CNI or Service Mesh?
Comparing Security Policies Across Providers

Christine Kim - Isovalent
@xtineskim

Rob Salmond - SuperOrbital
mastodon.social/@rsalmond

What we'll cover . . .

- What's a CNI? What's a Service Mesh?
- The What and How of Policy Enforcement
- Security Gotchas
- Mitigation and How the Field is Evolving
- What You Can Do

cilium

Istio

cilium

Istio

Top Ten CNCF Projects by:

cilium

Istio

Top Ten CNCF Projects by: commits

cilium

Istio

Top Ten CNCF Projects by: contributors

cilium

Istio

Top Ten CNCF Projects by: comments

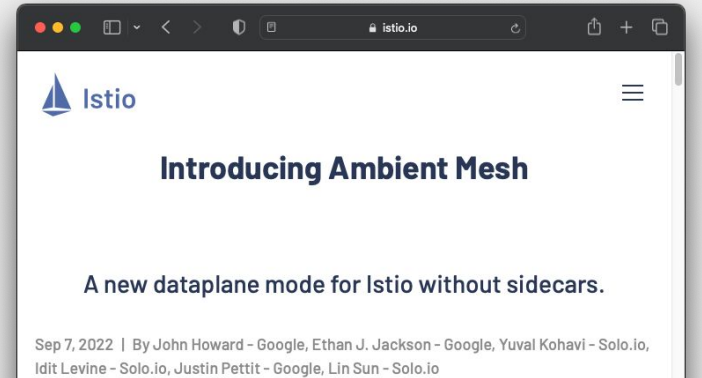
cilium

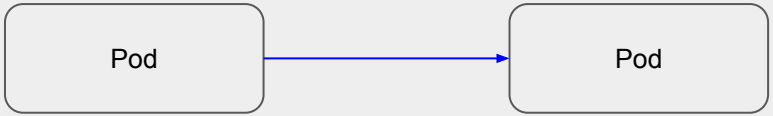
Istio

Top Ten CNCF Projects by: issues

cilium

Istio

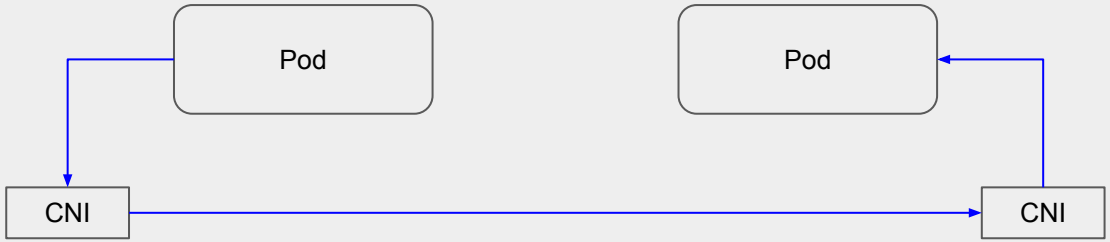




“pods can communicate with all other pods on any other node without NAT”

https://kubernetes.io/docs/concepts/services-networking/

data


Pod
Pod
CNI
CNI
data

What is CNI?

Container
Network
Interface

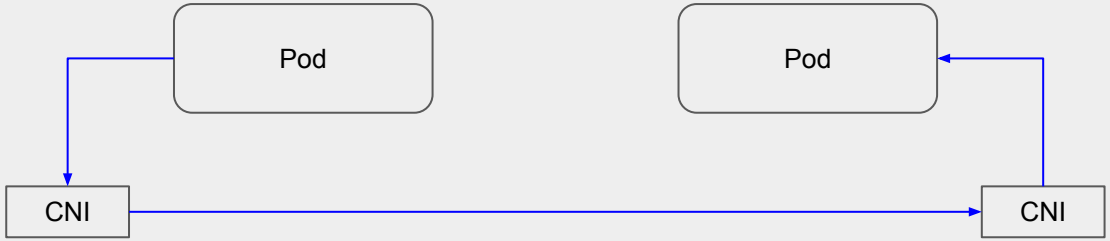
What is CNI?

Container
Network
Interface

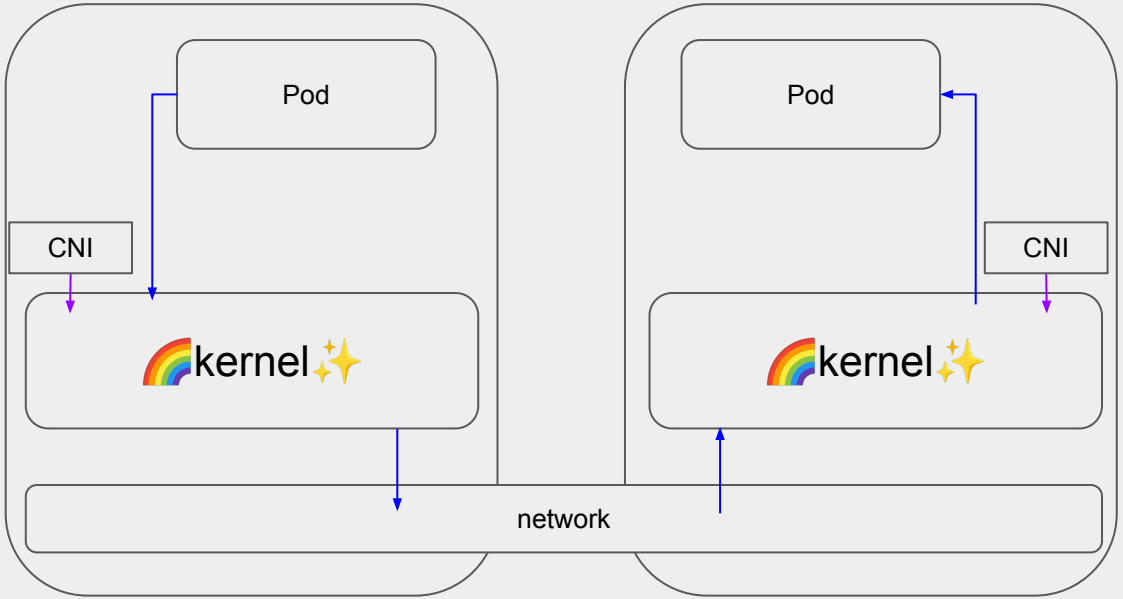
“A way to ask for changes to be made to a container’s network config.”

What kind of changes?

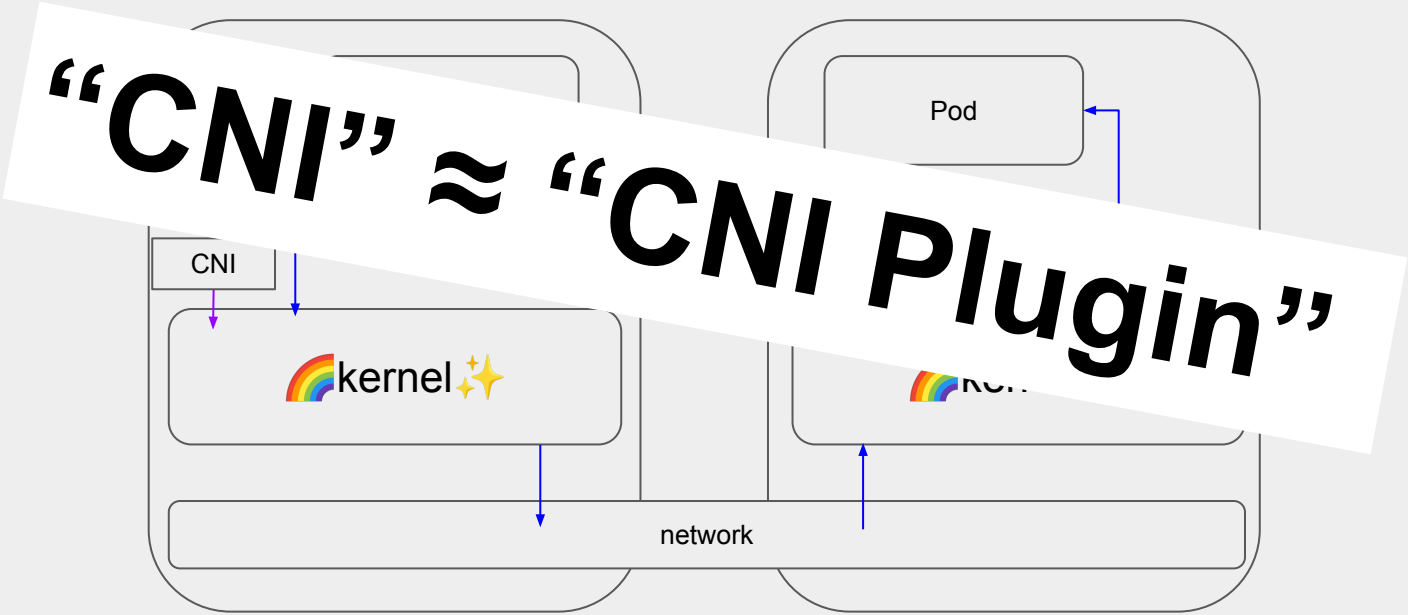
What kind of changes?



What kind of changes?



What kind of changes?



What is a CNI plugin?

CNI

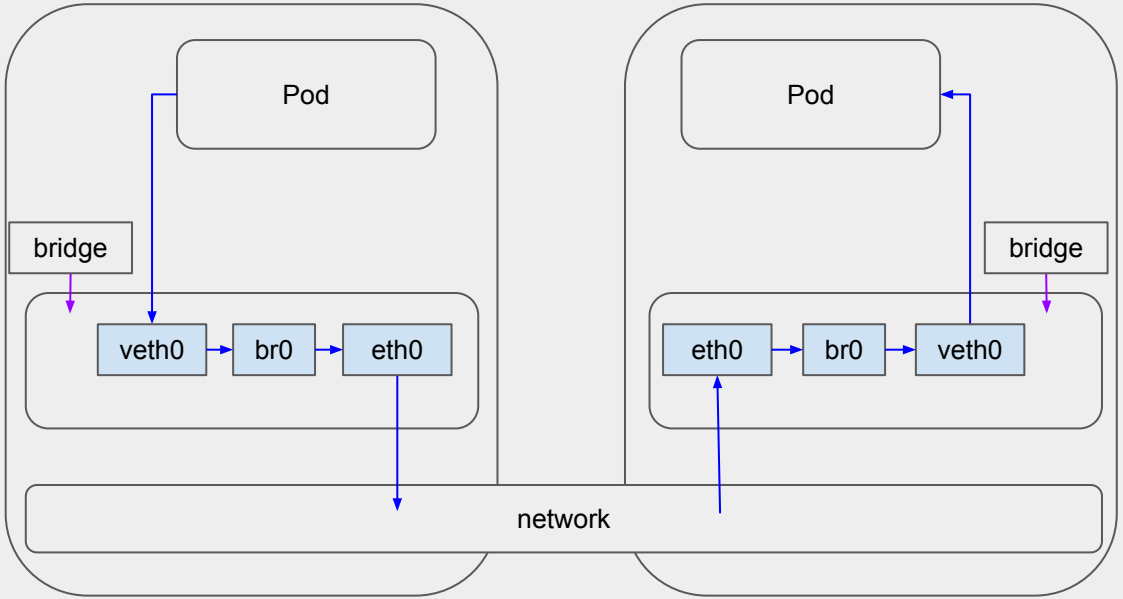
⇐ implements

What is a CNI plugin?

