

<https://pad.n39.eu/qlvTIM7hQnKZ9jzdo6SZbg#>

Brainstorming Server durch Mini-Cluster ersetzen

Teilnehmer

- David
- Tux

Notizen

- handvoll raspi 4B
 - 3 + 3
- Micha hat Intel NUC übrig
- Hardware Loadbalancer?
- dedizierter Switch - 16 Ports k8s-kompatibel
- Powerbudget ausrechnen
- SAN Lösung
 - nfs CSI Treiber
 - (S3?)
- kleinere USV Lösung

erste Schritte

- k3s Cluster auf VMs - Stefan hat ein Ansible Playbook
- 3 control plane (k3s-c[1-3])
- 3 worker (k3s-w[1-3])
- an einem der Knoten NFS mit passendem label, dort den NFS Provider hosten

Infrastruktur

Maschinen

- Control Plane
 - k3s-c1: 172.23.52.130
 - k3s-c2: 172.23.52.131
 - k3s-c3: 172.23.52.132
- Workers
 - k3s-w1: 172.23.52.140
 - k3s-w2: 172.23.52.141
 - k3s-w3: 172.23.52.142

DNS/DHCP-Einträge: siehe [descartes/PR #38](#)

Konfiguration

https://git.n39.eu/Netz39_Admin/netz39-infra-k8s

Netzwerk

- Herausforderung: Netzwerkverkehr an die richtigen Nodes senden, ohne dass einer der Knoten den gesamten Traffic durchleiten muss. (Insbesondere RPi ist nicht besonders performant, wenn es um Netzwerk geht.)
- Vorschlag: [kube-vip](#)
 - leaderElection per Service
 - IP-Adressen an LoadBalancer binden
 - lokale Routing-Policy verwenden
 - Wenn pro Service (LoadBalancer) eine eigene IP-Adresse vergeben wird, kann der Switch mit ARP Routing den Netzwerkverkehr direkt an die Knoten senden, auf denen die LBs laufen.
 - Wir brauchen dann keinen speziellen Switch

Storage

- Lokale Platte an einem der Worker Nodes
 - k3s-w1 hat 256gb zusätzliche Platte angehängt
 - label node-role.kubernetes.n39.eu/nfs-host: true
- csi-driver-nfs - <https://github.com/kubernetes-csi/csi-driver-nfs/tree/master>

From:

<https://wiki.netz39.de/> - **Netz39**

Permanent link:

<https://wiki.netz39.de/projects:2023:greenit>

Last update: **2023-09-29 07:23**

