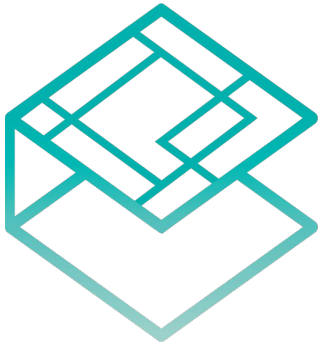
CNI Unleashed: how to deal with CNI plugin chains

February 5th, 2023, FOSDEM Network Dev Room

Daniel Mellado
dmellado@redhat.com
https://github.com/danielmellado

Miguel Duarte
mdbarroso@redhat.com
https://github.com/maiqueb



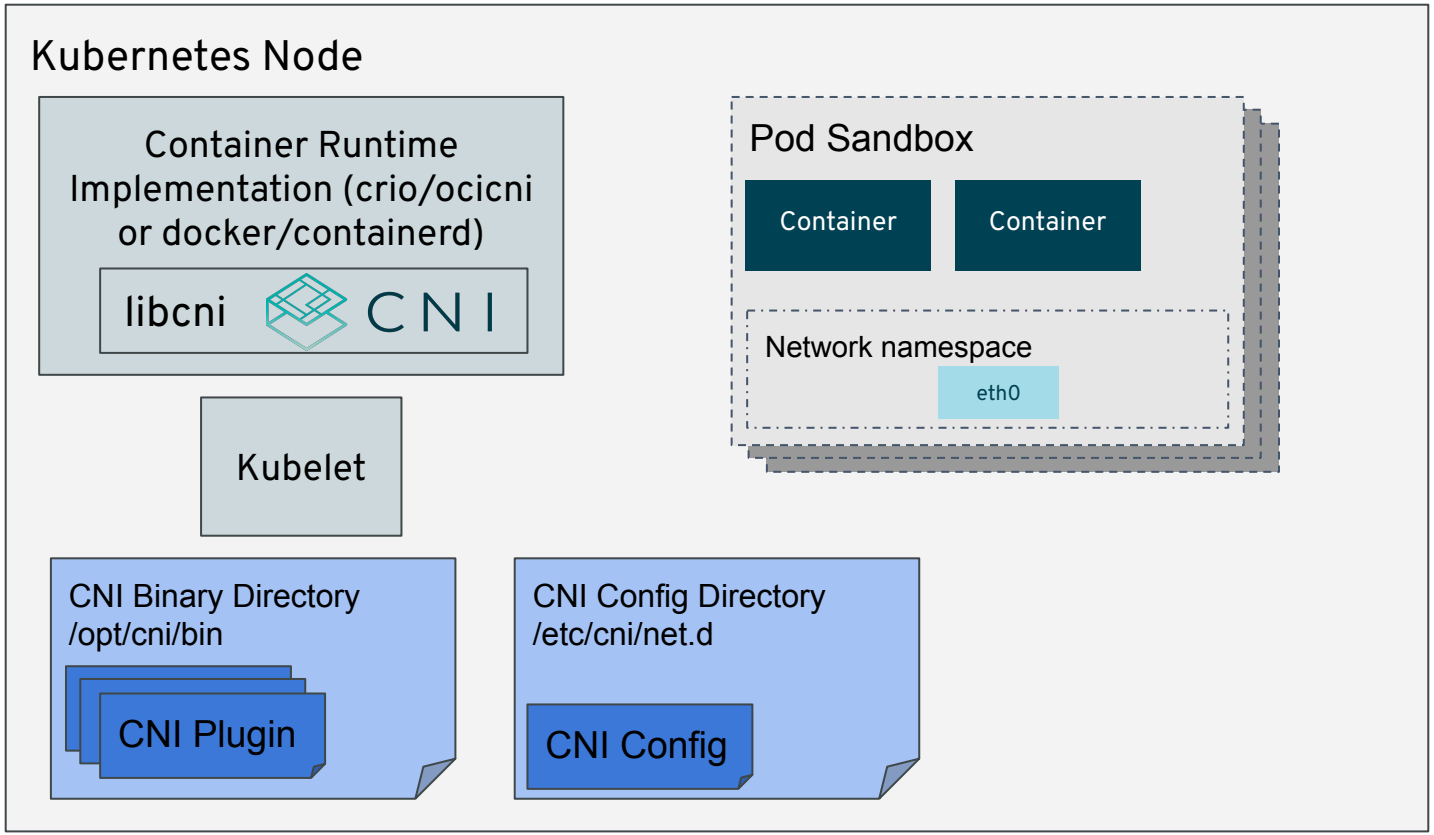
CNI

Agenda

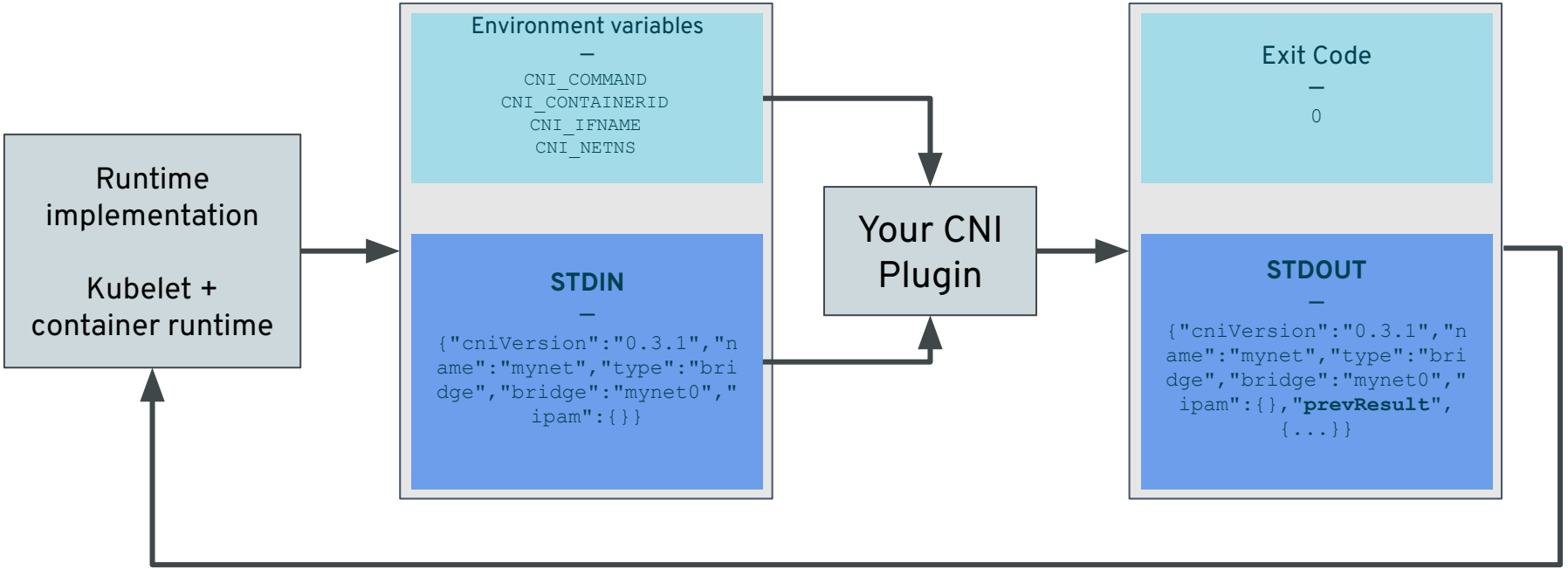
- CNI intro
- CNI plugin chains
- Plugin chain use cases
- Demo

