Switch CNI to Cilium - Our journey

Adfinis

Plan. Innovatively | **Build.** Sustainably | **Run.** Resiliently

Potential. Unlocked

Jan Müller

Senior Systems Engineer, into 🐧 , 🥋 and 🥊.

Adfinis

- Zurich, Switzerland
- jan.mueller@adfinis.com
- https://www.linkedin.com/in/jamue/
- github.com/janaurka

Norbert Gruszka

DevOps Engineer

Adfinis

Zurich, Switzerland

norbert.gruszka@adfinis.com

linkedin.com/in/norbert-gruszka

github.com/norbertgruszka

Container Networking Interface (CNI)

CNI Introduction

#CNI

#kubernetes

#networking

#cilium

What is a CNI?

- Provides networking capabilities to your container (Pod)
- Vendor-neutral specification - describes a (small) common featureset
- Also used by Mesos, CloudFoundry, podman, etc.

What does a CNI (next to providing network access)?

- Traffic encryption
- Integration into an IPAM
- Security features
- Monitoring capabilities
- Service Mesh capabilities

Popular implementations

- Cilium
- Calico
- Flannel
- Cloud Provider-specific CNIs (AWS, Azure, etc.)

adfin.is/cni-intro

Want to know more about CNIs? Check this introduction from KubeCon NA 2019

Why we talk about CNIs and migrations

Migration project for Dectris Ltd.

Our customer - Dectris Ltd. - approached us to migrate their current CNI to Cilium to leverage it's encryption and observability capabilities.

Researching CNI migrations turned out to be a topic not many people talk (openly) about. So we do it

DECTRIS
detecting the future

How to migrate a CNI

Let’s lay foundations for this project

What we had:

- We were working with cluster managed by **Kubespray** version 2.20.0
- CNI in use: **Flannel**
- Permission to have downtime

What we wanted to have:

- CNI in use: **Cilium**
- Node-to-node **encryption enabled**

Node-to-node encryption

For Cilium you can choose between two solutions: **WireGuard** and **IPsec**

We decided to go with **IPsec** because:

- WireGuard **incompatibility with L7 policy** enforcement and visibility.
- Host-level encryption. **Only traffic between two Cilium-managed endpoints (i.e. pod-to-pod traffic) is encrypted.** Traffic between two nodes and traffic between a Cilium-managed pod and a remote node currently won't be encrypted.

Obviously IPsec also has limitations but they did not concern us in this project.

Source: https://docs.cilium.io/en/stable/security/network/encryption-wireguard/#limitations

CNI installation - Kubespray

You can install and configure Cilium CNI with Kubespray with parameter:

```
kube-network-plugin: cilium
```

However, we faced some issues:

- We had to **upgrade Kubespray to version 2.21.0** - bug in IPsec configmap.
- **Hubble was not working** out-of-the box.

CNI installation - use Cilium Helm chart

```
helm repo add cilium https://helm.cilium.io/
helm install cilium cilium/cilium \
    --version 1.12.6 \
    --namespace kube-system \
    --values cilium.yml
```

Uninstall Flannel - kubernetes resources

```
kubectl -n kube-system delete ds kube-flannel-*
kubectl -n kube-system delete configmap kube-flannel-cfg
kubectl -n kube-system delete serviceaccount flannel
kubectl delete clusterrole flannel
kubectl delete clusterrolebinding flannel
```

