BGP-EVPN to the Kubernetes Host

Running FRRouting since 2020 | Christopher Dziomba | May 2025

Kubernetes @ Deutsche Telekom

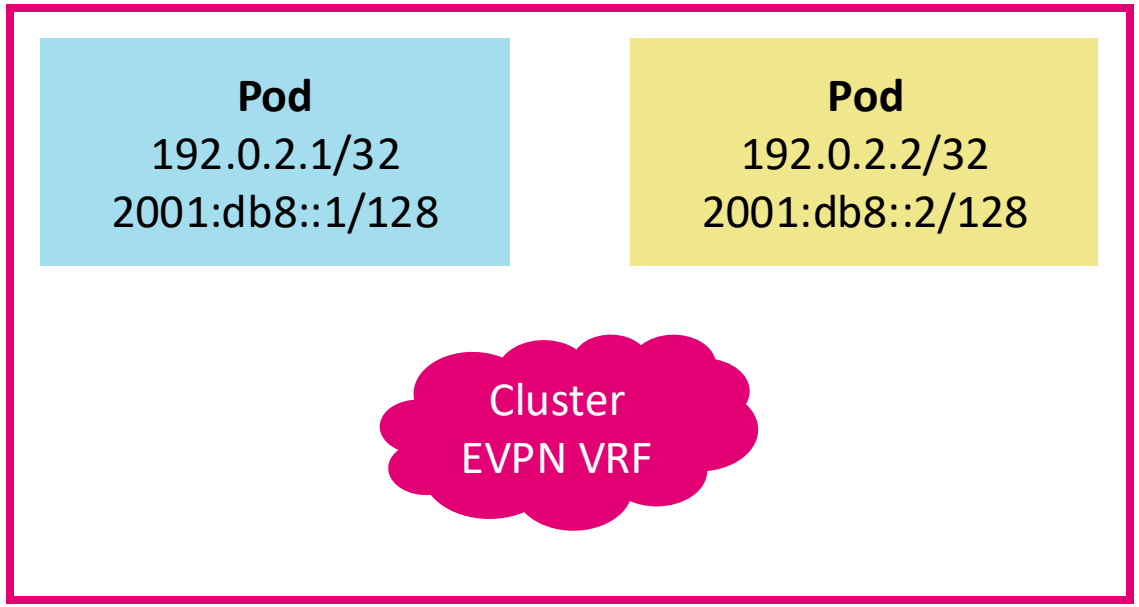
- T-CaaS (formerly “Das SCHIFF”)
- Kubernetes Platform for
  - 5G Standalone Core
  - Telephony Services
  - other CNFs
  - OSS
- Build with Vanilla Kubernetes and Open Source components
- Bare Metal platform
- Common Platform for European DT NatCos