CNI ⇦ implements

“A thing that can make container networking changes.

bridge plugin

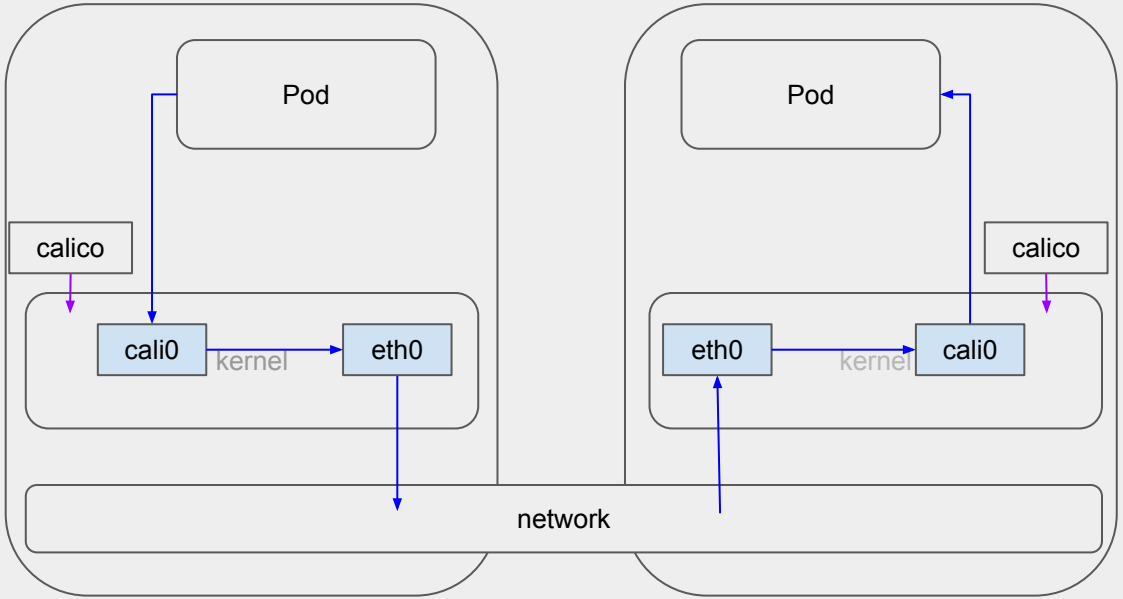


https://github.com/containernetworking/plugins

control

data

Calico plugin

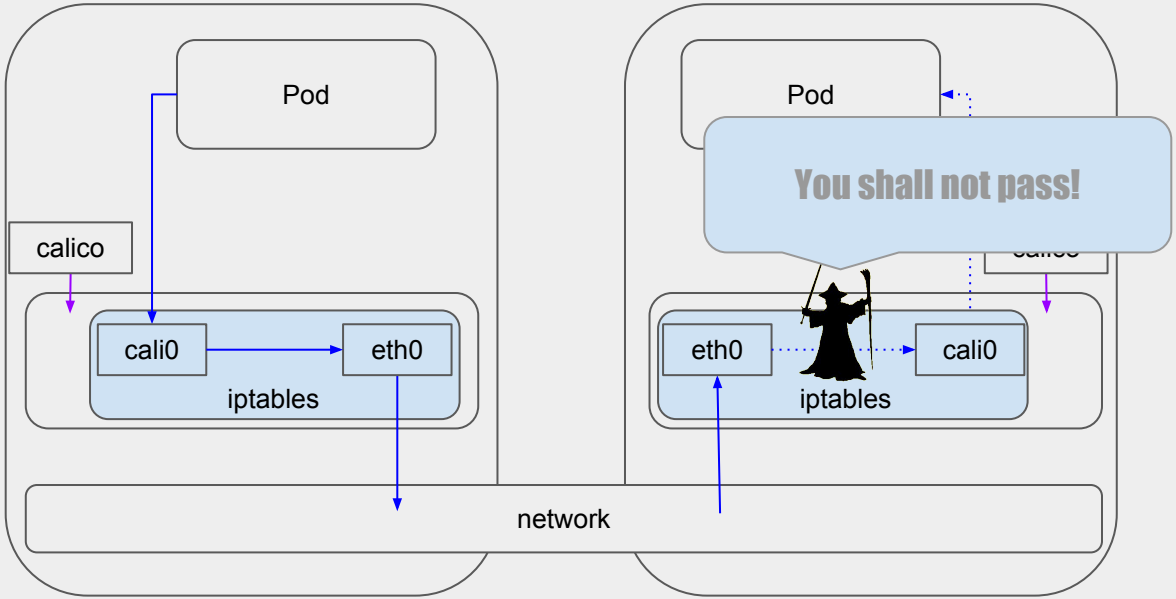


https://github.com/projectcalico/calico

control

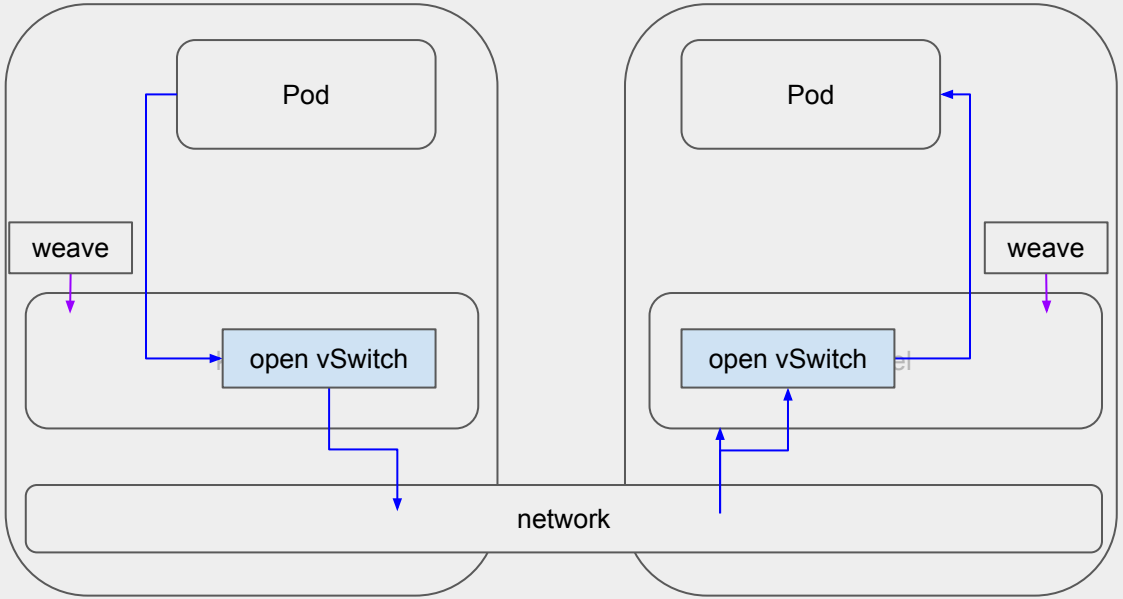
data

Calico plugin



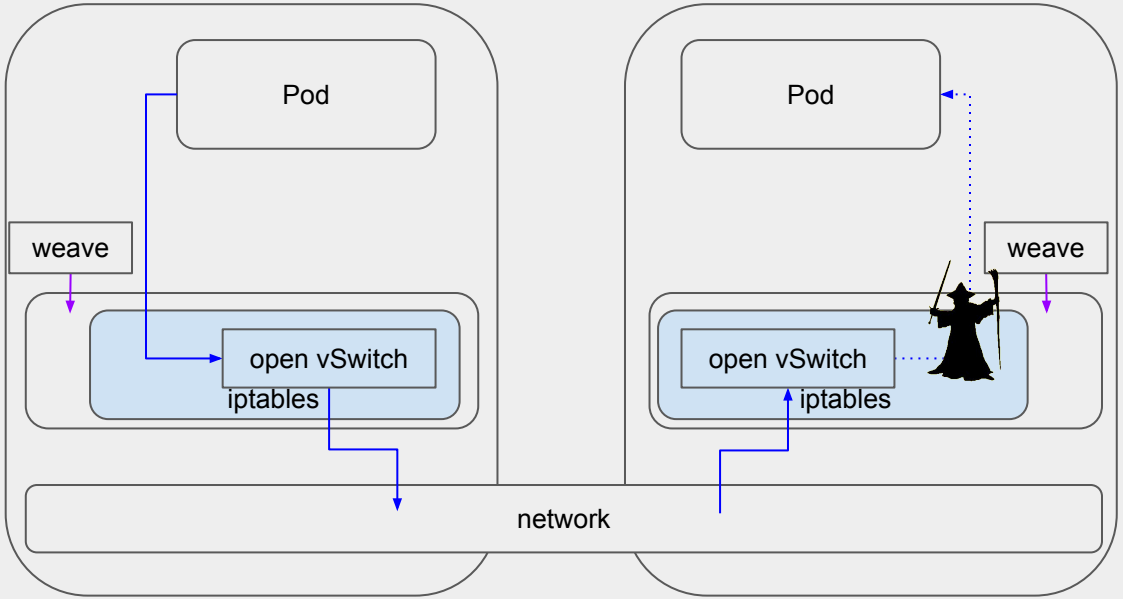
https://github.com/projectcalico/calico

weave plugin



https://github.com/weaveworks/weave

weave plugin

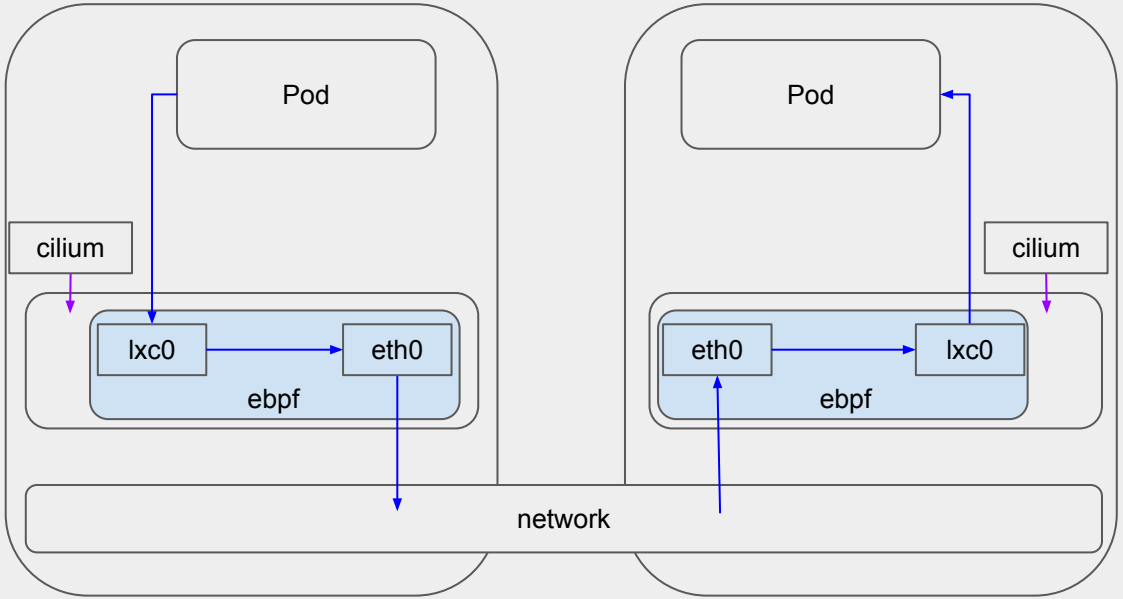


https://github.com/weaveworks/weave

control

data

cilium plugin

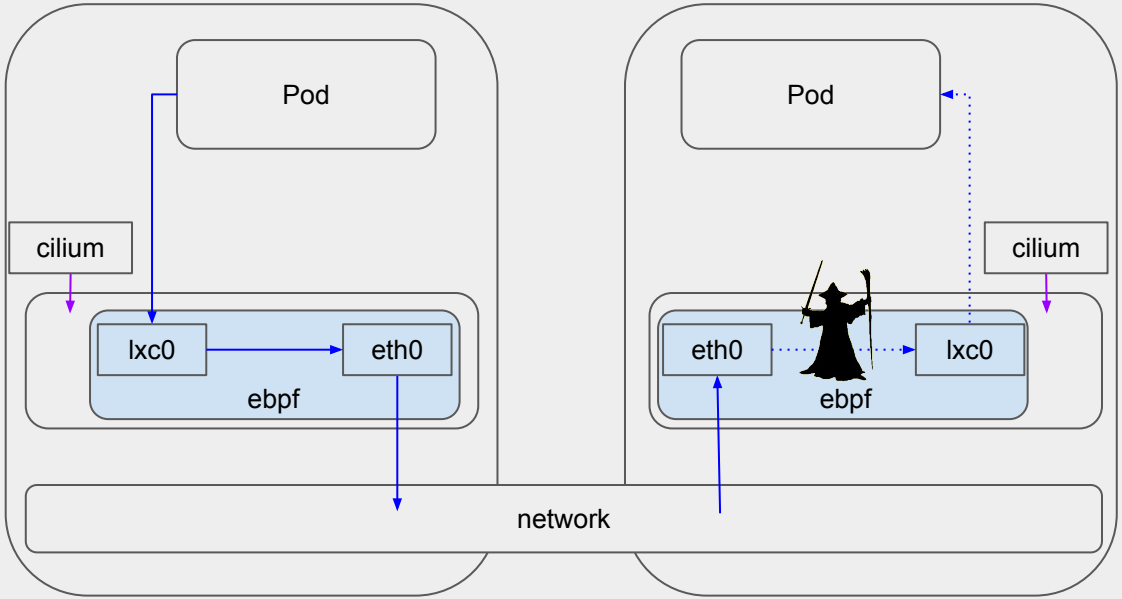


https://github.com/cilium/cilium

control

data

cilium plugin



https://github.com/cilium/cilium

control

data

Most of the popular CNI Plugins:

Most of the popular CNI Plugins:

- Configure pod to pod networking

Most of the popular CNI Plugins:

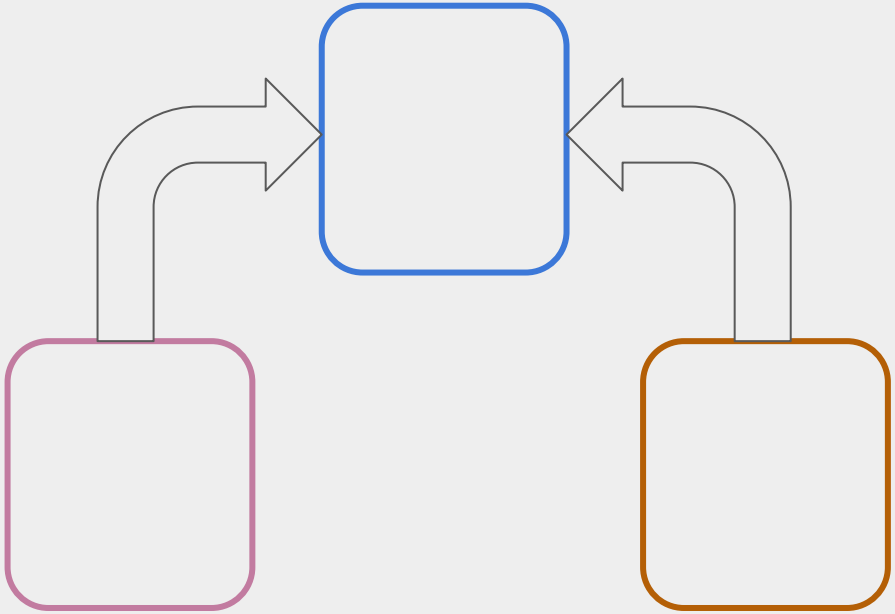
- Configure pod to pod networking
- Support NetworkPolicy enforcement

Most of the popular CNI Plugins:

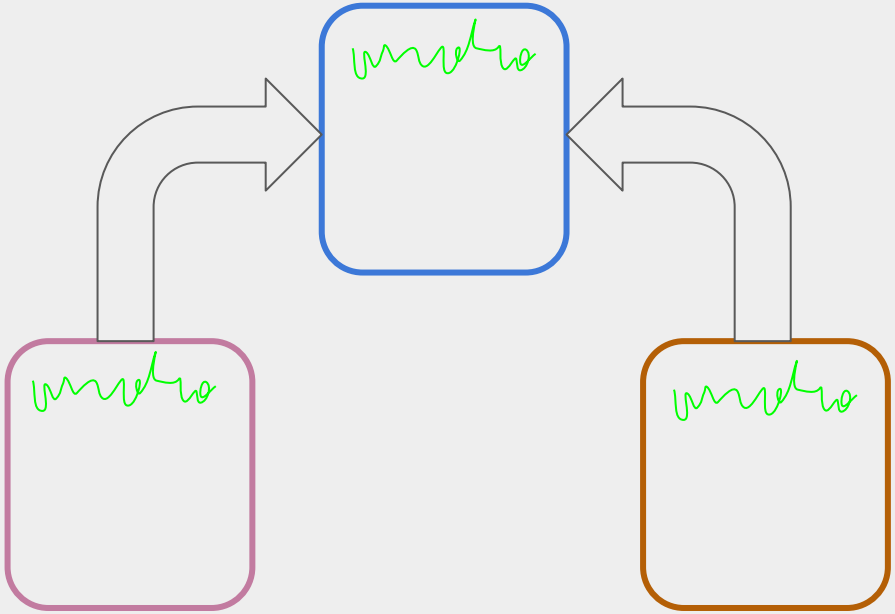
- Configure pod to pod networking
- Support NetworkPolicy enforcement
- ≈ Cloud Native software defined network

What is a Service Mesh?

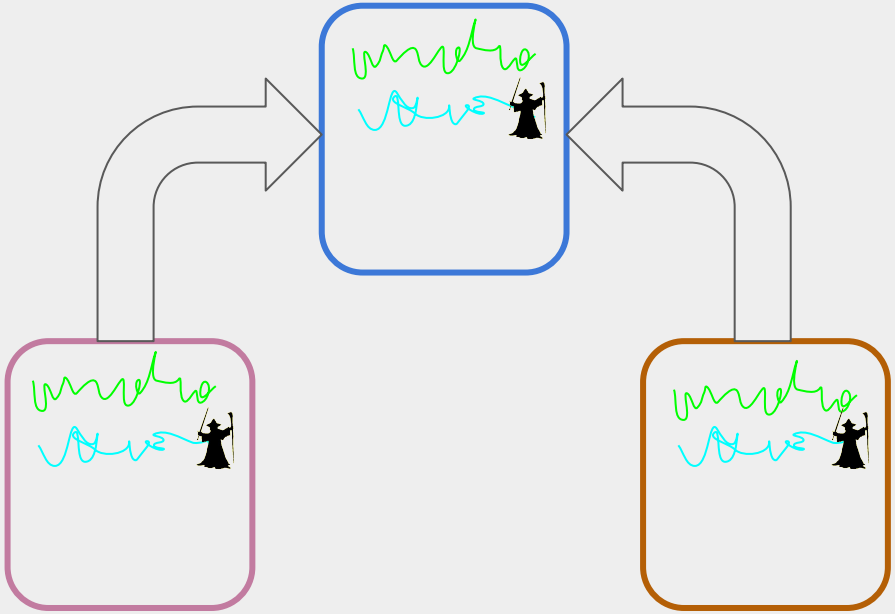
What is a Service Mesh?



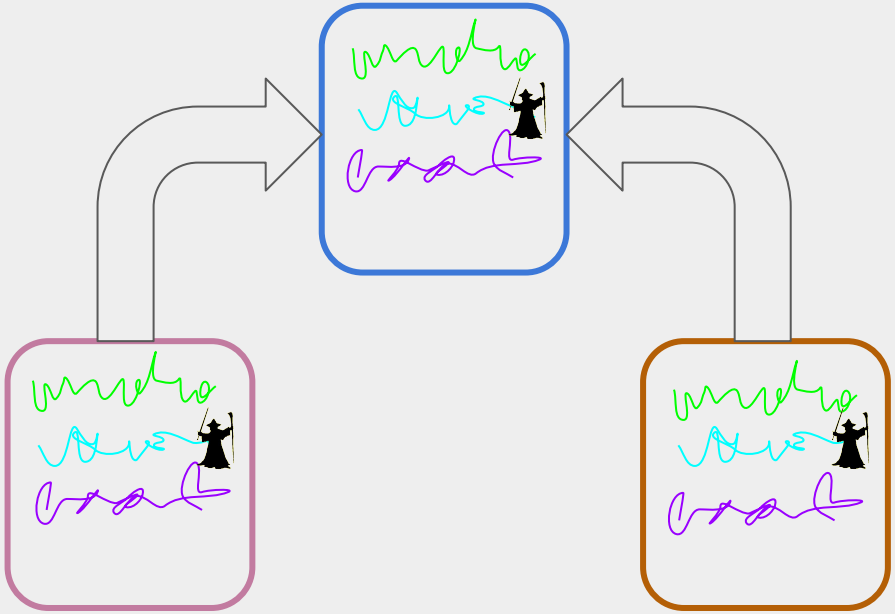
What is a Service Mesh?



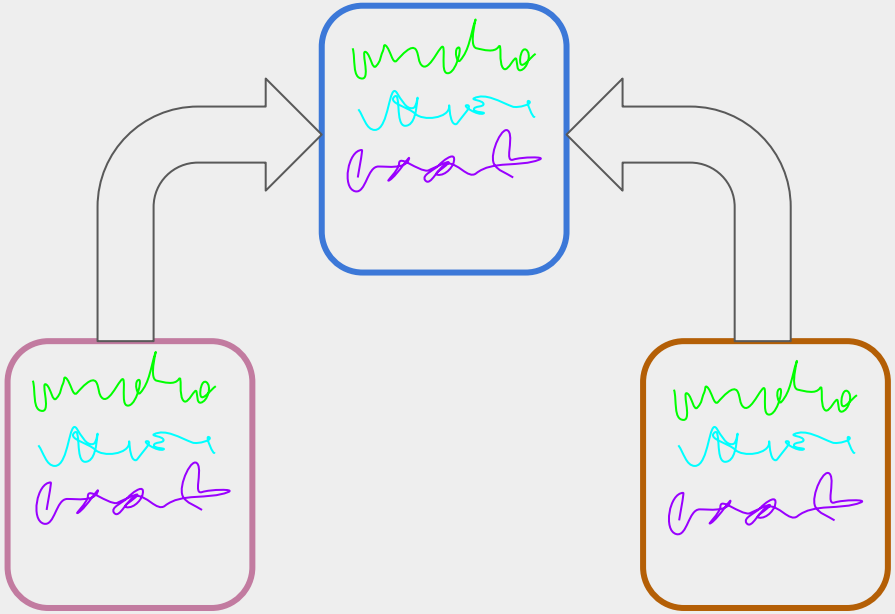
What is a Service Mesh?

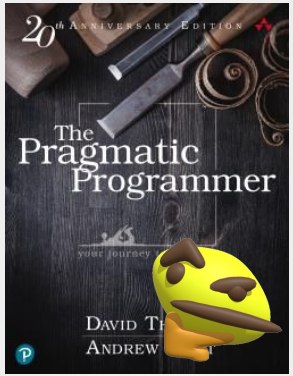


What is a Service Mesh?

