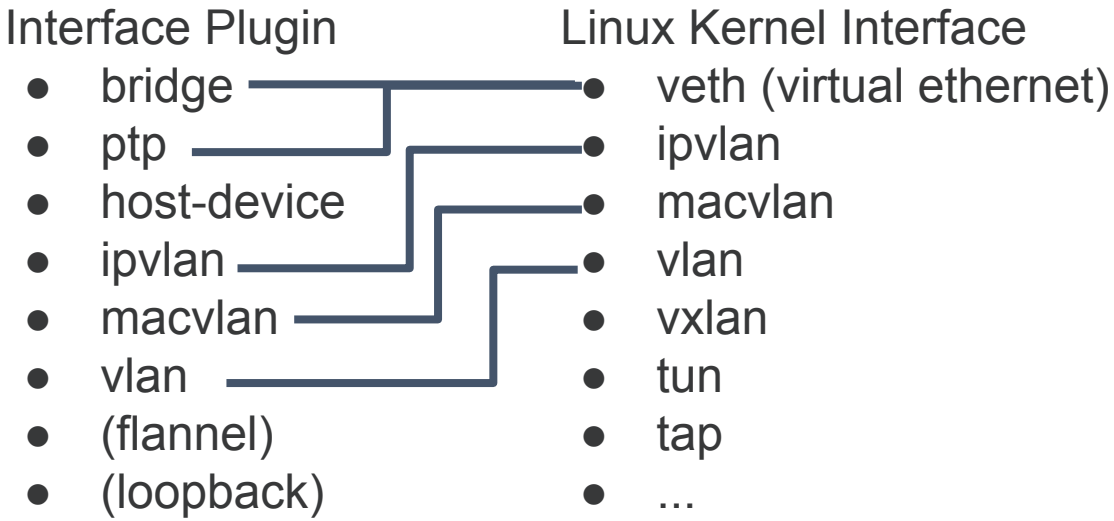
+

IPAM Plugin
- host-local
- dhcp
- static

+

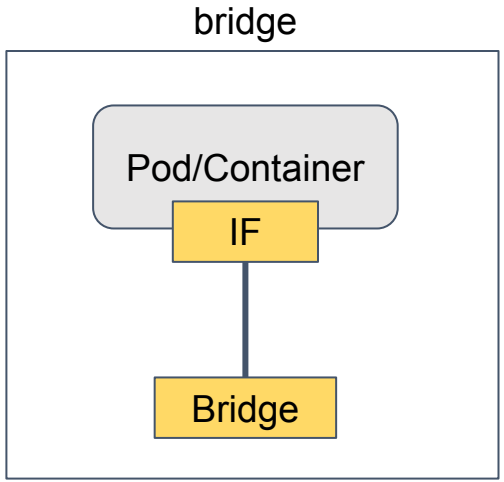
(Meta Plugin)
- (bandwidth)
- (portmap)
- (tuning)
- (sbr)

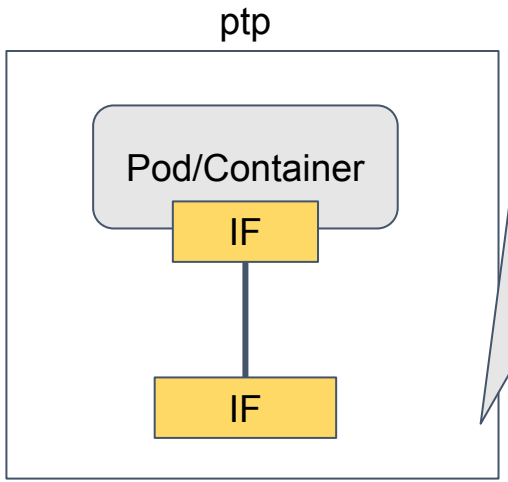
# So Which Plugin is Good for Net-attach-def? (Cont'd)



# Veth: bridge/ptp case

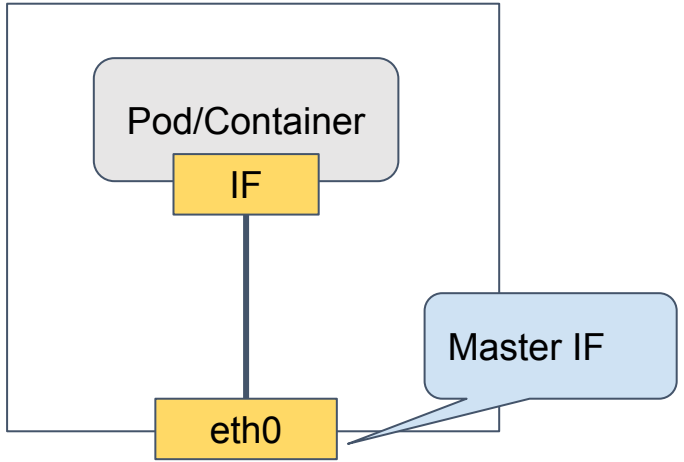
```
16: veth16b995e0@if3:
<BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
qdisc noqueue state UP group default
    link/ether 62:c6:3e:e7:e6:42 brd ff:ff:ff:ff:ff:ff
link-netnsid 0
    inet 10.1.1.1/32 scope global veth16b995e0
       valid_lft forever preferred_lft forever
    inet6 fe80::60c6:3eff:fee7:e642/64 scope link
       valid_lft forever preferred_lft forever
```



