



KubeCon



CloudNativeCon

North America 2024





KubeCon



CloudNativeCon

North America 2024

A Mad Scientist's Guide to Automating CNI with Generative AI

Doug Smith, Red Hat



- Engineer at Red Hat, focusing on OpenShift Networking
- Background in, ironically, telephony and emerging technologies
- Member of the K8s Network Plumbing Working Group (lots of CNI stuff!)
- Lives in Vermont, USA
 - A.K.A “Vtah” for its powder snow (by east coast standards)
- Mad science credential: Has a trained cat, Juniper
 - Yes, I am that weird.

SHOW OF HANDS!

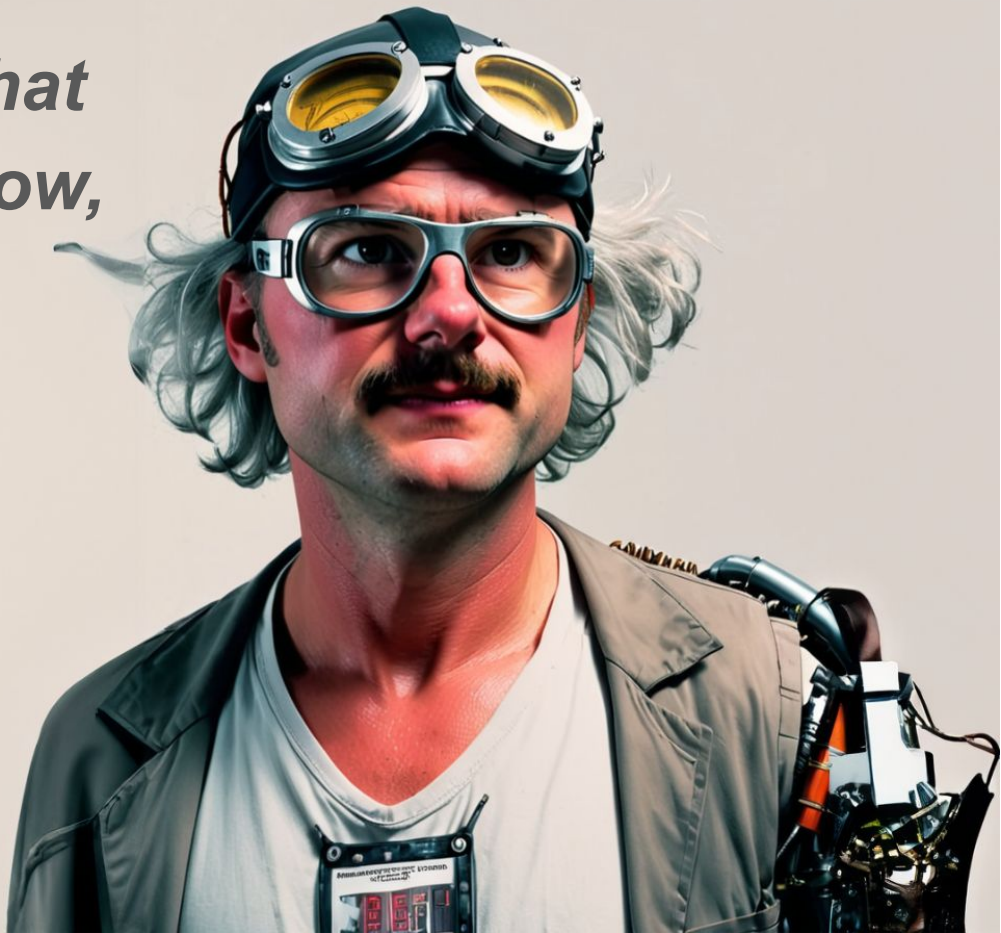
- Is this your first Kubecon?
- Who has used ChatGPT, Gemini, or Copilot?
- Who knows what an LLM is?
- Who has used `kubectl` before?
- Who has used KIND before?
 - Kubernetes-in-Docker
- Who knows what CNI is?
- Who has written a CNI configuration before?
 - (Spoiler: You don't need to know how to!)

- I'll keep it to 15 minutes of blabbing, then!
 - Origins
 - What we're doing today.
 - Quick overview of involved technology.
- Hands on workshop and tutorial!
 - All you need is a laptop that can run KIND (kubernetes-in-docker)
 - Conference wifi is always a gamble. Let it ride!
 - Feel free to watch, too.
- Follow up questions, discussion, experiments, etc!
 - Time permitting.

