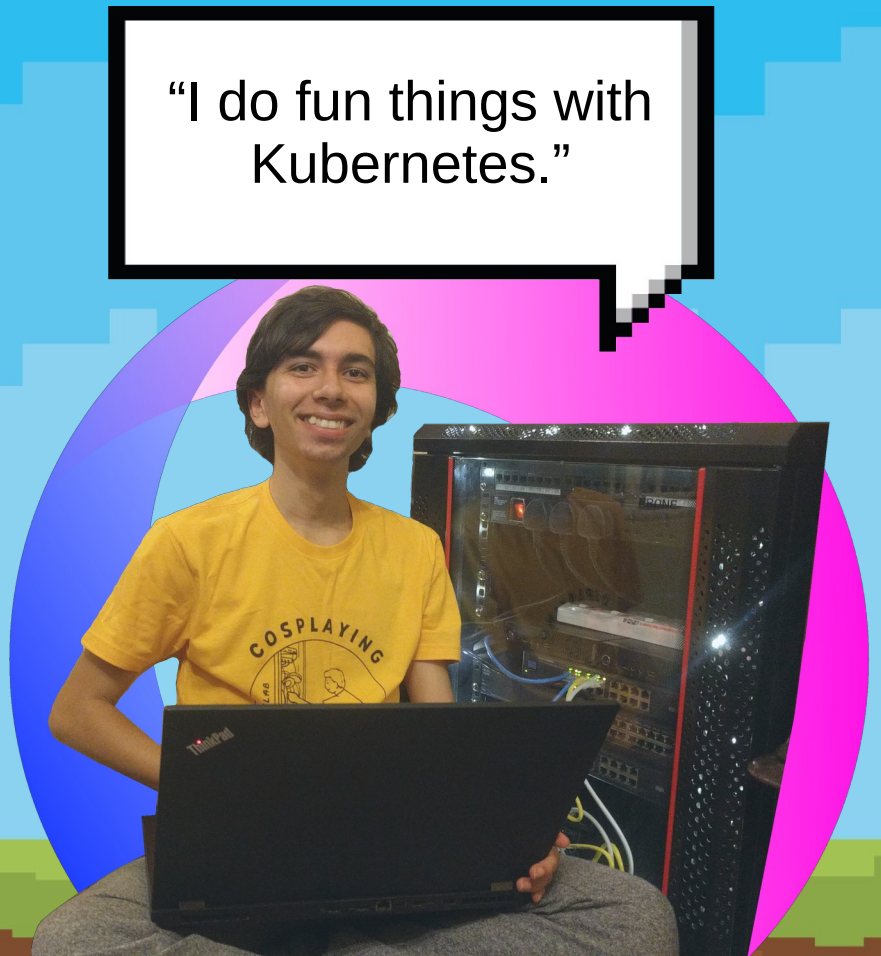
Krafting the Cloud

Building a Free, Open, and Accessible Cloud

By Alex Bissessur

alex.yaml

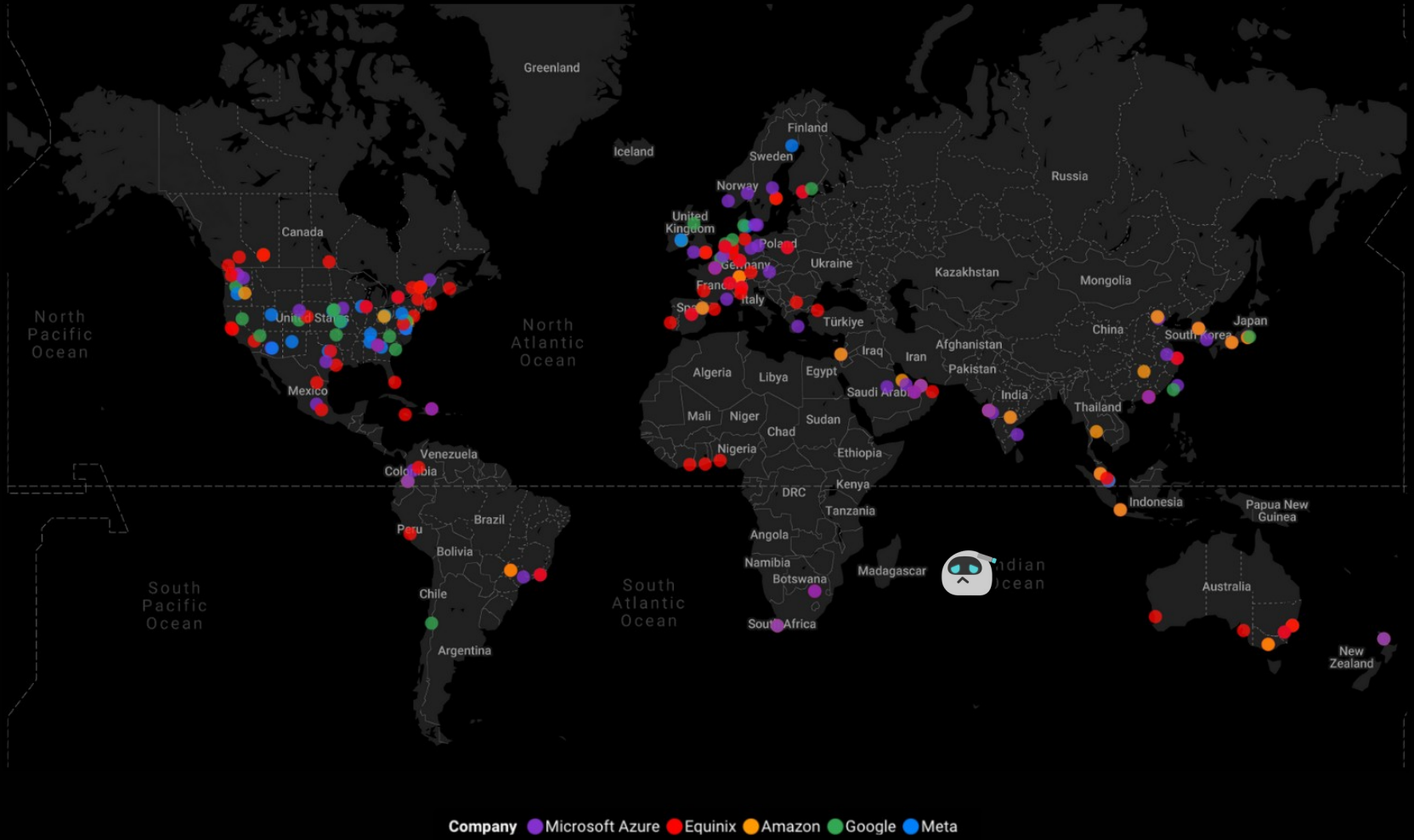
```yaml
---
apiVersion: v1
kind: Person
metadata:
 name: Alex Bissessur
spec:
  work:
    company: La Sentinelle
    role: Kubernetes Person
    location: Mauritius
  contact:
    website: alexbissessur.dev
    mastodon: moris.social/@AlexB
    github: github.com/xelab04
  interests:
    - Kubernetes
    - Linux
    - Free & Open Source Software
  hobbies:
    - Playing kubectl with Homelab
```




SCC
DEVELOPERS CONFERENCE
2025
PLATINUM TIER
SWAN
Google
Axiz
Microsoft
NETWORKING HOUR
spoon consulting
GOLD TIER
INTERNET SPONSOR
KALDERA
MEDIA PARTNER
LANYARDS & BADGES SPONSOR
cloud.mu

Placeholder for my Kubernetes workshop at DevCon 2024

Data Center Locations (Microsoft, Equinix, Amazon, Google, Meta)



How do we build a cloud?

CSP requirements

- Isolation – different users must stay separate and unable to interact
- Cloud – it must run on someone else’s computer
- Self-managed – I do not provision VMs manually

Virtual Clusters

- KinD – run Kubernetes in Docker, perfect for testing
- Virtual clusters can be deployed to an existing K*s cluster – Kubernetes in Kubernetes
- Existing options:
  - VCluster, Loft Labs
  - K3k, Rancher/SUSE

K3k and VCluster

- Provision virtual clusters on a host
- Virtual cluster lives in pods, with the API server exposed
- Option to passthrough host resources, such as: storageclass, ingressclass, configmaps, secrets
- Virtual clusters are isolated across different namespaces

KRaft

Pronunciation: /krɑːft/

Origin:

- K for Kubernetes
- Raft, compared to the Kubernetes ship
- Craft, for building and creating

Built with:

- K3k
- Longhorn
- Traefik

How is KRaft Made?