# ipvlan/macvlan: ipvlan/macvlan case

```
{
        "cniVersion": "0.3.0",
        "name": "mynet",
        "type": "ipvlan", (or “macvlan”)
        "master": "eth0",
        “mode”: “<mode>”
        "ipam": {
                "type": "host-local",
                "subnet": "10.1.2.0/24"
        }
}
```



# ipvlan/macvlan: ipvlan/macvlan case (Cont'd)

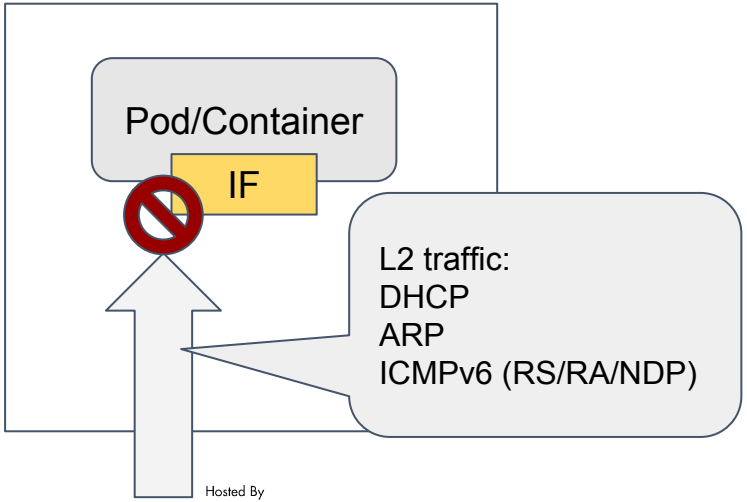
- ipvlan/macvlan is exclusive for each master interface
  - NG: macvlan@eth0, ipvlan@eth0
  - OK: macvlan@eth0, ipvlan@eth1
- macvlan uses different MAC addresses for each interface
- ipvlan uses master IF's MAC address for each interface
- macvlan/ipvlan does not send traffic to its master interface (due to Linux Kernel for additional isolation)

# ipvlan/macvlan: ipvlan/macvlan case (Cont'd)

- macvlan has ‘mode’: “bridge”(default), “private”, “vepa”, “passthru”
  - “bridge”/”vepa” supports to traffic to other macvlan IF in same node (hair-pinning)
  - “vepa” requires physical switch that support 802.1qbg
- ipvlan module support flag (from v4.15): “bridge” (default), “private”, “vepa”
- **macvlan support L2/L3 traffic**
- **ipvlan (mode: l2) support L2/L3 traffic, other mode only support L3 traffic**

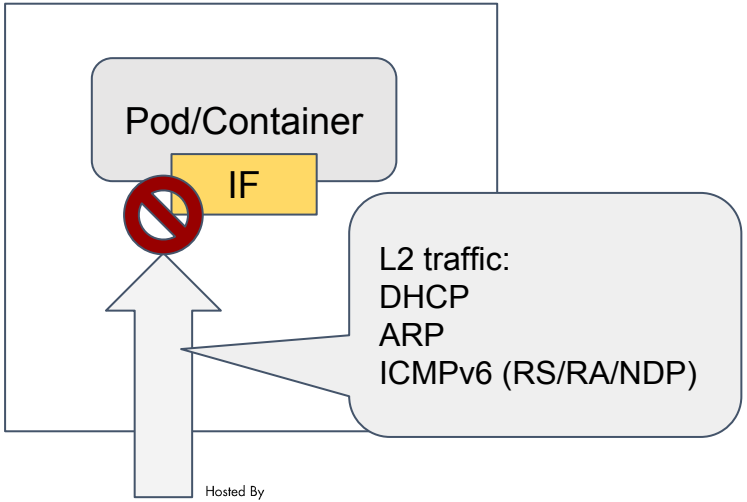
# ipvlan/macvlan: ipvlan/macvlan case (Cont'd)

- ipvlan(l3/l3s) does support L3 traffic only:
  - DHCP (IPv4) is unsupported
    - Need to use other IPAM
  - ARP is unsupported
    - Need to configure static or
    - Need to have proxy ARP