The origins!

“I wouldn’t recommend that you have an LLM, you know, automatically configure systems for you”

- A Data Scientist I was in a meeting with
(paraphrased)



I have users who have to write...



What are we doing today?



KubeCon



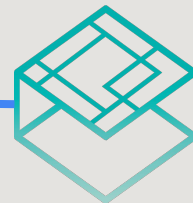
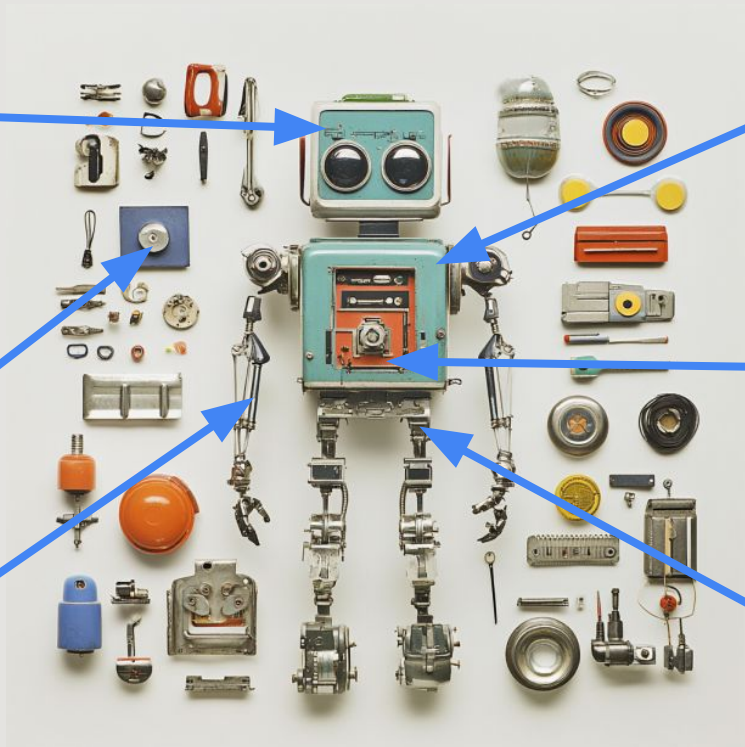
CloudNativeCon

North America 2024

We are going to Voltron together a science experiment!