CNI basics

CNI anatomy: from a Kubernetes perspective



CNI specification

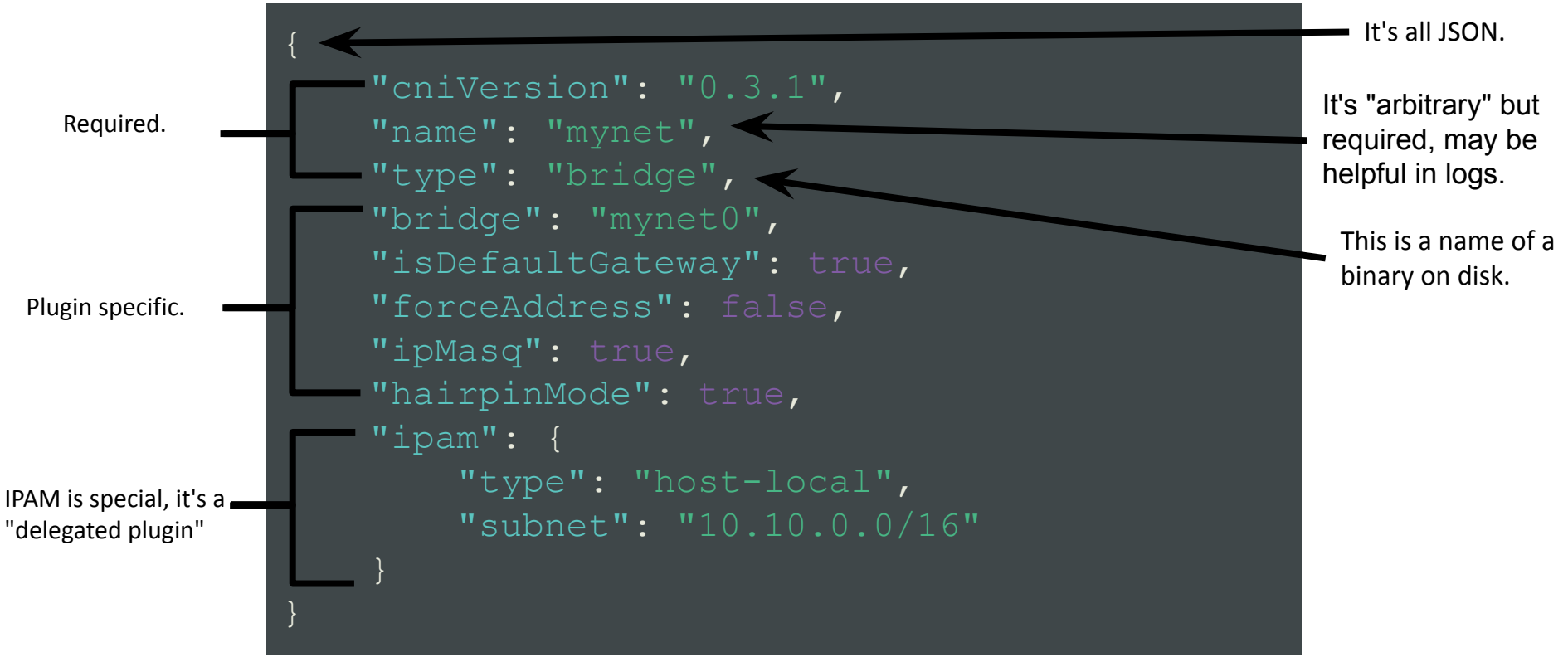


CNI operations

ADD	*Add container to network, or apply modifications*
DEL	*Remove container from network, or un-apply modifications* Do garbage collection!
CHECK	*Check container's networking is as expected* Generally called right after pod creation succeeds. Exit non-zero if check doesn't succeed.
VERSION	*probe plugin version support* Check the spec for the exact format.

https://github.com/containernetworking/cni/blob/main/SPEC.md#cni-operations

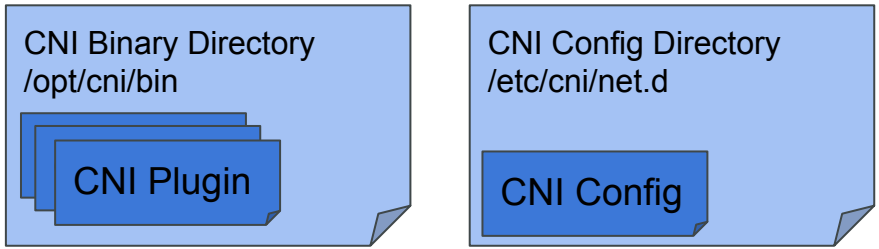
CNI configuration



How does CNI find the binaries / configs ?

Relevant parameters:

- `cni-conf-dir` => path to the CNI configuration
  - Defaults to `/etc/cni/net.d`
  - Smallest lexicographical order
- `cni-bin-dir` => path to CNI executables
  - Defaults to `/opt/cni/bin`