- Kraft is split into ‘microservices’
- Auth service for login, registration, pwd reset
- Cluster service to provision clusters, expose the API, and retrieve kubeconfig file
- Resource service to retrieve cluster metrics
- Frontend of static website with Tailwind & Alpine
- MariaDB

KRaft

```
NAME                                    CPU(cores)   MEMORY(bytes)
kraft-auth-7c4646b85c-m76jf             1m           5Mi
# written in Rust
kraft-cluster-manage-6cf9d74c9b-hnrmv   1m           9Mi
# also written in Rust
kraft-db-8566c5b599-4mtrb               1m           176Mi
# simple MariaDB pod
kraft-frontend-68d8c7698c-z5nck         0m           7Mi
# nginx with static files
kraft-resource-manage-7d7d979fc4-5qlwz  1m           290Mi
# python container, Flask
```

KRaft by Alex

> Create a cluster

> View clusters

> ReadMe.md

> Account

KRaft by Alex

> Account

email: dummy@user.com

username: dummyuser

Sign out

> Password

Password :

New Password :

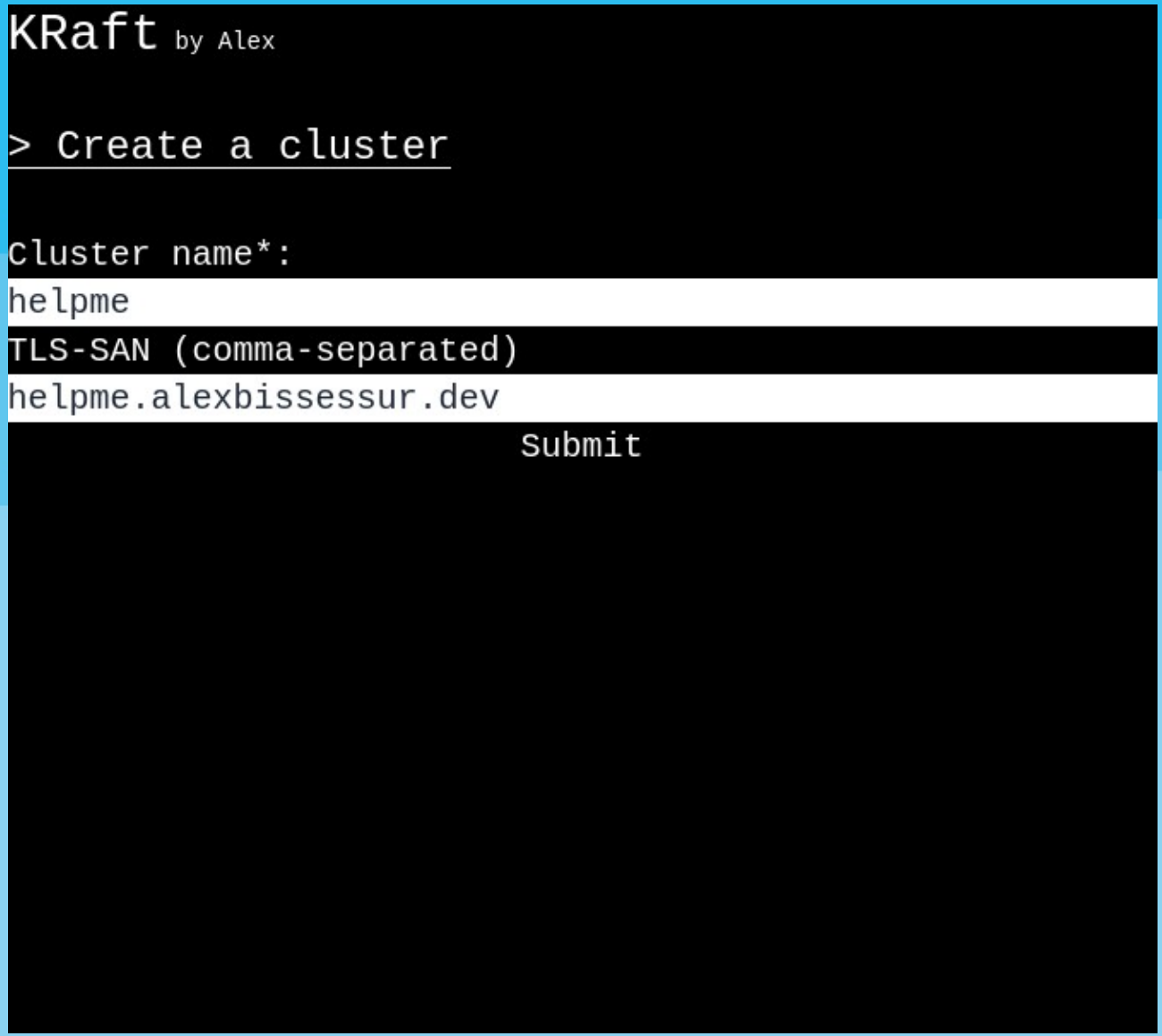
Confirm Password :