ollama



CNI

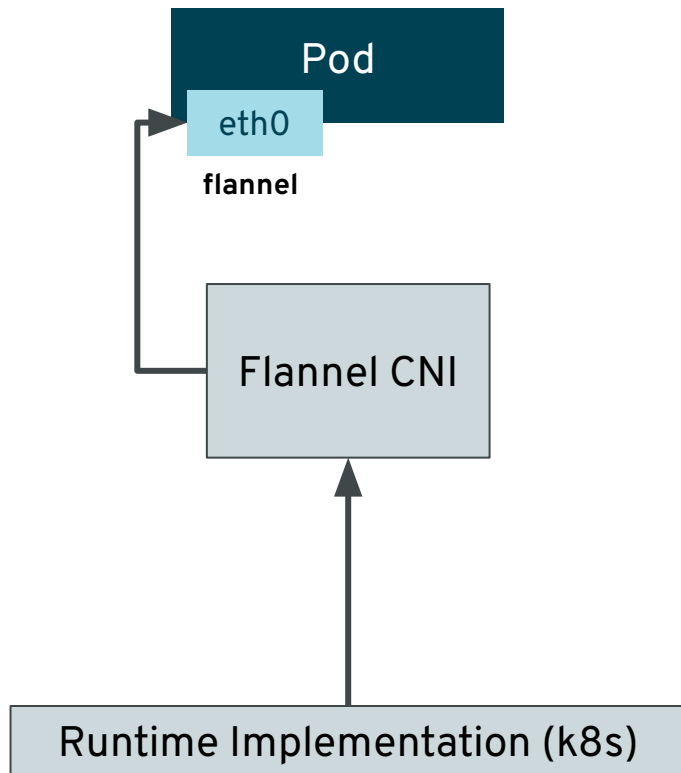


MULTUS

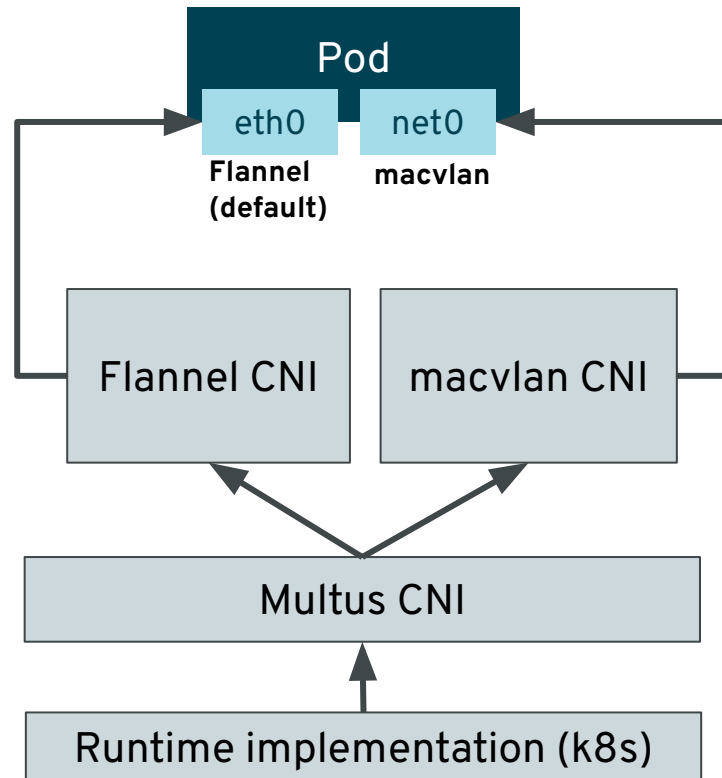
Generated with MidJourney, try "knolling" in your prompts.