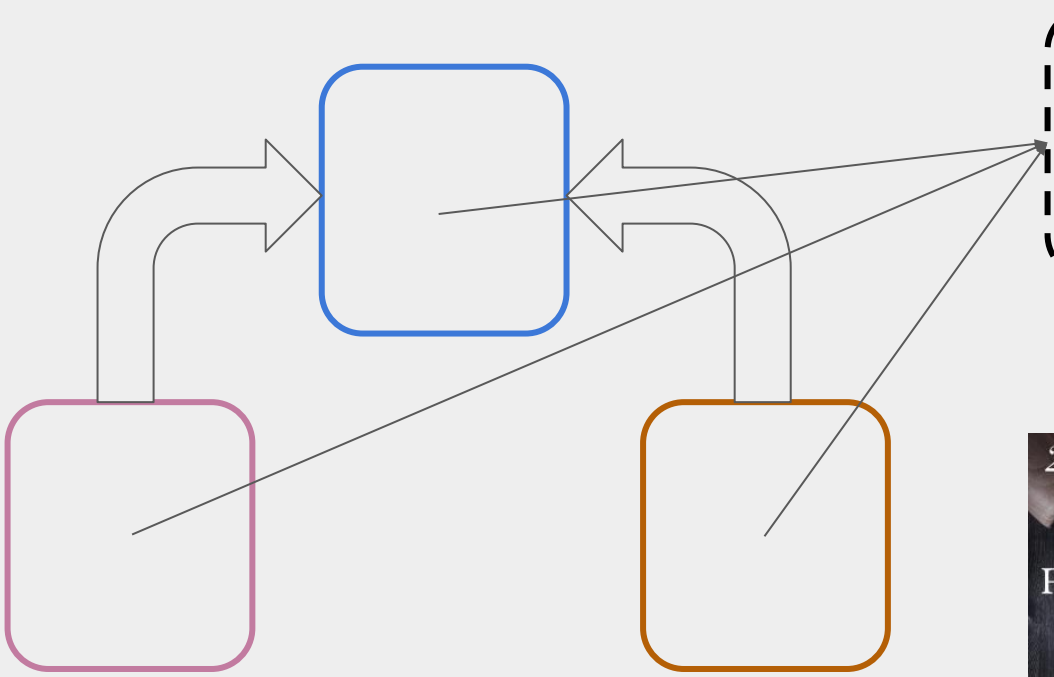


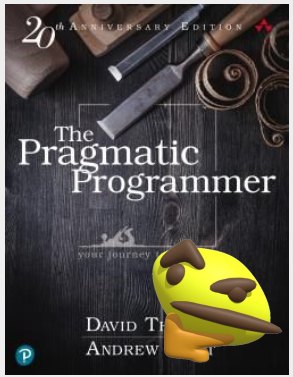
What is a Service Mesh?



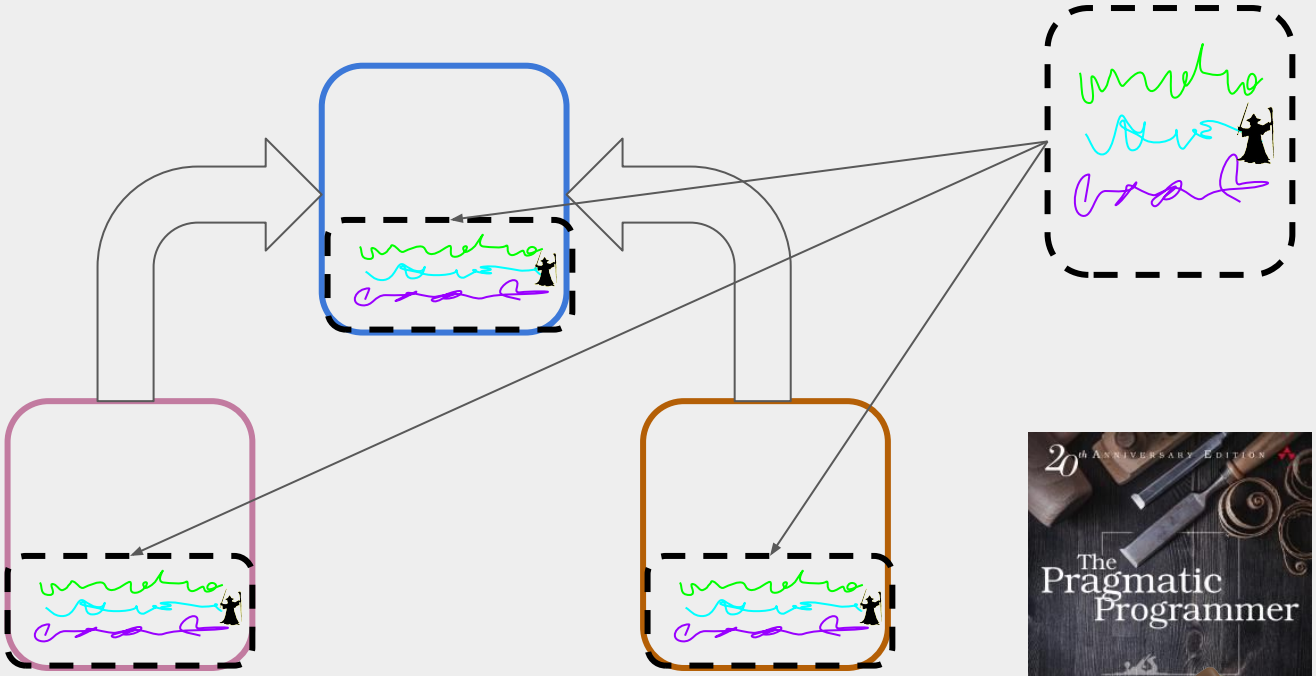


What is a Service Mesh?

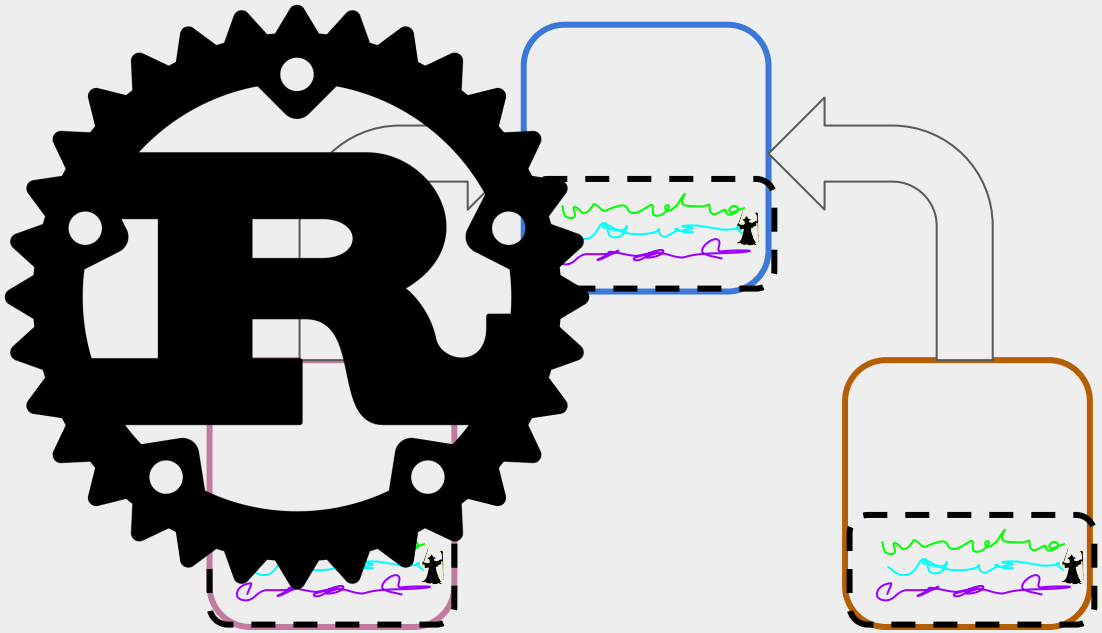




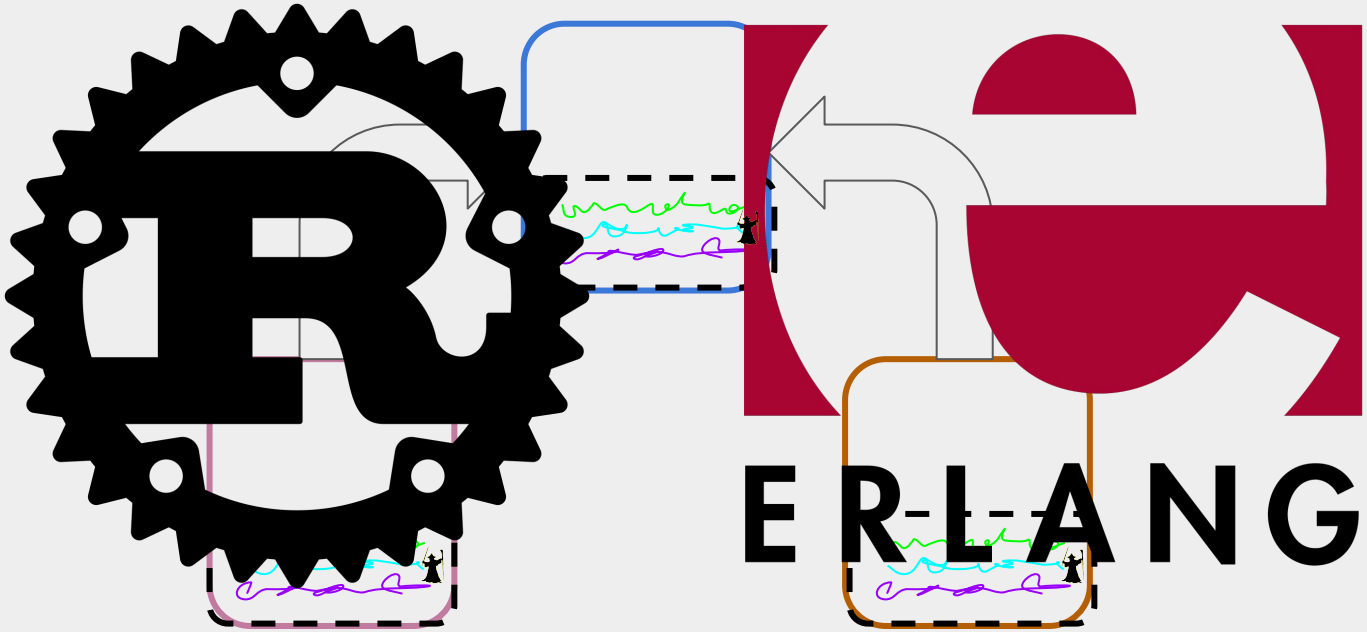
What is a Service Mesh?



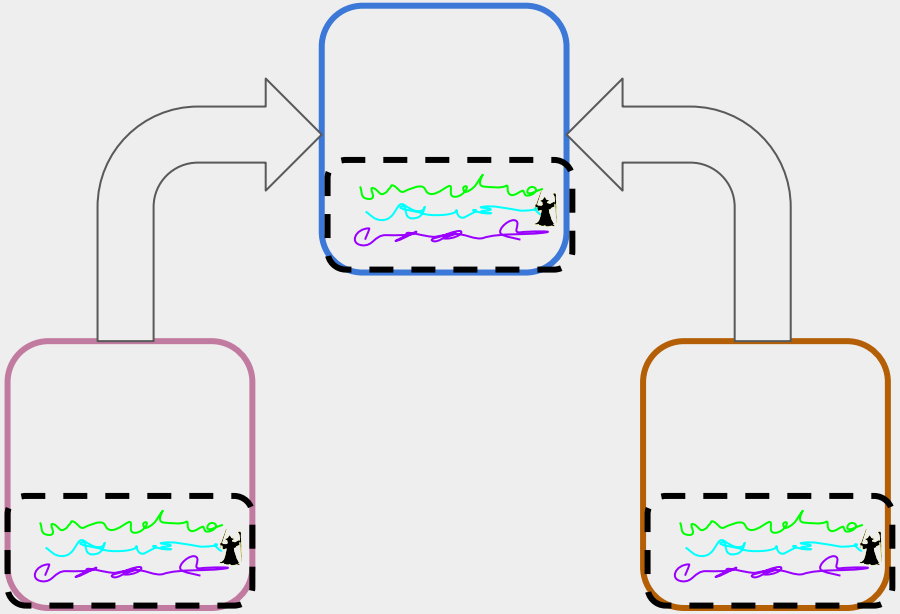
What is a Service Mesh?