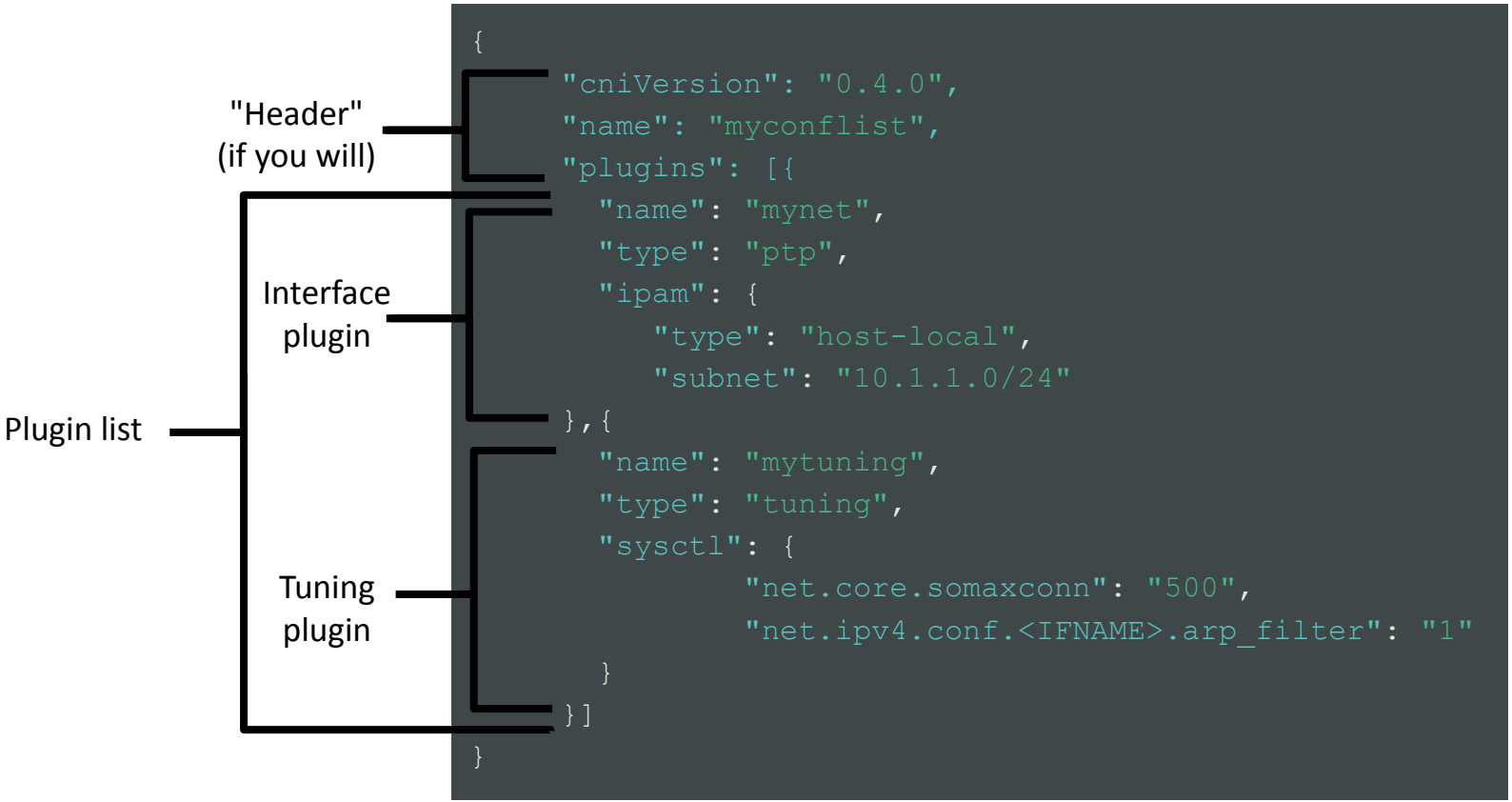
https://kubernetes.io/docs/concepts/extend-kubernetes/compute-storage-net/network-plugins/

CNI plugin chains

Chained plugins

- Adjust the configuration of an already-created interface
  - may need to create more interfaces to do so
- Available since CNI v0.3.0
- **Required** since CNI v1.0.0
- `.conflist` file extension when checking CNI configuration
  - `.conf` won't work …
- When a meta plugin is passed a `prevResult`
  - **MUST** handle it: either passing it through, or modifying it appropriately
- Delete considerations
  - The list of plugins is executed in reverse order (add: x->y->z ; delete: z->y->x)
  - The previous result provided is always the final result of the add operation.