Submit

> Delete Account

WARNING: This deletes all your clusters, all your data, and your entire account. There is no turning back.

Delete


KRaft by Alex
> Create a cluster
cluster name*:
helpme
TLS-SAN (comma-separated)
helpme.alexbissessur.dev
Submit

KRaft by Alex

> View Clusters

Cluster	Endpoint	Actions
3-test	bigeneric-commissure	Expand
3-cnmu	sloppily-skijorings	Expand

Cluster Name

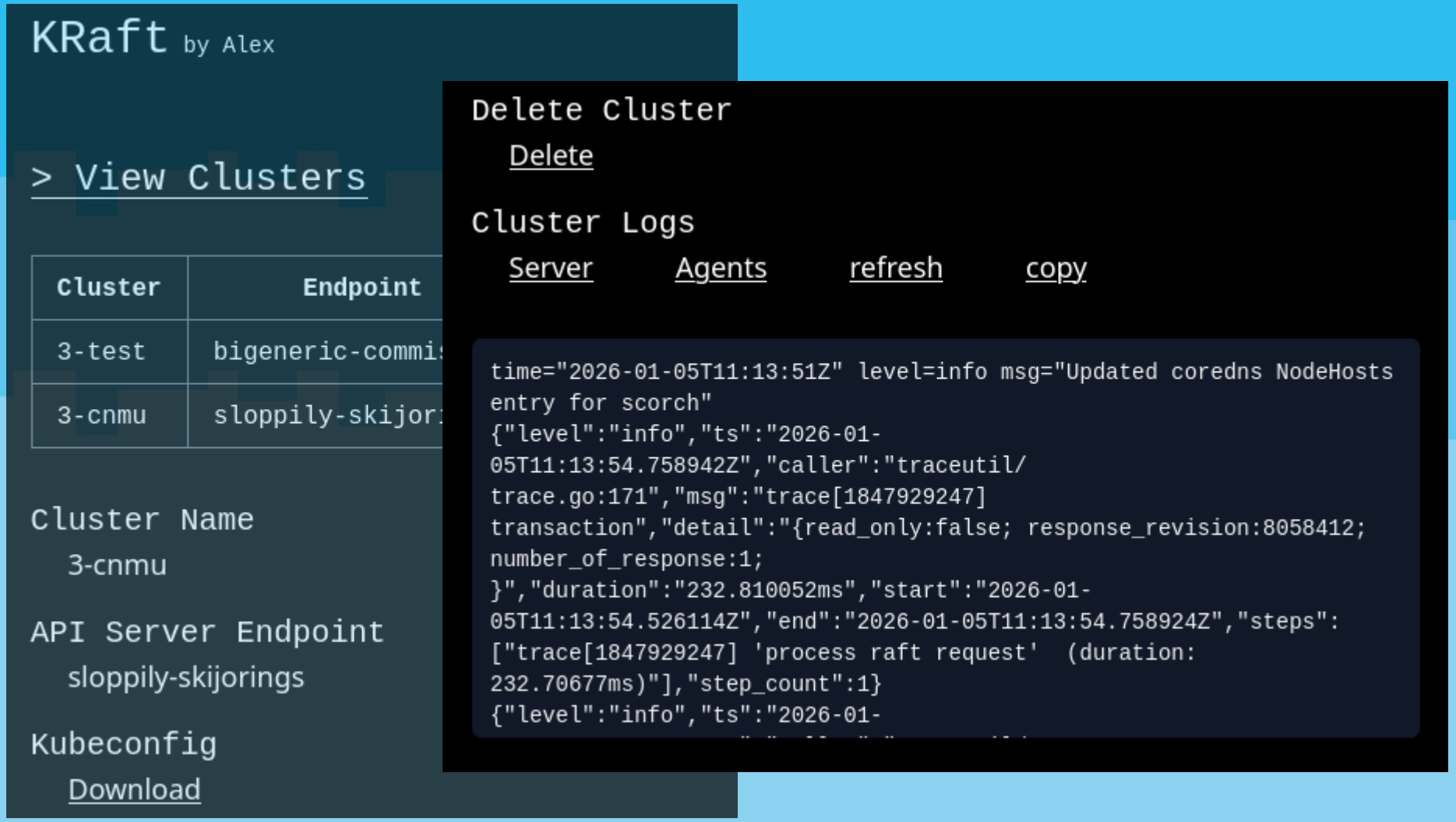
3-cnmu

API Server Endpoint

sloppily-skijorings

Kubeconfig

Download


KRaft by Alex
> View Clusters
Cluster
Endpoint
3-test
bigeneric-commi
3-cnmu
sloppily-skijor
Cluster Name
3-cnmu
API Server Endpoint
sloppily-skijorings
Kubeconfig
Download
Delete Cluster
Delete
Cluster Logs
Server
Agents
refresh
copy
time="2026-01-05T11:13:51Z" level=info msg="Updated coredns NodeHosts entry for scorch"
{"level":"info","ts":"2026-01-05T11:13:54.758942Z","caller":"traceutil/trace.go:171","msg":"trace[1847929247] transaction","detail":"{read_only:false; response_revision:8058412; number_of_response:1; }","duration":"232.810052ms","start":"2026-01-05T11:13:54.526114Z","end":"2026-01-05T11:13:54.758924Z","steps":["trace[1847929247] 'process raft request' (duration: 232.70677ms)"],"step_count":1}
{"level":"info","ts":"2026-01-

Keeping Workloads Safe

- Users interact with only their virtual API