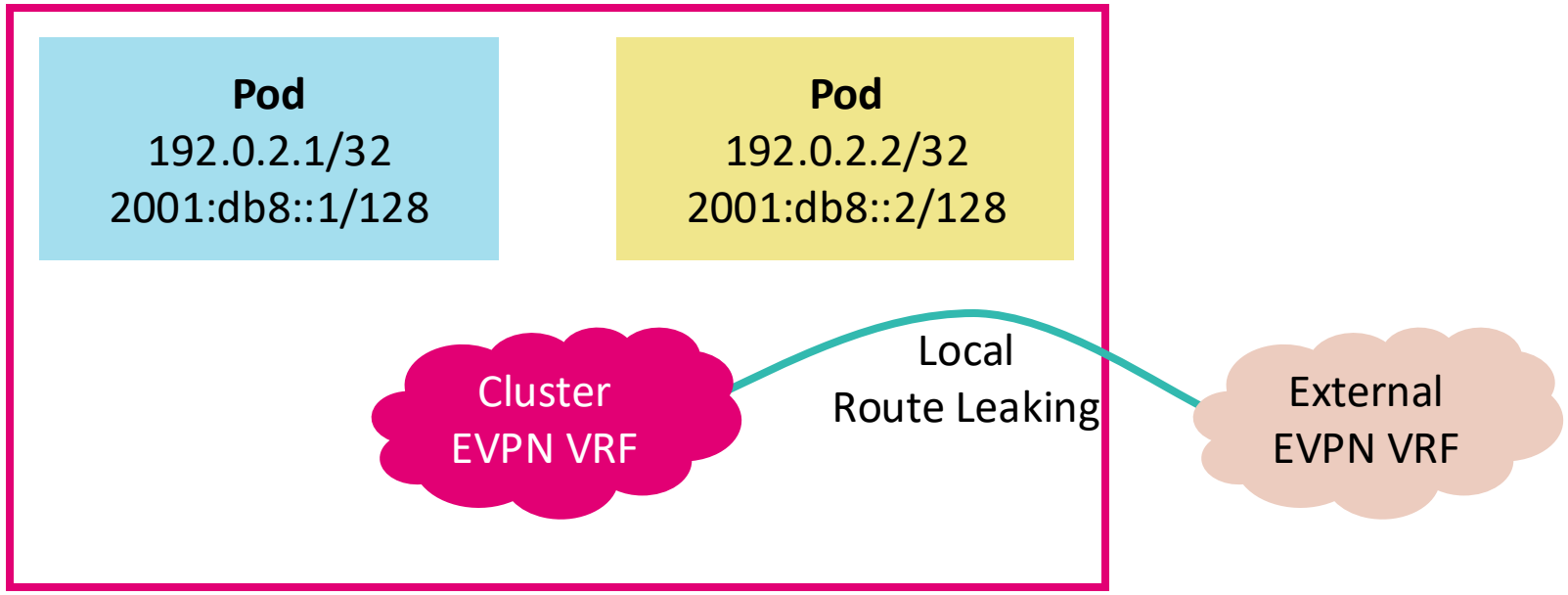
https://github.com/telekom/das-schiff

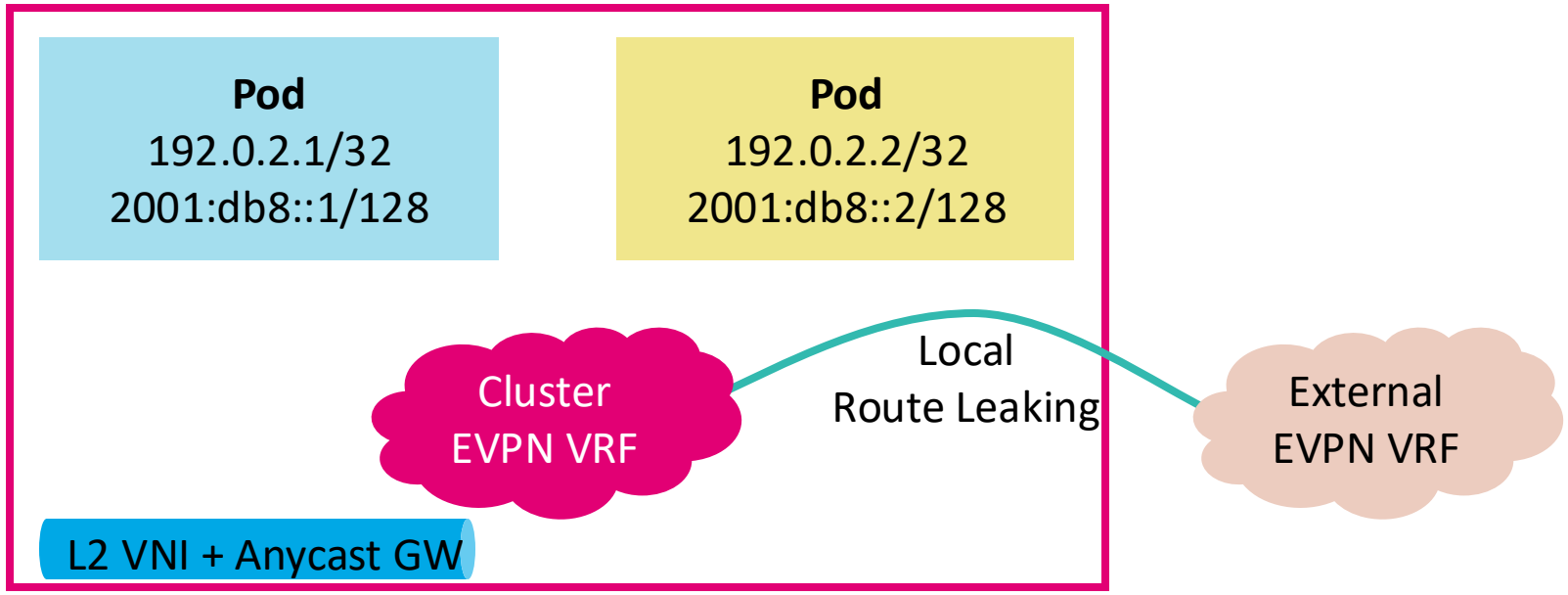
Why extend BGP-EVPN to the server

- We all 💔 BGP
- BGP in the data center well-known (RFC 7938; 2016)
- Inspired by presentation from Attilla de Groot at RIPE77 in Amsterdam (2018)
- Kubernetes Pods are essentially Layer 3 endpoints
- CNFs sometimes require secondary Layer 2 interfaces
- EVPN can solve both (Type-2 MAC-IP & Type-5 Prefix)
- Also well known for VMs (e.g. Proxmox SDN; 2019)