CNI configuration: chained plugins



Use cases

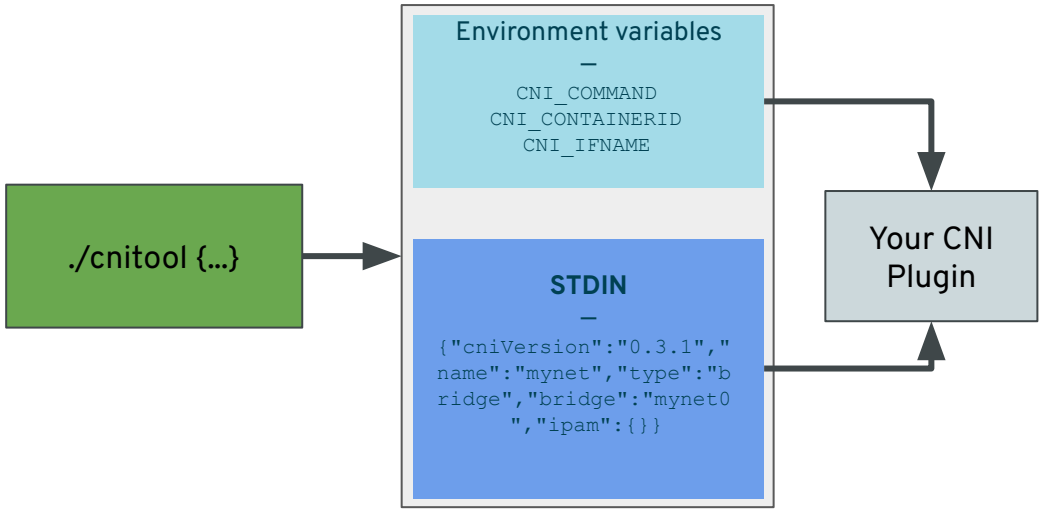
Use cases

- Tuning CNI
  - Sysctl allow-list / sysctl button pusher
- Bandwidth CNI
  - Throttle ingress/egress traffic
- Firewall
  - allow traffic to/from container IP address
- Port mapping
- …

Demo

CNI Tool: Your CNI swiss army knife

- Full tutorial / DIY workshop @ https://dougbtv.com/nfvpe/2021/05/14/using-cnitool/
- It allows you to execute your plugins without having to launch a pod, cnitool calls your binary with the ENV variables and CNI configs.



https://github.com/maiqueb/fosdem2023-cni-unchained#bandwidth

https://github.com/maiqueb/fosdem2023-cni-unchained#debug-cni

Conclusions

- Plugins only useful when used in addition to other plugins => meta plugins
- Meta-plugins **enable** plenty of use cases
  - Prevent IP spoofing / bandwidth throttle / port-forward / configure sysctls /…
- Meta-plugins **must** handle the result of previous plugins in the chains
- Plugin chains are the **only allowed** CNI configuration from CNI v1.0.0
- Know your `prevResult`

Thank you! Questions ?...

CNI config example - calico

```
{
    "name": "any_name",
    "cniVersion": "0.1.0",
    "type": "calico",
    "kubernetes": {
        "kubeconfig": "/path/to/kubeconfig"
    },
    "ipam": {
        "type": "calico-ipam"
    }
}
```

https://docs.projectcalico.org/reference/cni-plugin/configuration

CNI config example - calico

```
{
    "name": "any_name",
    "cniVersion": "0.1.0",
    "type": "calico",
    "kubernetes": {
        "kubeconfig": "/path/to/kubeconfig"
    },
    "ipam": {
        "type": "calico-ipam"
    }
}
```

CNI bin dir

```
$ podman exec node01 "ls -lah /opt/cni/bin"
...
-rwxr-xr-x. 1 root root  35M Nov 15 09:12 calico
-rwxr-xr-x. 1 root root  35M Nov 15 09:12 calico-ipam
```

https://docs.projectcalico.org/reference/cni-plugin/configuration

“Full” bandwidth

```
{
  "cniVersion": "0.4.0",
  "name": "full-steam-ahead",
  "plugins": [
    {
        "type": "bridge",
        "bridge": "mynet0",
        "isDefaultGateway": true,
        "capabilities": { "ips": true },
        "ipam": {
           "type": "static"
        }
    }
  ]
}
```

Throttled bandwidth

```
{
  "cniVersion": "0.4.0",
  "name": "limited-bandwidth",
  "plugins": [
    {
        "type": "bridge",
        "bridge": "mynet0",
        "isDefaultGateway": true,
        "capabilities": { "ips": true },
        "ipam": {
           "type": "static"
        }
    }, {
        "type": "bandwidth",
        "ingressRate": 500000,
        "ingressBurst": 50000
    }
  ]
}
```