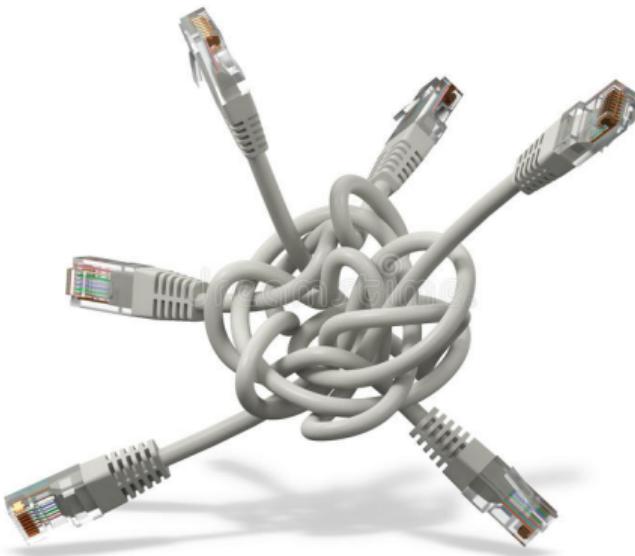


container networking from scratch



The aim

The network needs to satisfy the following (Kubernetes) requirements:

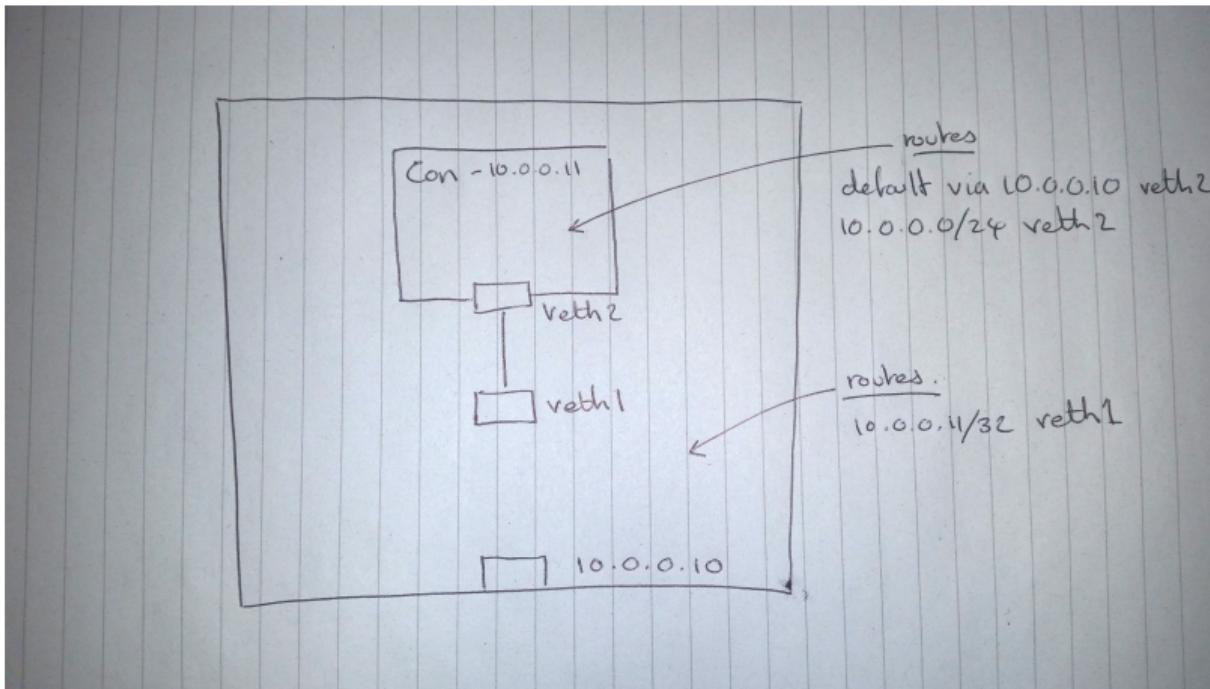
- All containers can communicate with all other containers without NAT
- All nodes can communicate with all containers (and vice-versa) without NAT
- The IP that a container sees itself as is the same IP that others see it as

The plan

To work our way from nothing, to a (flannel style) overlay network in 4 'easy' steps:

- Step 1: Single network namespace
- Step 2: Single node
- Step 3: Multi node
- Step 4: Overlay network