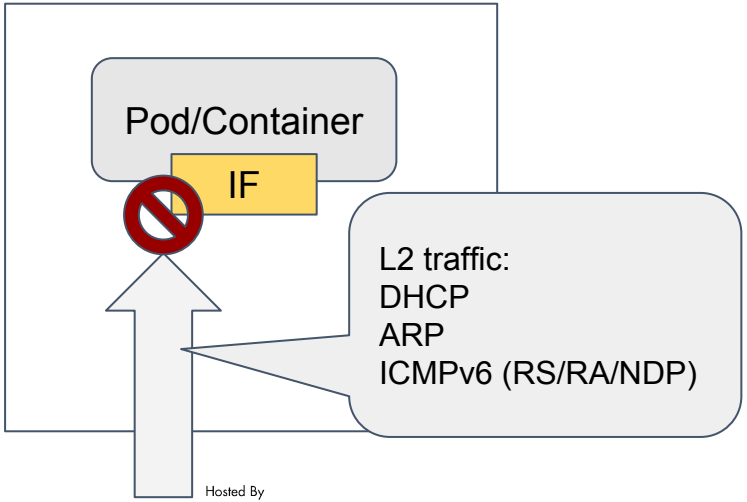
# ipvlan/macvlan: ipvlan/macvlan case (Cont'd)

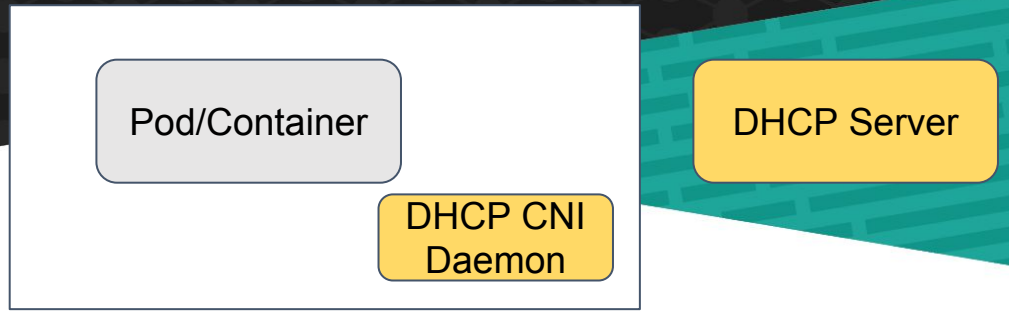
- ipvlan(l3/l3s) does support L3 traffic only:
  - ICMPv6 is unsupported
    - Need to have proxy NDP
    - Need to have some way for router discovery/address configuration
    - (DHCPv6 uses UDP, so ipvlan can get DHCPv6 packets, but DHCPv6 needs RA, hence DHCPv6 is also impossible)



# ipvlan/macvlan: ipvlan/macvlan case (Cont'd)

- ipvlan(l2) does support L2/L3 traffic:
  - DHCP (IPv4) is unsupported due to sharing MAC address for now
  - Need to wait “Client-id” features in:
    - DHCP server
    - DHCP client (= CNI plugin)






# IPAM Plugins

DHCP

- Don't forget to run DHCP Server and DHCP CNI Daemon
  - DHCP CNI Daemon for each node
  - DHCP Server for each network
- ipvlan (mode:l3/l3s) does not support DHCP
- ipvlan (mode:l2) needs some change to support client-id at
  - DHCP CNI Plugin
  - DHCP server

# IPAM Plugins (Cont'd)

static

- all interface support
- available only in 'master' branch, not released yet...

host-local

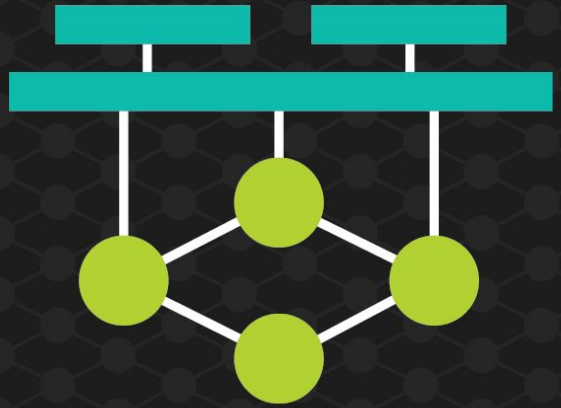
- all interface support
- (note: host-local is just "host-local"! not cluster-local!!!)

# Wrap up

Interface Plugin
- bridge
- ptp
- host-device
- ipvlan
- macvlan
- vlan
- (flannel)
- (loopback)

IPAM Plugin
- host-local
- dhcp
- static

(Meta Plugin)
- (bandwidth)
- (portmap)
- (tuning)
- (sbr)



# ons

## NORTH AMERICA

**OPEN NETWORKING //**
Enabling Collaborative
Development & Innovation

Hosted By
THE LINUX FOUNDATION | LF NETWORKING

ons
NORTH AMERICA
OPEN NETWORKING //
Enabling Collaborative
Development & Innovation

# Thanks!

ons

NORTH AMERICA

**OPEN NETWORKING //**
Enabling Collaborative
Development & Innovation

Hosted By

THE LINUX FOUNDATION | LF NETWORKING

ons
NORTH AMERICA
OPEN NETWORKING //
Enabling Collaborative
Development & Innovation

## Click to add Title

- Click to add Text

ons
NORTH AMERICA
OPEN NETWORKING //
Enabling Collaborative
Development & Innovation

## Click to add Title

- Click to add Text

- Click to add Text