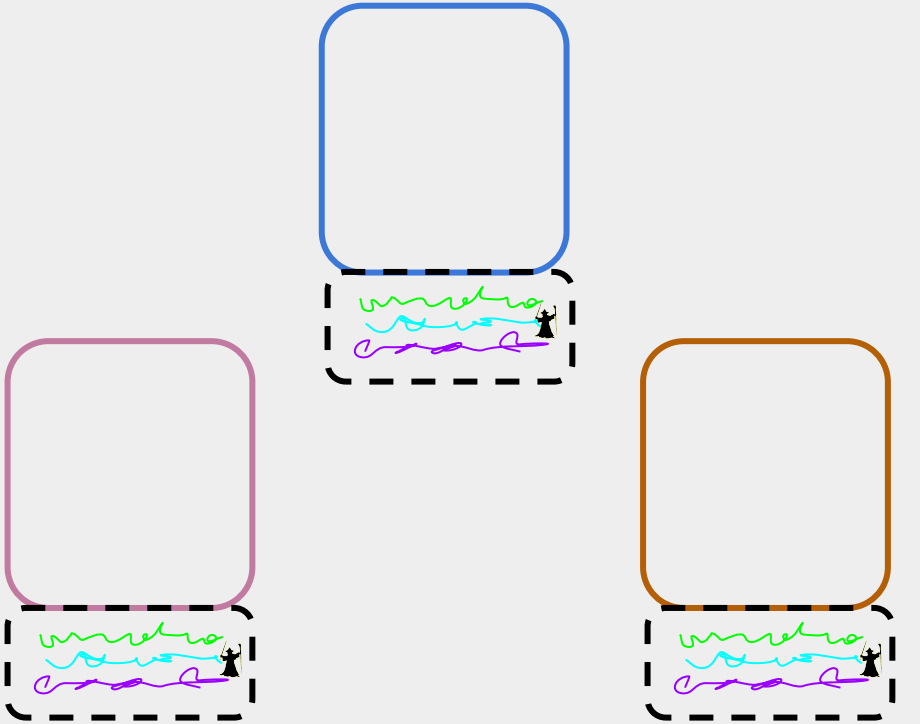
What is a Service Mesh?



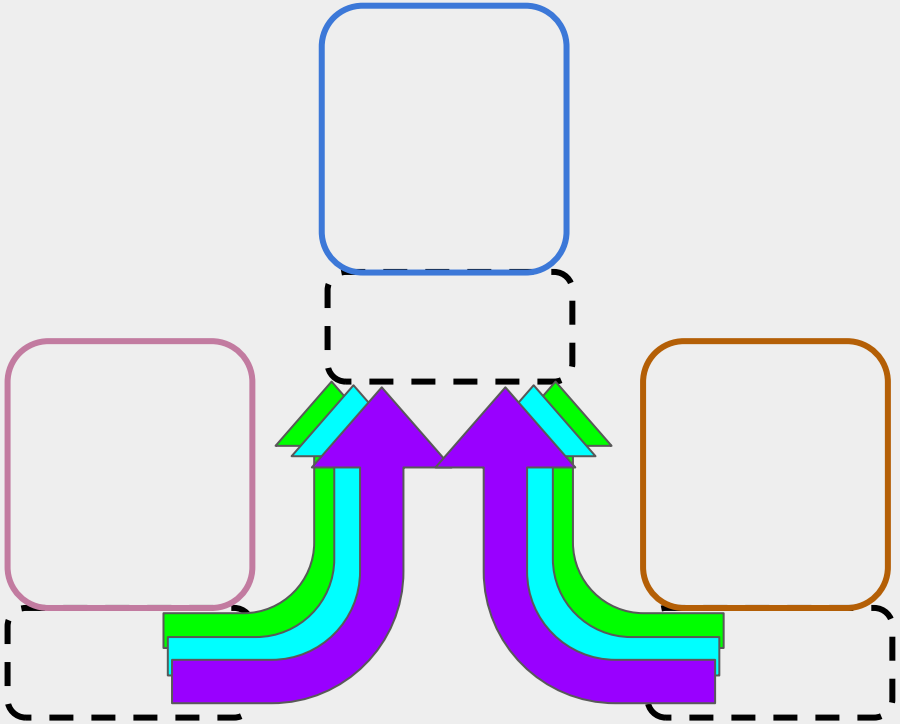
What is a Service Mesh?



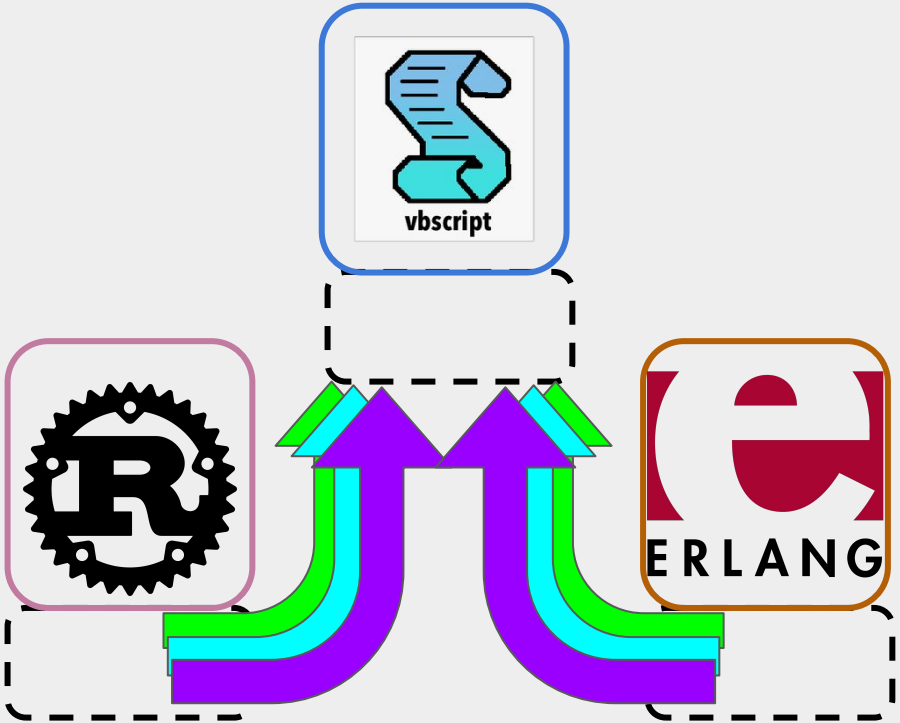
What is a Service Mesh?



What is a Service Mesh?



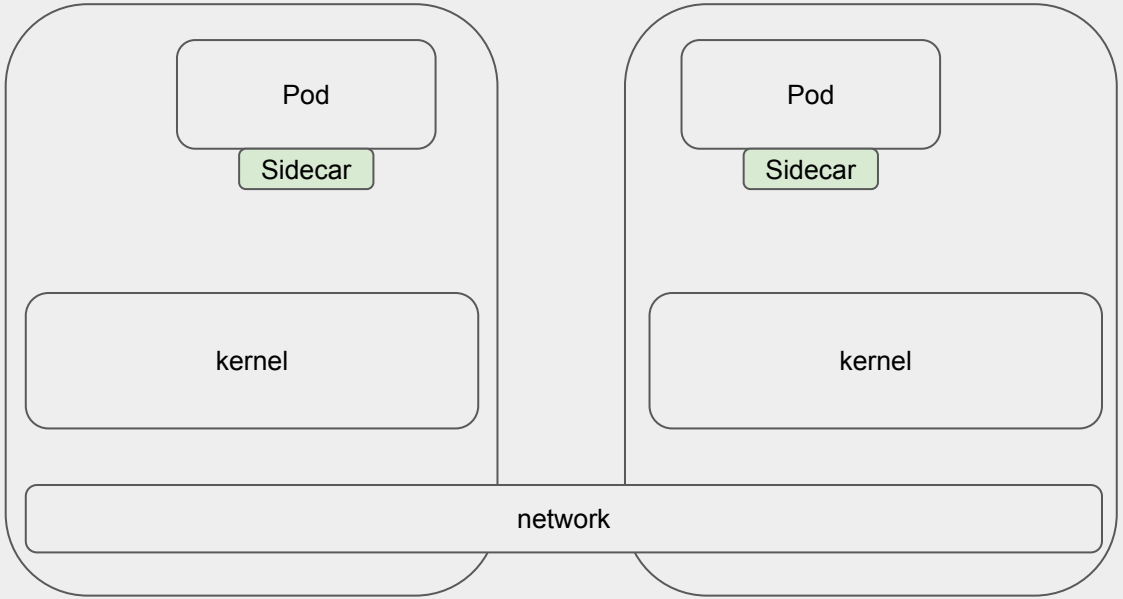
What is a Service Mesh?



The most popular Meshes offer:

- Observability
- Identity
- Encryption
- Access Control
- Load Balancing

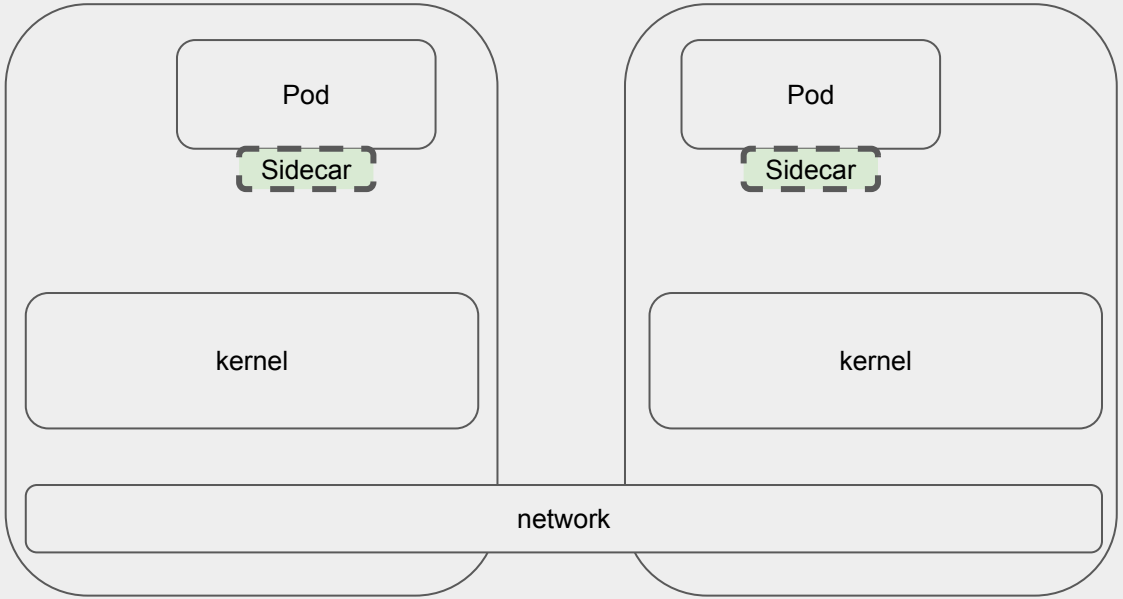
service mesh



control

data

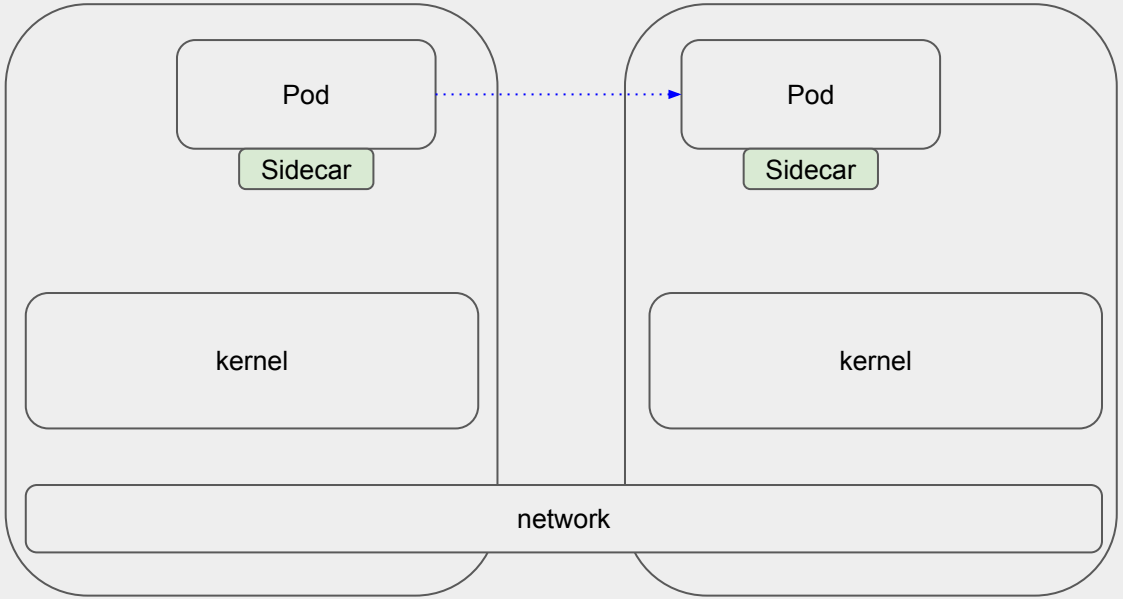
service mesh



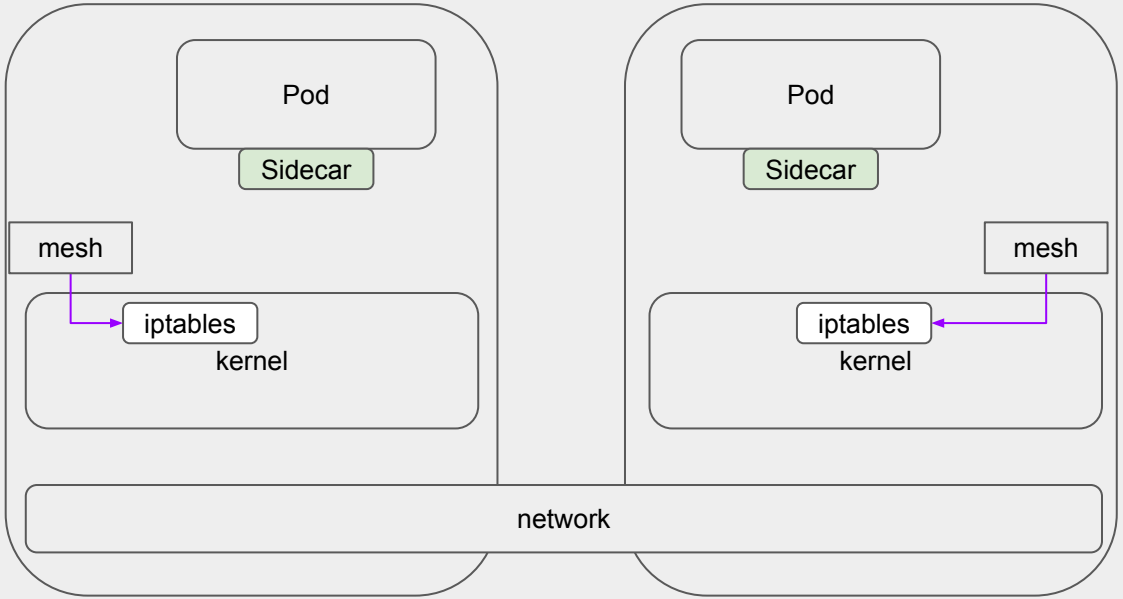
control

data

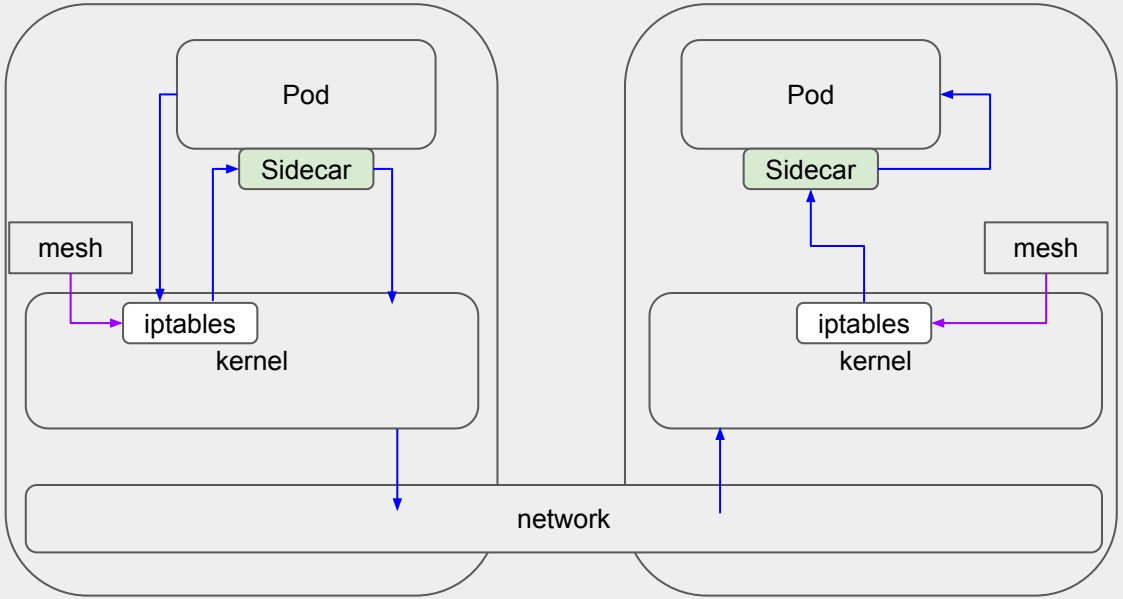
service mesh



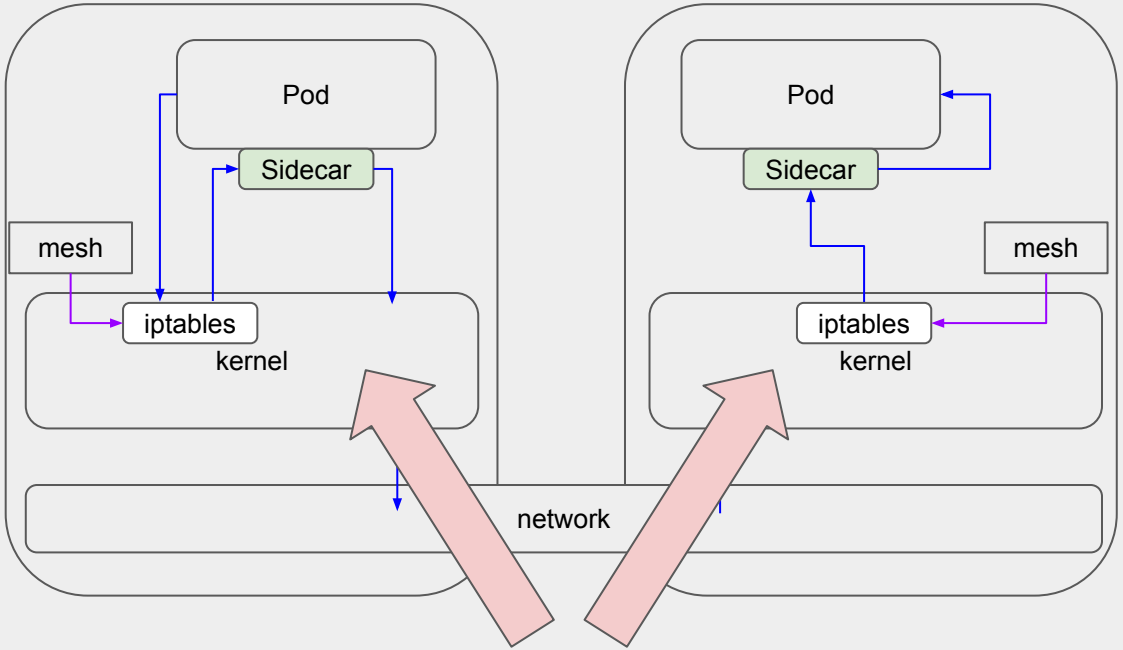
service mesh



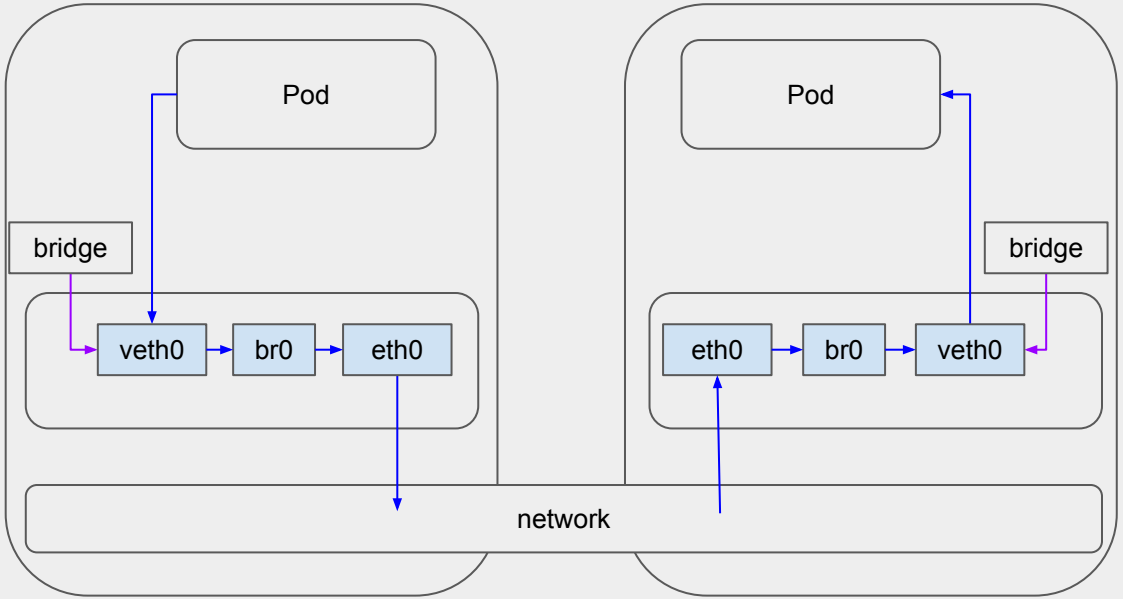
service mesh



service mesh



bridge plugin

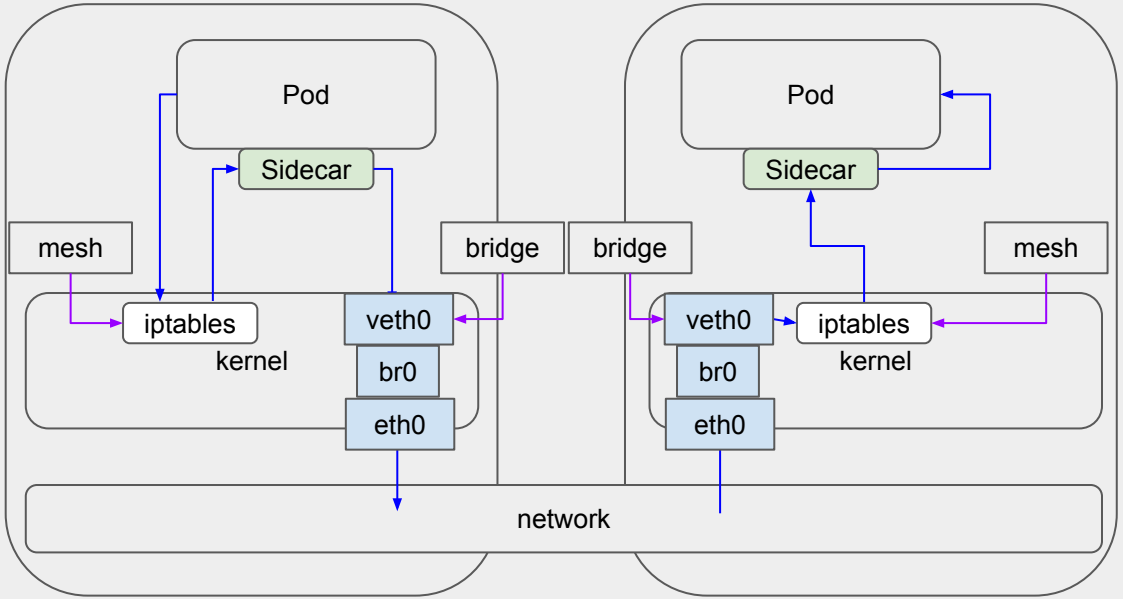


https://github.com/containernetworking/plugins

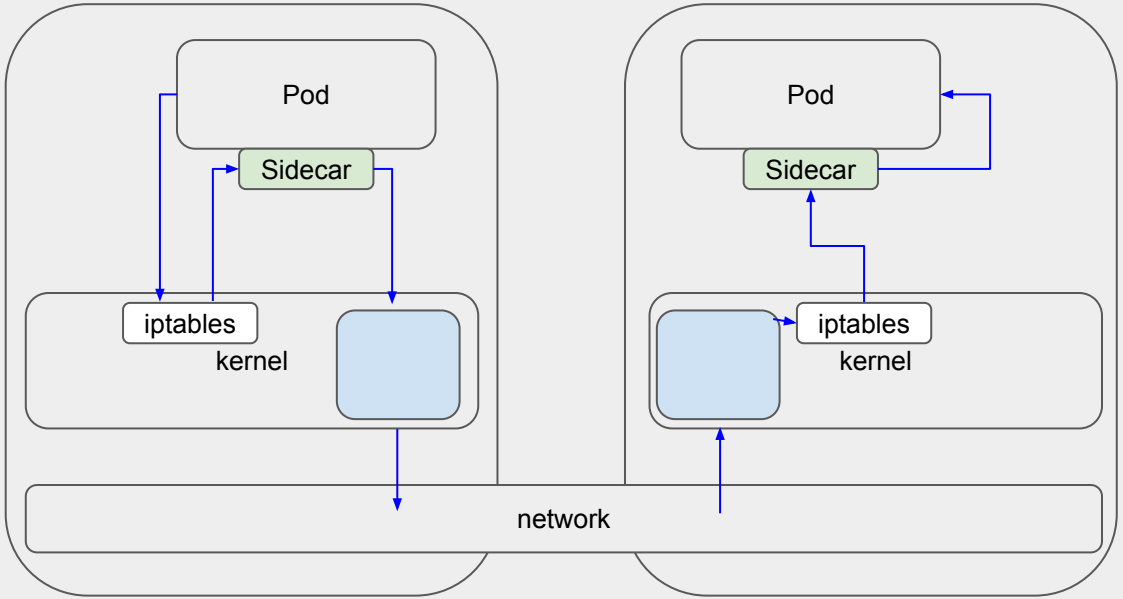
control

data

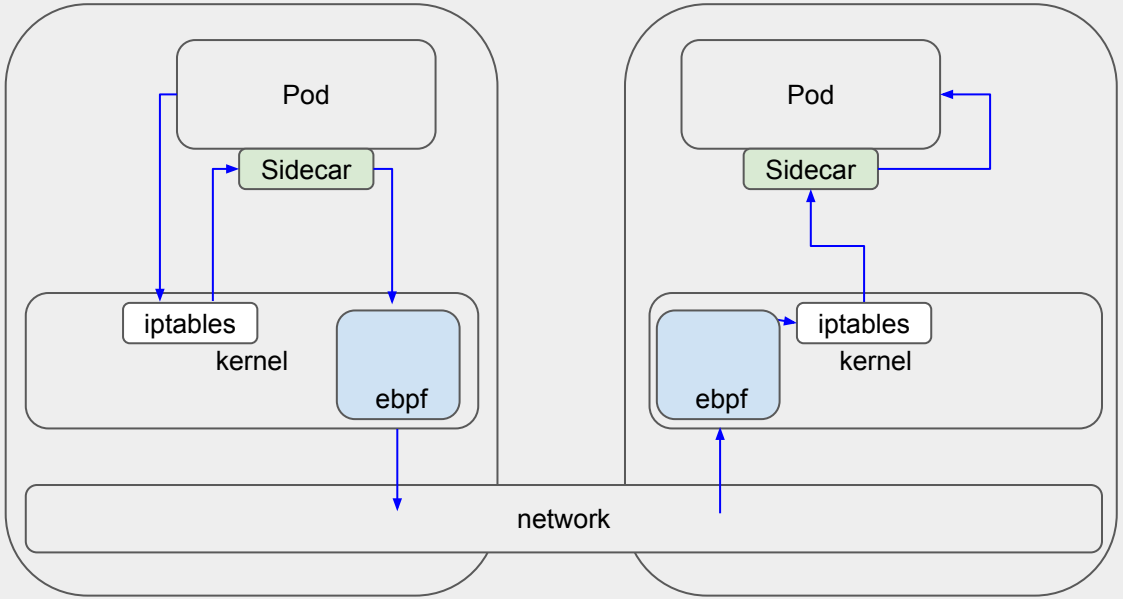
service mesh with bridge plugin