container networking from scratch



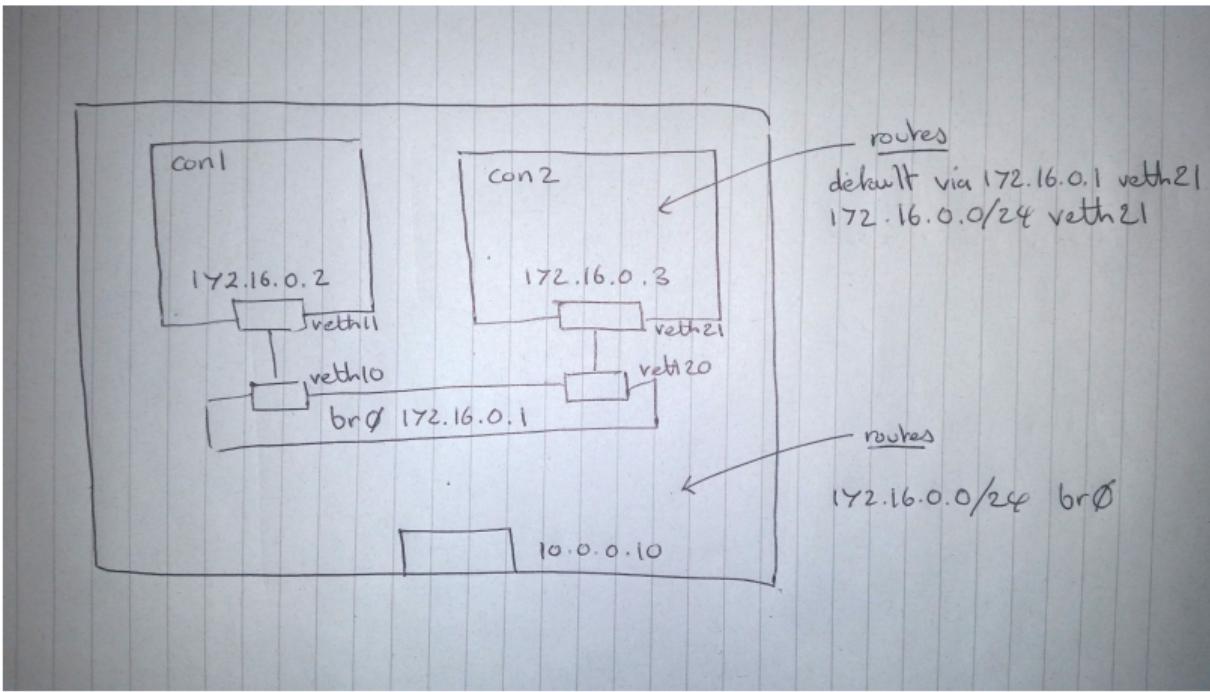
Routing rules 101

4 Types of routing rules (in order of precedence):

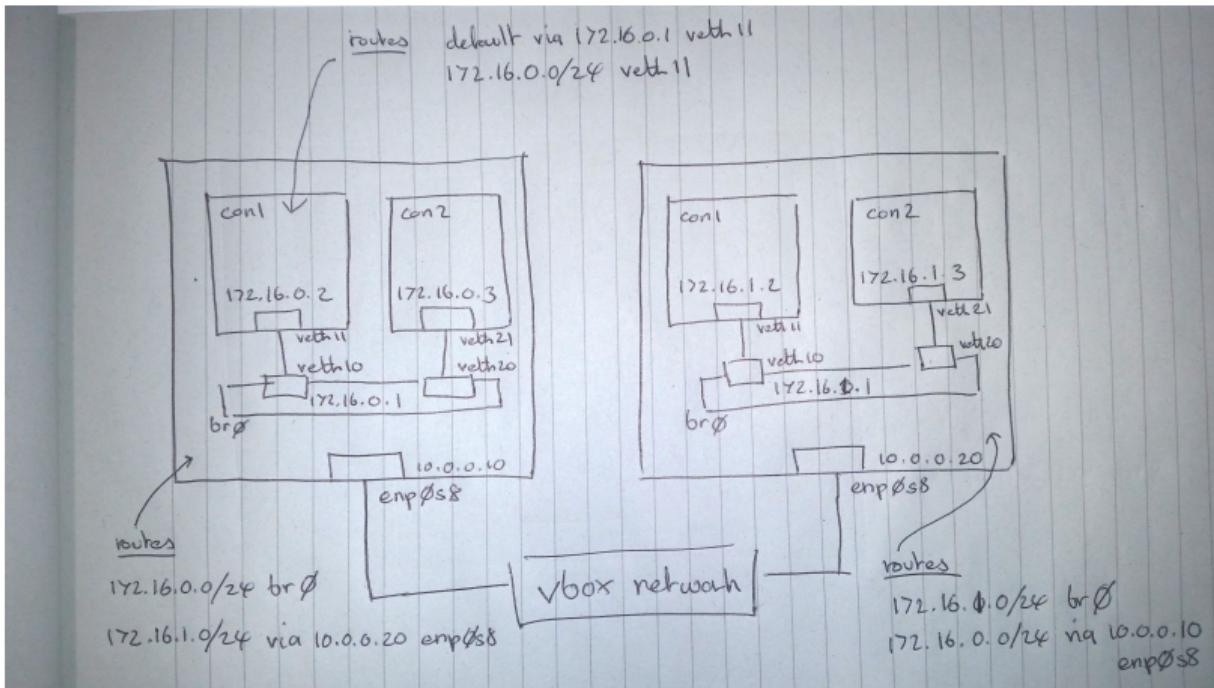
1. Directly connected network, e.g. `10.0.0.0/24 eth1`
2. Static (manually added) routing rule, e.g. `10.0.0.0/24 via 10.0.0.1 eth0`
3. Dynamic (automatically added) routing rule, e.g. `10.0.0.0/24 via 10.0.0.1 eth0`
4. Default rule, e.g. `default via 10.0.0.1 eth0`

Within each of the above, the most specific CIDR range takes priority.