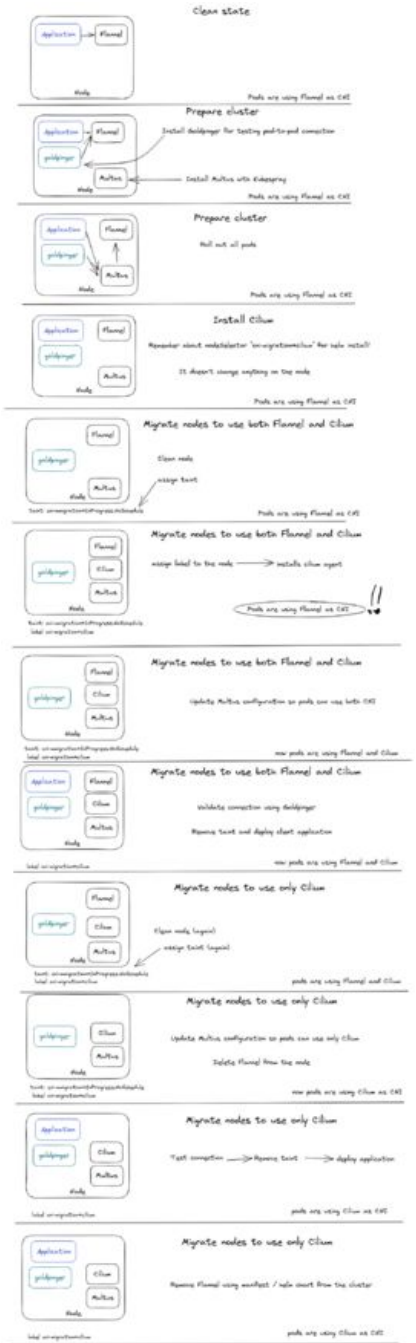
Uninstall Flannel - clean up nodes and reboot them

```
- ansible.builtin.file:
    path: /etc/cni/net.d/10-flannel.conflist.cilium_bak
    state: absent


- ansible.builtin.file:
    path: /run/flannel/
    state: absent


- ansible.builtin.shell: ifconfig flannel.1 down && ip link delete flannel.1


- name: Reboot Node
  ansible.builtin.reboot:
```

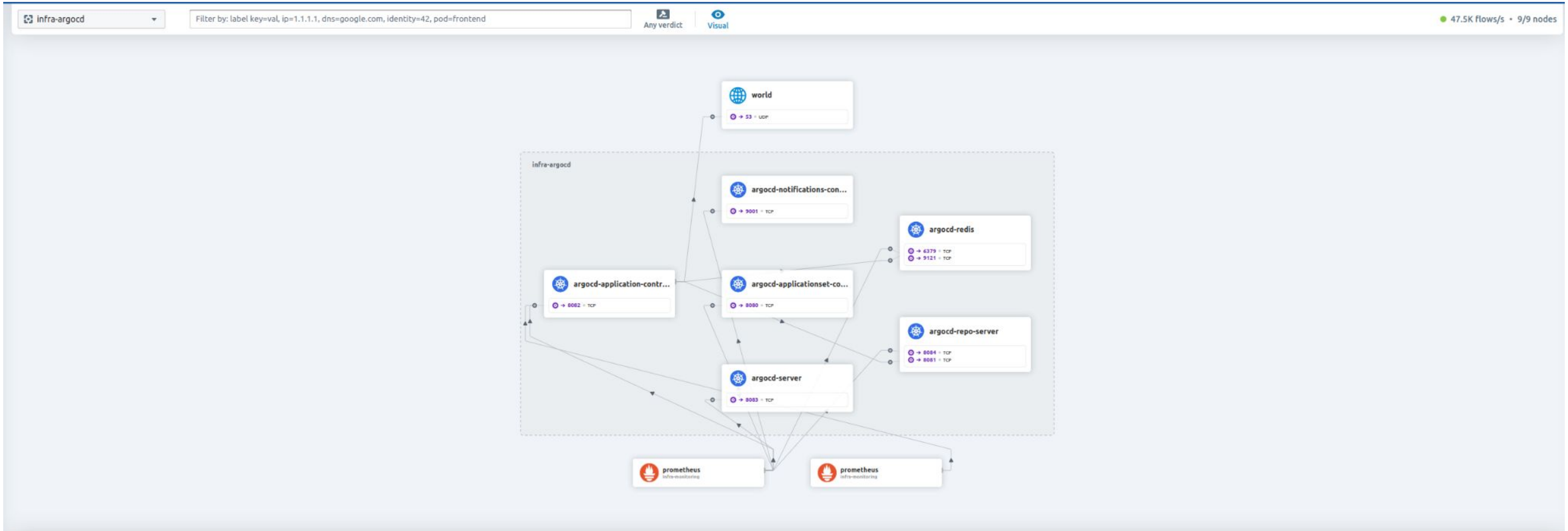
How about migration but without a downtime?

Previous option assumed that you have the luxury of having permission to restart a whole cluster. But what if you cannot?

- It is obviously possible, but it is **more complex procedure and it requires more time to complete.**
- You will need to maintain connection between pods deployed with 2 different CNIs, which is where **Multus** is handy.
- Rolling out changes node by node will require you to play around with labels and taints when installing Cilium or removing Flannel binaries.
- You will **roll out your deployments multiple times.**
- Depending on your needs, you will need to put in **additional effort to uninstall extra dependencies,** like Multus, after you are done.