We're going to use Multus CNI, to get 2 network interfaces

Pod without Multus



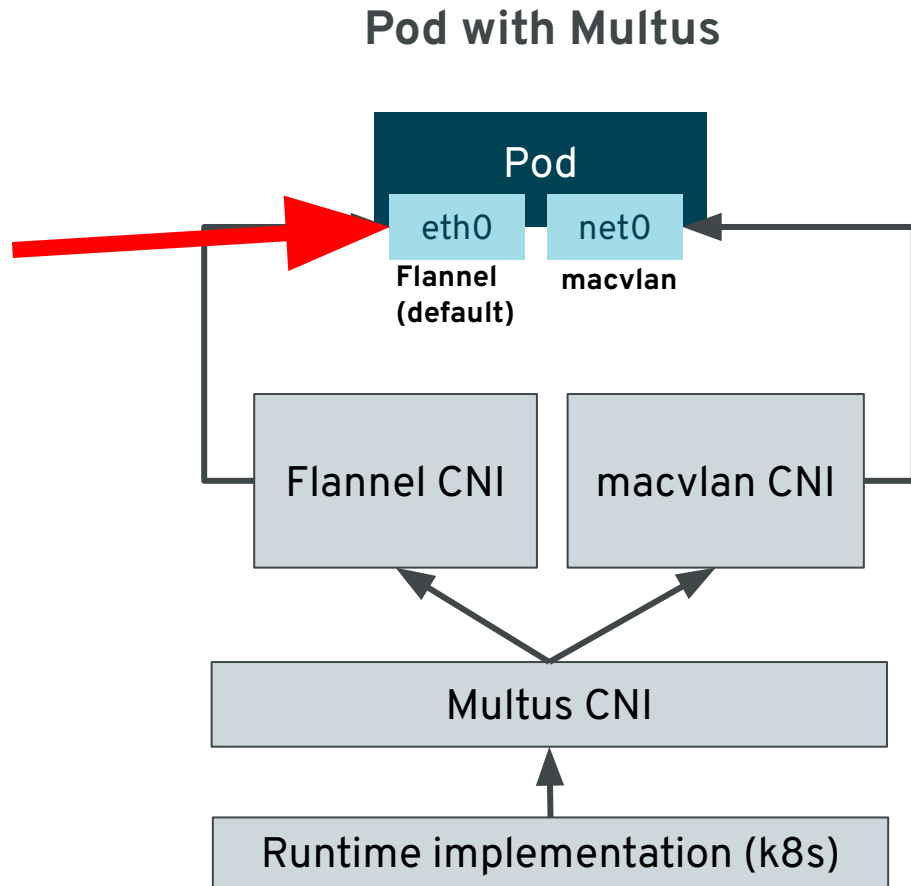
Pod with Multus



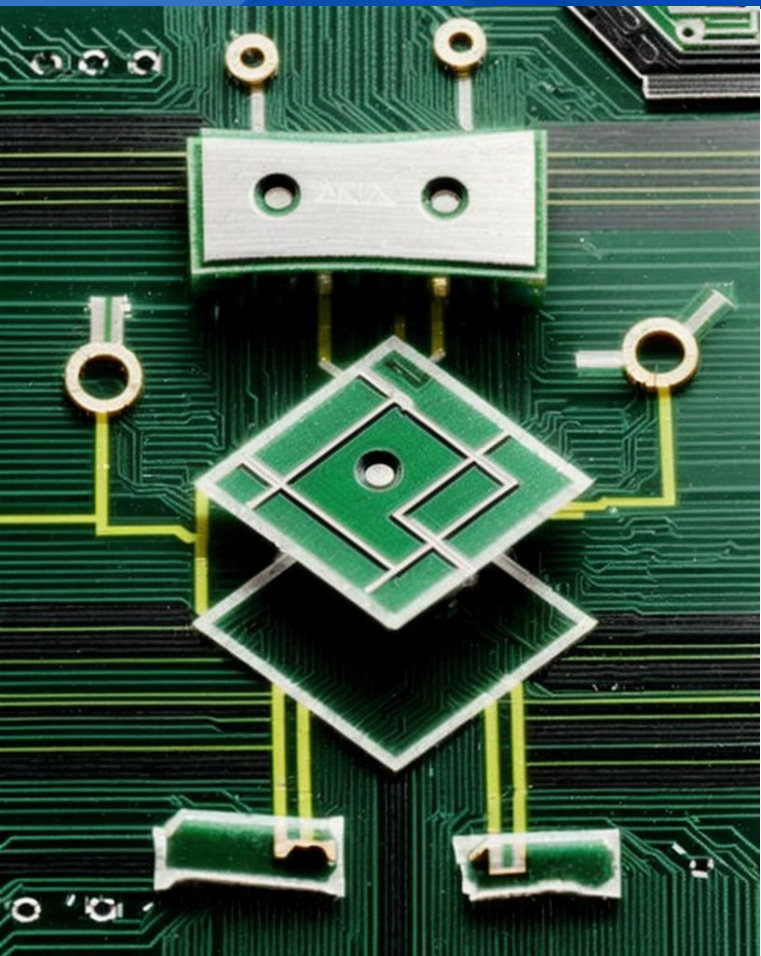
First: Do no harm (The Hippocratic Oath)

We will leave this untouched, so that pods function normally without a robot's hallucinogenic experience mucking with our "primary network"

This way our pods pass kubernetes health checks, can access the k8s API, all that good stuff.



Enter: RoboCNIConfig!



<https://github.com/dougbtv/robocniconfig>

A golang app to query an LLM and generate CNI configurations.

It uses a prompt engineering technique called “[context scaffolding](#)” and uses a kind of “pre prompt” that has a lot of information about CNI plugins that I created, with stuff like:

```
If no IP Addressing is provided in the hint, use IP addresses in the
10.20.0.0/16 range.
If a CNI configuration has a "master" field (as for bridge and CNI) set it by the
list of interfaces and routes provided or from the hint.
Do not mix up the different types, e.g. bridge, macvlan and ipvlan.
Do not use parameters that are not in the examples. Do not use optional fields
unless the hint implies their usage.
Set the value of the "name" field as DNS-1123 name based a short "slug" that you
create based on a summary of the hint (the name field is always required).
(...and more)
```