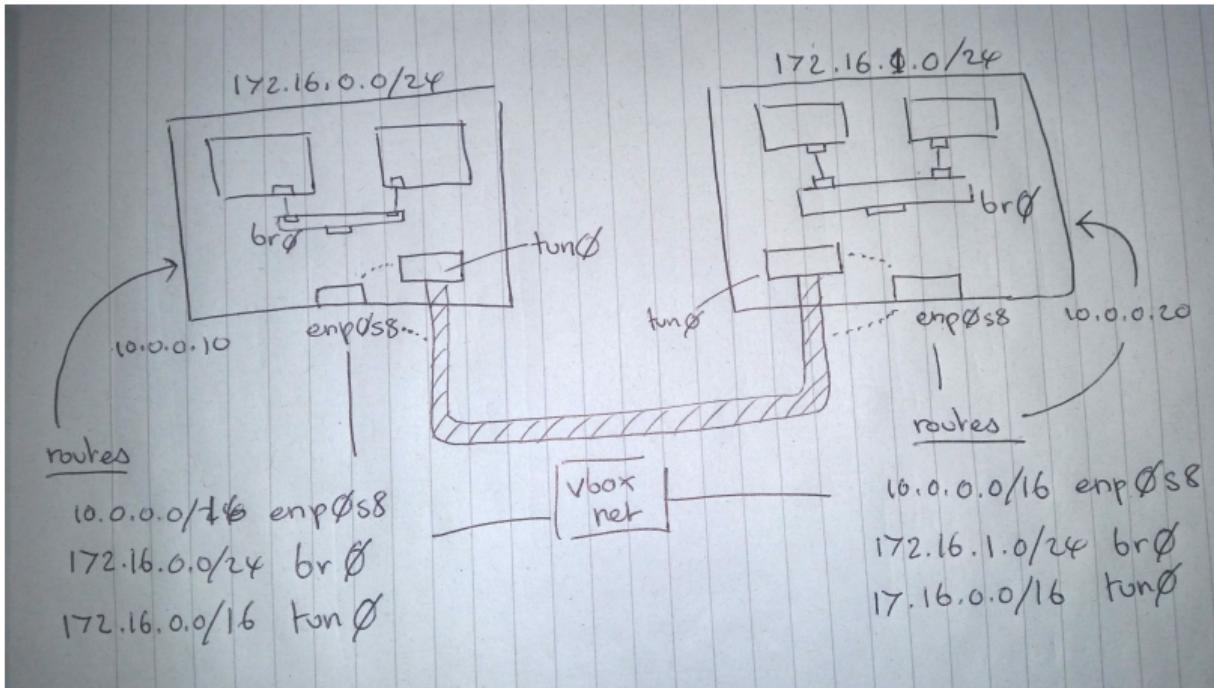
container networking from scratch



container networking from scratch



container networking from scratch



Putting it all together

1. *Flannel*

- *host-gw*: Step 3.
- *udp*: Step 4.
- *vxlan*: Step 4, but implemented in the kernel => more efficient!
- *awsvpc*: Sets routes in AWS.
- *gce*: Sets routes in GCE.
- Node->pod-subnet mapping stored in *etcd*.

2. *Calico*

- No overlay for intra L2. Uses next-hop routing (step 3).
- For inter L2 node communication, uses IPIP overlay.
- Node->pod-subnet mappings distributed to nodes using BGP.

3. *Weave*

- Similar to *Flannel*, i.e. uses *vxlan* overlay for connectivity.
- No need for *etcd*. Node->pod-subnet mapping distributed to each node peer to peer.

container networking from scratch

The screenshot shows a GitHub repository page for 'kristenjacobs / container-networking'. The repository title is 'kristenjacobs / container-networking'. The repository summary is 'Container networking from scratch, from a single namespace to an overlay network.' It has 40 commits, 3 branches, 0 releases, and 1 contributor. The latest commit was 16 minutes ago. The repository has 1 star and 0 forks. The repository page includes a list of recent commits:

Commit	Message	Time Ago
Added routing rules 101 slide	kristenjacobs	Latest commit 4329581 16 minutes ago
1-network-namespace	Consistency updates	22 hours ago
2-single-node	Consistency updates	22 hours ago
3-multi-node	Consistency updates	22 hours ago
4-overlay-network	Consistency updates	22 hours ago
slides	Added routing rules 101 slide	16 minutes ago
.gitignore	Added git ignore	17 days ago
README.md	Removed the single multinode L2 network example.	2 months ago

container networking from scratch

C I D R
(classless Inter-Domain Routing)

$x \cdot x \cdot x \cdot x/y$

$x \cdot x \cdot x \cdot x$ = IP address

y = number of bits in the network mask

$32 - y$ = number of bits in the host identifier

e.g. 192.168.100.14/24

The network mask is 192.168.100.0

This is host 14 out of a possible 254

(0 is the network ID,
255 is the broadcast address).

container networking from scratch