Clean state
application
Flannel
Node
Pods are using Flannel as CNI
Prepare cluster
Install daidegger for testing pod-to-pod connection
Install Multus with Kubespray
Pods are using Flannel as CNI
Prepare cluster
Roll out all pods
Pods are using Flannel as CNI
Install Cilium
Remember about nodeSelector 'cni-migration=cilium' for safe install!
It doesn't change anything on the node
Pods are using Flannel as CNI
Migrate nodes to use both Flannel and Cilium
Clean node
assign taint
Pods are using Flannel as CNI
Migrate nodes to use both Flannel and Cilium
assign label to the node
install cilium agent
Pods are using Flannel as CNI
Migrate nodes to use both Flannel and Cilium
Update Multus configuration so pods can use both CNI
new pods are using Flannel and Cilium
Migrate nodes to use both Flannel and Cilium
Validate connection using daidegger
Remove taint and deploy client application
new pods are using Flannel and Cilium
Migrate nodes to use only Cilium
Clean node (again)
assign taint (again)
pods are using Flannel and Cilium
Migrate nodes to use only Cilium
Update Multus configuration so pods can use only Cilium
Delete Flannel from the node
new pods are using Cilium as CNI
Migrate nodes to use only Cilium
Test connection
Remove taint
deploy application
pods are using Cilium as CNI
Migrate nodes to use only Cilium
Remove Flannel using manifest / helm chart from the cluster
pods are using Cilium as CNI
THE END

Result



Columns

Source Service	Destination Service	Destination Port	L7 info	Verdict	Timestamp
argocd-application-controller *infra-argocd*	argocd-redis *infra-argocd*	6379	—	forwarded	less than 5 seconds
prometheus *infra-monitoring*	argocd-server *infra-argocd*	8083	—	forwarded	less than 5 seconds
prometheus *infra-monitoring*	argocd-repo-server *infra-argocd*	8084	—	forwarded	less than 5 seconds
prometheus *infra-monitoring*	argocd-application-controller *infra-argocd*	8082	—	forwarded	less than 5 seconds
argocd-application-controller *infra-argocd*	argocd-redis *infra-argocd*	6379	—	forwarded	less than 5 seconds
prometheus *infra-monitoring*	argocd-redis *infra-argocd*	9121	—	forwarded	less than 5 seconds
prometheus *infra-monitoring*	argocd-applicationset-controller *infra-argocd*	8080	—	forwarded	less than 10 seconds
argocd-application-controller *infra-argocd*	argocd-repo-server *infra-argocd*	8081	—	forwarded	less than 10 seconds
argocd-application-controller *infra-argocd*	argocd-repo-server *infra-argocd*	8081	—	forwarded	less than 10 seconds
argocd-application-controller *infra-argocd*	argocd-repo-server *infra-argocd*	8081	—	forwarded	less than 10 seconds
argocd-application-controller *infra-argocd*	argocd-repo-server *infra-argocd*	8081	—	forwarded	less than 10 seconds
argocd-application-controller *infra-argocd*	argocd-repo-server *infra-argocd*	8081	—	forwarded	less than 10 seconds
argocd-application-controller *infra-argocd*	argocd-repo-server *infra-argocd*	8081	—	forwarded	less than 10 seconds
argocd-application-controller *infra-argocd*	world *169.254.25.10*	53	—	forwarded	less than 10 seconds
argocd-application-controller *infra-argocd*	world *169.254.25.10*	53	—	forwarded	less than 10 seconds
argocd-application-controller *infra-argocd*	argocd-redis *infra-argocd*	6379	—	forwarded	less than 10 seconds
prometheus *infra-monitoring*	argocd-notifications-controller *infra-argocd*	9001	—	forwarded	less than 10 seconds
prometheus *infra-monitoring*	argocd-application-controller *infra-argocd*	8082	—	forwarded	less than 20 seconds
argocd-application-controller *infra-argocd*	argocd-repo-server *infra-argocd*	8081	—	forwarded	less than 20 seconds

Adfinis Blog - How to migrate CNIs

#blog

#oss_community

#summary

#ad

We created a blogpost summarizing the content of this talk

adfin.is/cni-switch

Stay in Touch

adfin.is/gQ

We are hiring!

/adfinis

/adfinis

adfinis.com

info@adfinis.com

/adfinis