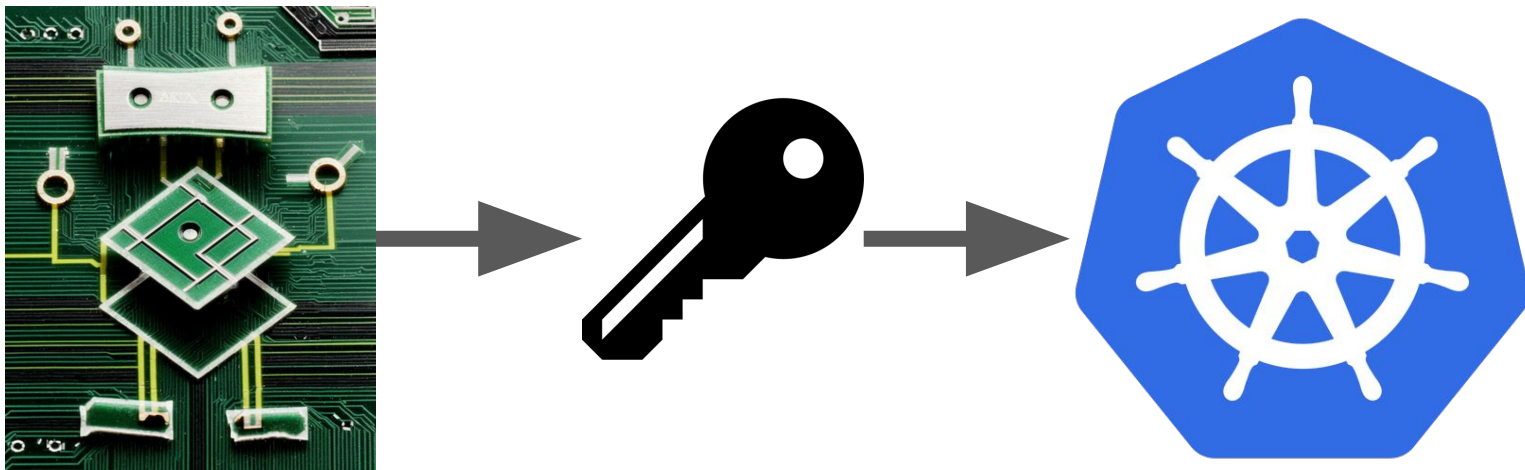
“Like rolling a million sided die on a million yard field.”

Our context scaffold reduces the size of the field.



DON'T TRY THIS IN PROD, KIDS!

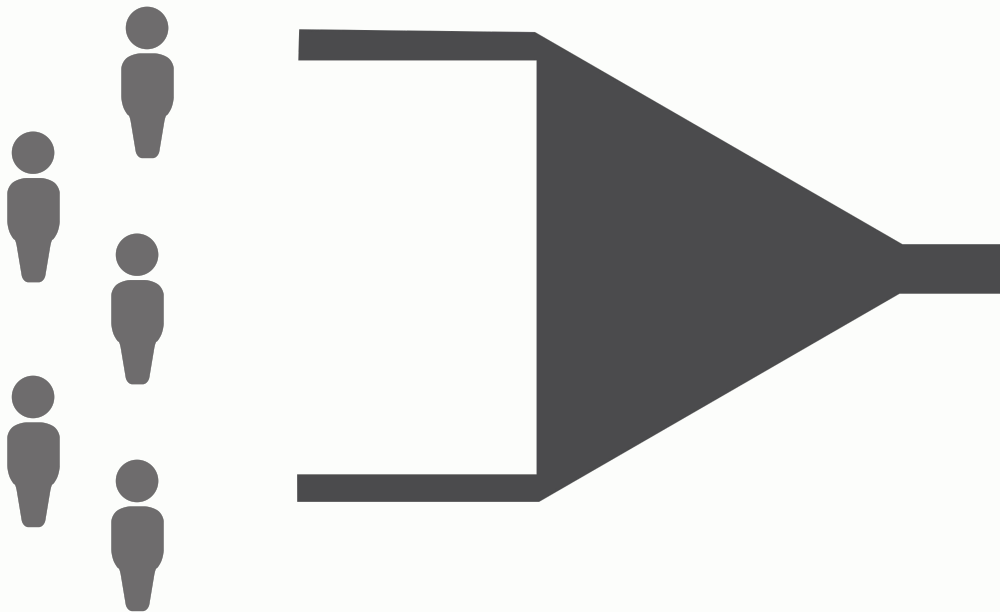
Robocni calls `kubect1` from a shell, and will use our creds to create, update and delete Kubernetes resources (pods and custom resources)



In other words, you just handed the keys to the ship to a robot.

GPU: The great filter in this tutorial

If you don't have a GPU, I will provide you a cloud instance of ollama to use!



Tutorial instruction steps!

<https://tinyurl.com/kubecon-cni>

<https://github.com/dougbtv/tutorial-mad-science-cni>



THE END!



KubeCon



CloudNativeCon

North America 2024

{INTENTIONALLY BLANK}