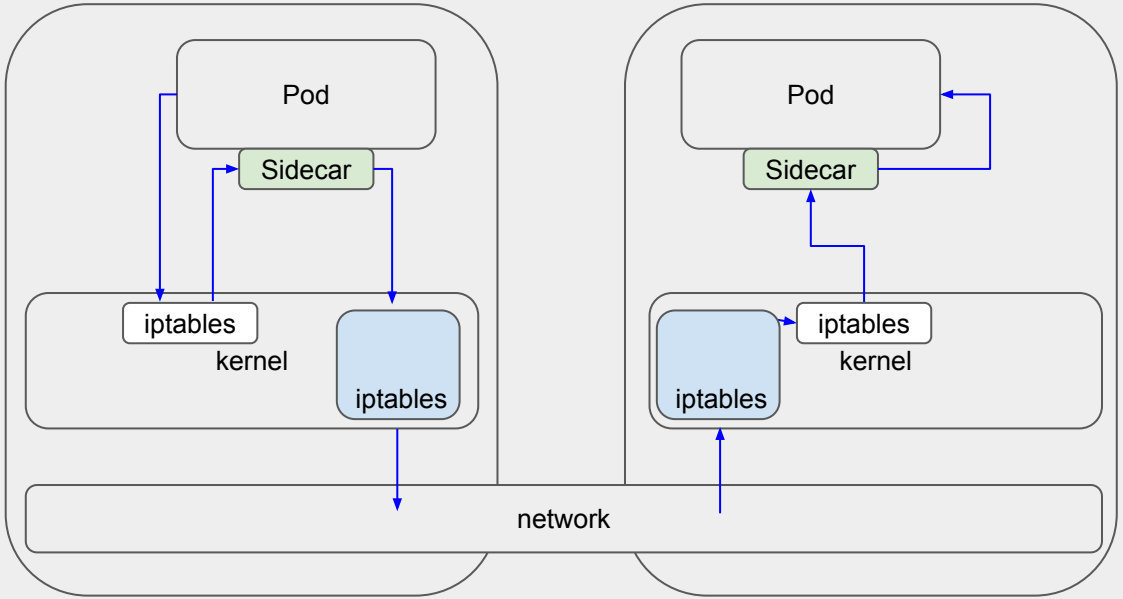
CNI + service mesh



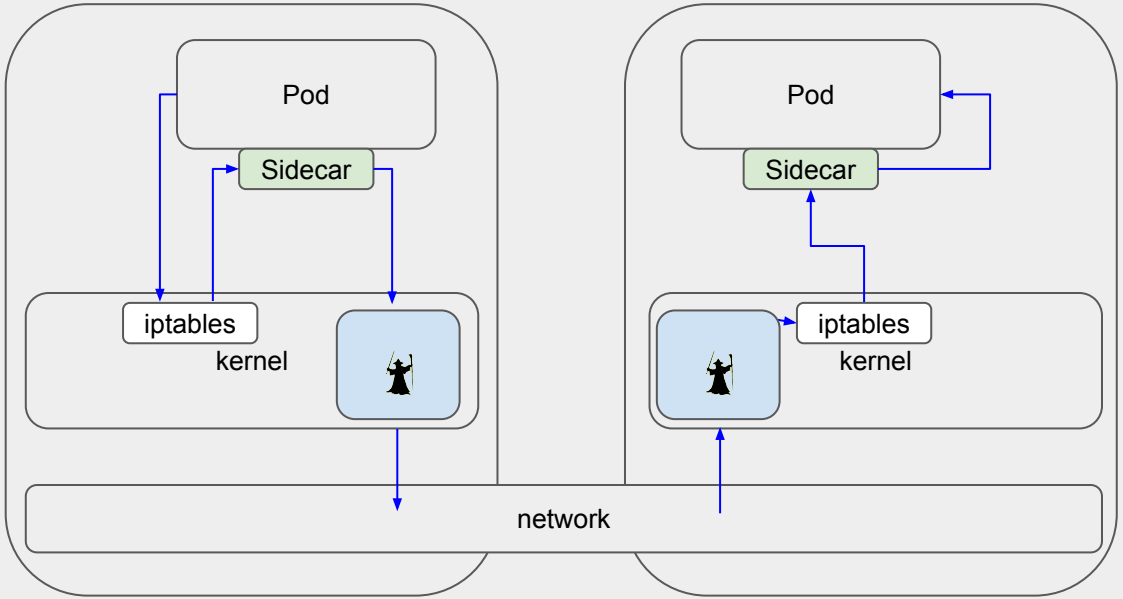
CNI + service mesh



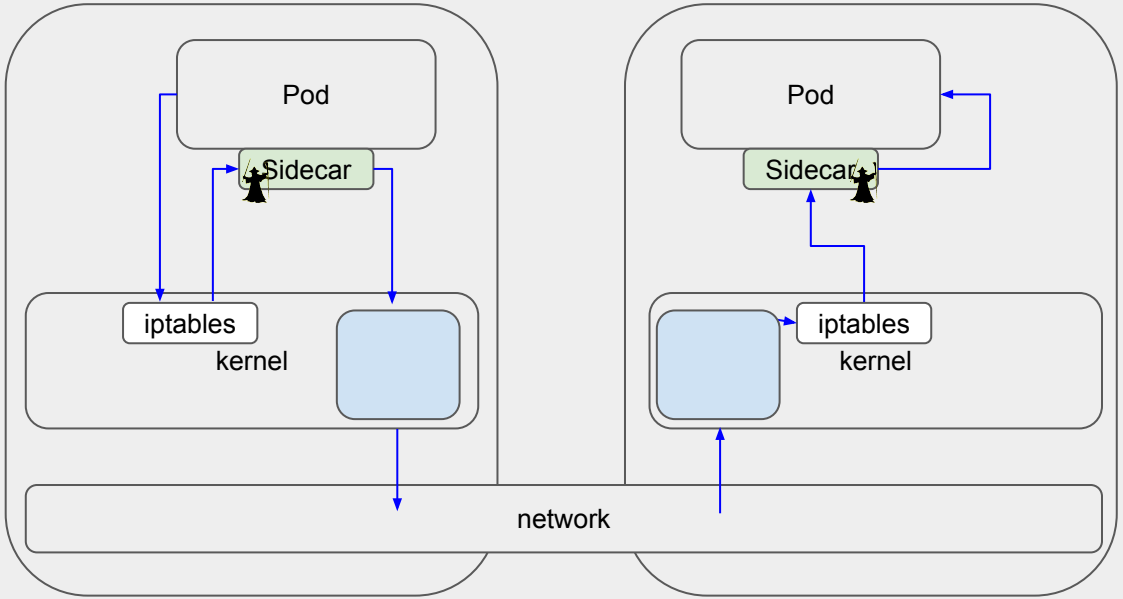
CNI + service mesh



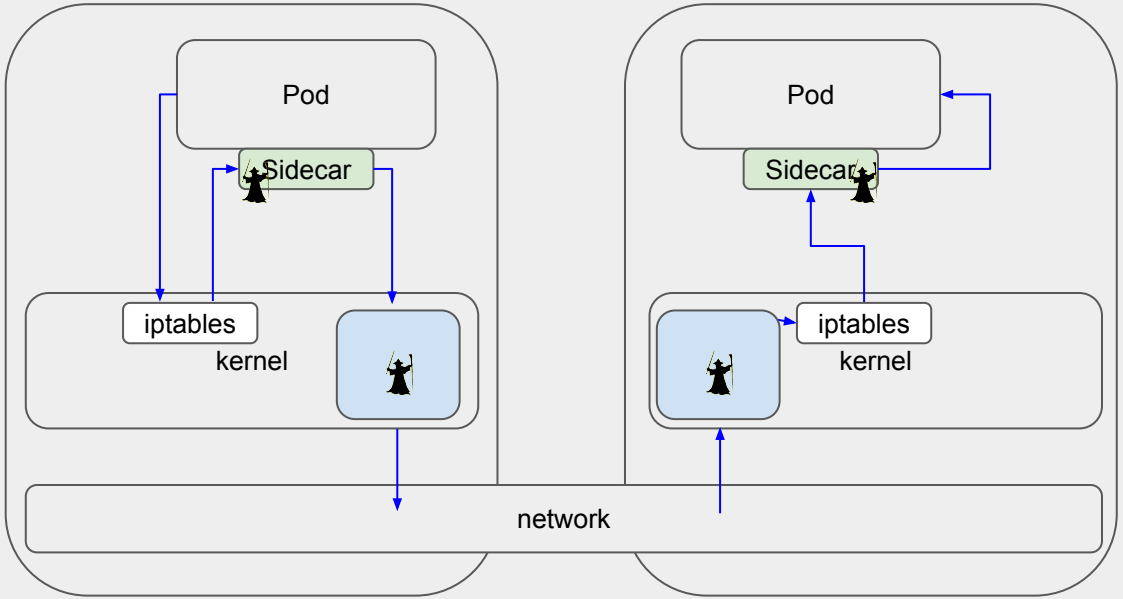
CNI + service mesh



CNI + service mesh



CNI + service mesh




WIZARD FIGHT!

CNI - what shall not pass?

CNI - ~~what shall not pass?~~
- what is enforceable?

CNI - what is enforceable?

```
$ kubectl explain networkpolicy.spec
```

- Allows you to apply policy on traffic which:
  - is going to any* IP or CIDR
  - is going to pods that match some label selector
  - is going to specific port(s)
  - is going to a specific namespace(s)