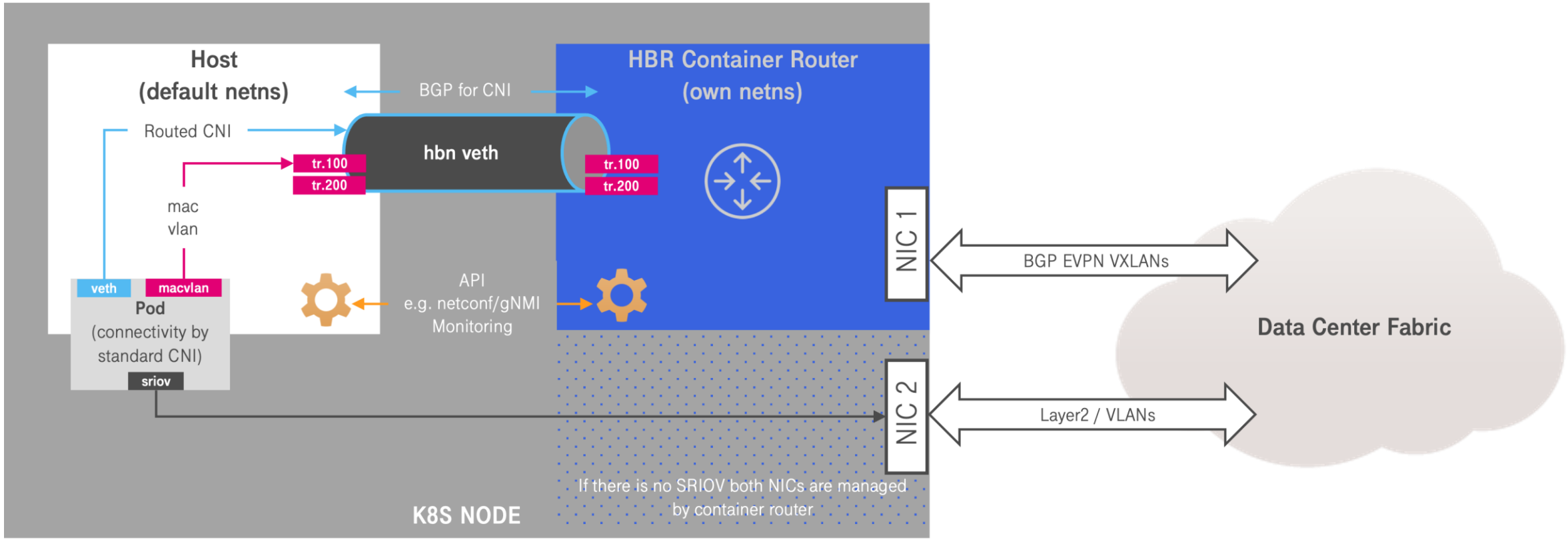
https://ripe77.ripe.net/archives/video/2001/
https://pve.proxmox.com/pve-docs/chapter-pvesdn.html


Pod
192.0.2.1/32
2001:db8::1/128
Pod
192.0.2.2/32
2001:db8::2/128
Cluster
EVPN VRF


Pod
192.0.2.1/32
2001:db8::1/128
Pod
192.0.2.2/32
2001:db8::2/128
Cluster
EVPN VRF
Local
Route Leaking
External
EVPN VRF


Pod
192.0.2.1/32
2001:db8::1/128
Pod
192.0.2.2/32
2001:db8::2/128
Cluster
EVPN VRF
Local
Route Leaking
External
EVPN VRF
L2 VNI + Anycast GW

New Design


Host
(default netns)
BGP for CNI
HBR Container Router
(own netns)
Routed CNI
hbn veth
tr.100
tr.200
tr.100
tr.200
mac
vlan
NIC 1
BGP EVPN VXLANs
veth
macvlan
Pod
(connectivity by
standard CNI)
API
e.g. netconf/gNMI
Monitoring
Data Center Fabric
sriov
NIC 2
Layer2 / VLANs
If there is no SRIOV both NICs are managed
by container router
K8S NODE

Automation

- Configuration by using “Kubernetes Custom Ressources”
- Open Source component “network-operator”
- Staged rollout (node-by-node) in new version
- Targeting pluggable architecture for BGP-EVPN router software

```yaml
apiVersion: network.schiff.telekom.de/v1alpha1
kind: Layer2NetworkConfiguration
metadata:
  name: "chris-test"
spec:
  id: 500
  vni: 3050001
  mtu: 1500
  vrf: "vrf1"
  anycastMac: "52:54:00:00:00:01"
  anycastGateways:
  - 198.51.100.1/24
  - 2001:db8::1/64
```

```yaml
apiVersion: network.schiff.telekom.de/v1alpha1
kind: VRFRouteConfiguration
metadata:
  name: t-vrf1
spec:
  vrf: vrf1
  seq: 1
  import: []
  export:
  - cidr: 198.51.100.0/24
    le: 32
    action: permit
  - cidr: 2001:db8::/64
    le: 128
    action: permit
```

https://github.com/telekom/das-schiff-network-operator

Future Work

- Finalizing Host Network-Operator 2.0 with new design
- Can be deployed on Kubernetes clusters or before, setup is a bit tricky today
- Collaboration with Sylva (Linux Foundation Europe project for Cloud Native in Telcos)
- Further evaluation of DPUs / IPUs, especially for SR-IOV

https://github.com/openperouter/openperouter - similar implementation by Federico Paolinelli (MetalLB)
https://github.com/opiproject/opi-evpn-bridge - EVPN on OPI supported network cards (e.g. Intel IPUs)
https://catalog.ngc.nvidia.com/orgs/nvidia/teams/doca/containers/doca_hbn - EVPN on NVIDIA Bluefield cards

Questions? Comments?