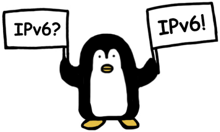
RIPE89: K8s IPv6 and Routing

ungleich



Nico Schottelius

<2024-10-31 Thu 17:00>

Introduction

- Nico working at ungleich.ch
- Many IPv6 (only) deployments
- Changed a lot of infrastructure to kubernetes in the last 3-4 years
  - Still ongoing process

Motivation (for the presentation)

- We all do routing here in this group
- Many different ways of management
  - Manual
  - Version controlled manual deployment
  - Configuration management systems
  - gitops?
- Using kubernetes (k8s) is a different way of router deployment

Motivation (for the routing in k8s)

- Why move routers into k8s in the first place?
- K8s allows a somewhat structured, somewhat consistent approach for app deployment
  - Deployments, Services, Pods, etc. are native objects in k8s
- Goal: using k8s to make routing reliable and reproducible

Kubernetes (k8s)

- Is used as a base for running infrastructure at ungleich
- Hosts various workloads from webserver, application server, mail server, etc.
- Generally speaking:
  - Private IPv4 web centric design
- Can it be used for deploying (IPv6 only) routers?

K8s standard networking: CNI

- CNI = "container network interface"
- Various implementations
  - Bridging, tunneling, encryption, . . .
  - Features dependi on the CNI implementation
- Provides one interface (one NIC)
- One of the most tricky choices to make when starting to deploy k8s

K8s multi networking = multus

- Allows to plug in multiple (virtual) network cards
- Hooks nicely into the k8s cni systems
- Reuses CNI
  - "Pod X has a NetworkAttachment of CNI type Y"
- Requires NetworkAttachment definitions that are usually cluster wide

K8s HostNetwork

- Using HostNetwork we get access to the physical machine's NICs
- Allows to easily use existing naming and needs virtually no outside configuration

Networking option summary

- CNI = one interface
- Multus = multiple interfaces
- HostNetwork = we use this for our routers

Deployment and infrastructure

- Kubernetes is usually used to form clusters of nodes
- Routers are often the backbone of the network and need to run without external dependencies
- Thus: we run routers as single node kubernetes "clusters"

External dependency: image pulling

- If image of "router" application is missing, the router cannot start
- Which route to take to to download the image?
- Need to ensure on upgrades that the image is
  - Either already present
  - Or can be locally (w/o routing daemon up) be downloaded

No default route issue

- kubeadm (a tool to provision k8s clusters) expects nodes to have a default route - otherwise upgrades and setup fails
- routers don't necessarily have that...
- Ongoing discussion, but k8s mindset far away from networking mindset

Gitops or no gitops

- Gitops = automatic deployment based on a (git) repository
- Great for automatic rollout
- Great for automatic blackouts
- At ungleich: config in git repo, rollout manually

Real world example

- Running about 20 single node k8s clusters
- Running various VPN instances in regular k8s clusters
- 90% on bare metal, ca. 10% in VMs
- Planning to move more workload into k8s virtualised workload in the near future
- In house helm chart, in house routing container

Bird container

- Simple dockerfile
- One container for routing, VPN, NAT64 capability

```
FROM alpine:3.20.1

RUN apk add bird tcpdump mtr tmux unbound \
    jool-tools \
    openvpn \
    wireguard-tools-wg wireguard-tools-wg-quick

ENTRYPOINT ["/usr/sbin/bird", "-f" ]
```

So much k8s. . .

- Routing can be done in k8s
- It's not as native as other apps
- K8s development mindset more application than networking centric
- Any questions?