- Clusters are
  - separated by namespace
  - isolated through network policies
  - secured with pod security admission level
- Each cluster has its own CoreDNS

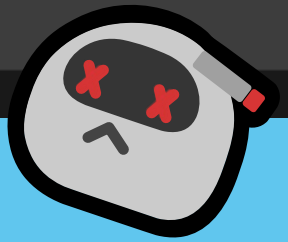
```
📦[bazzite@dump Downloads]$ export
KUBECONFIG=/home/bazzite/Downloads/4-help
📦[bazzite@dump Downloads]$ kubectl get
nodes
NAME        STATUS   ROLES    AGE
VERSION
northstar   Ready    agent    100m
v1.33.5-k3s1
ronin       Ready    agent    100m
v1.33.5-k3s1
scorch      Ready    agent    100m
v1.33.5-k3s1
```
22:56

It's aliiiive :o 22:57

WAIT IT WORKS!!!! 22:57

Haha, so surprised xD 22:57

Ah, yes.
HTTP 200 Success
Yet UI shows "An error occurred somewhere between here and the server."
Good start 21:12



Now 0:01

listen 0:01

it might not be covered by a TLS cert 0:01

Buuut 0:01

http://testing.kraftcloud.dev 0:01

I did get a thing to be hosted and it just freaking worked first try 0:01

I am impressed 0:01

(Also it only loads via curl, because you have hsts on that domain) 0:02

Work In Progress

- Ingresses are across the cluster – can cause conflicts
- Improvements needed on k3k-rs CRD structure
- K3k does not run in restricted mode
- Need to figure out backups
- Portability with Helm charts to install anywhere

Thank You!

to be replaced by a webpage for a list of socials