* except loopback or host traffic

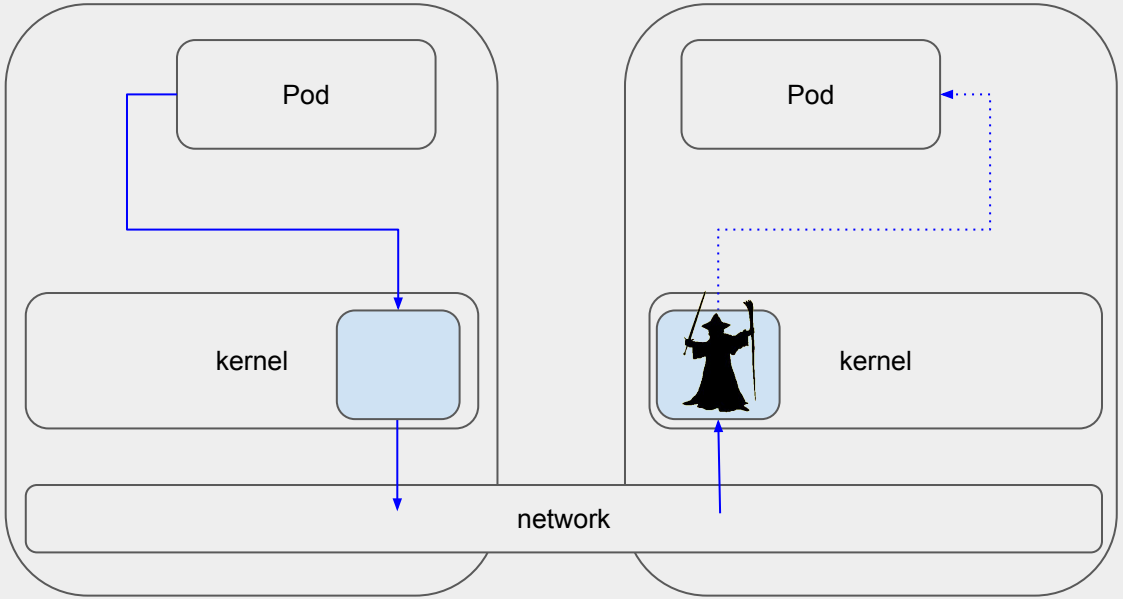
CNI - what is enforceable?

```
$ kubectl explain networkpolicy.spec
```

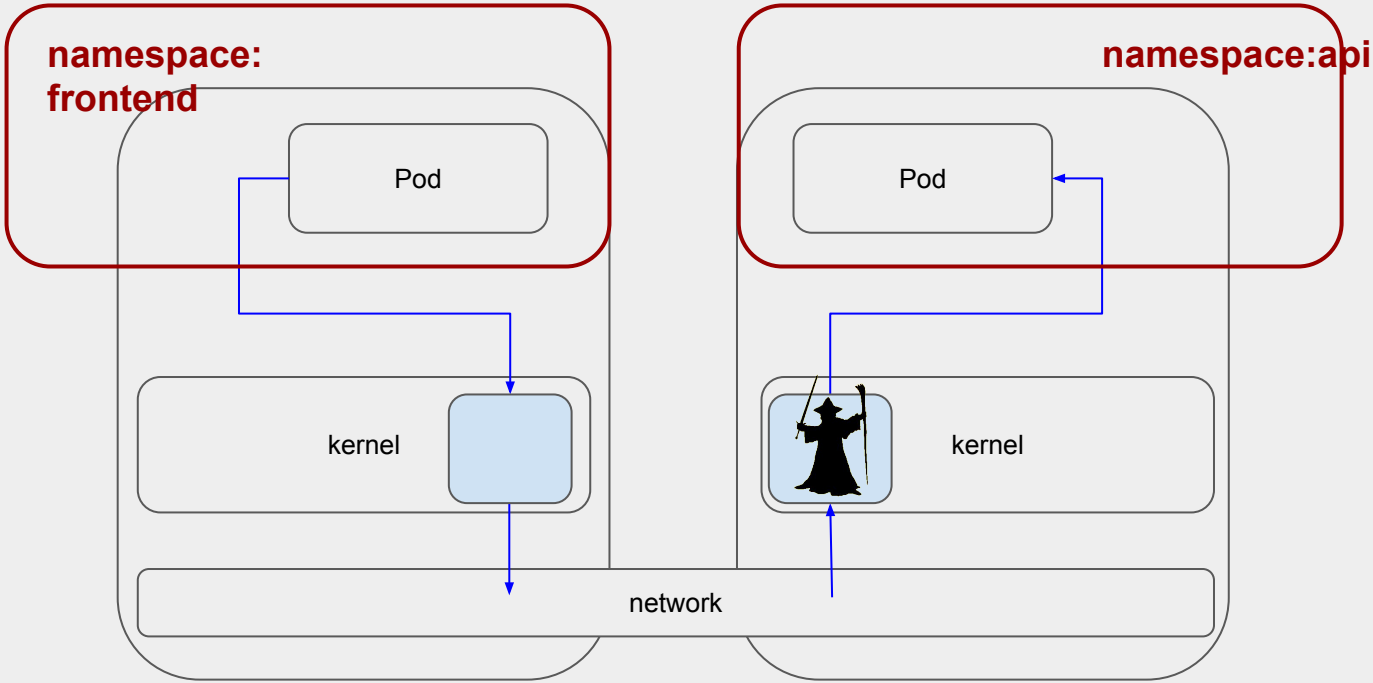
- Allows you to apply policy on traffic which:
  - is going to any* IP or CIDR
  - is going to pods that match some label selector
  - is going to specific port(s)
  - is going to a specific namespace(s)
- Allows you to conditionally block/permit traffic based on:
  - source IP or CIDR
  - source pods that match some label selector
  - source namespace

* except loopback or host traffic

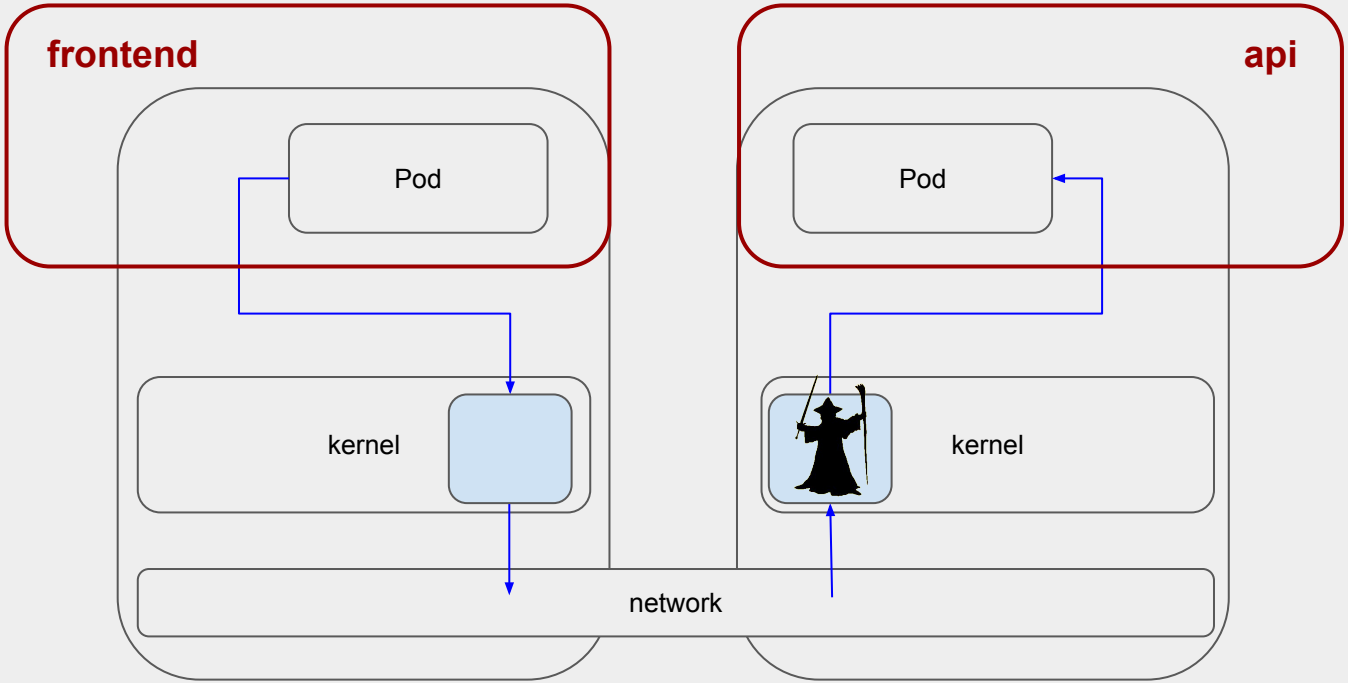
CNI - how is it enforced?



CNI - how is it enforced?

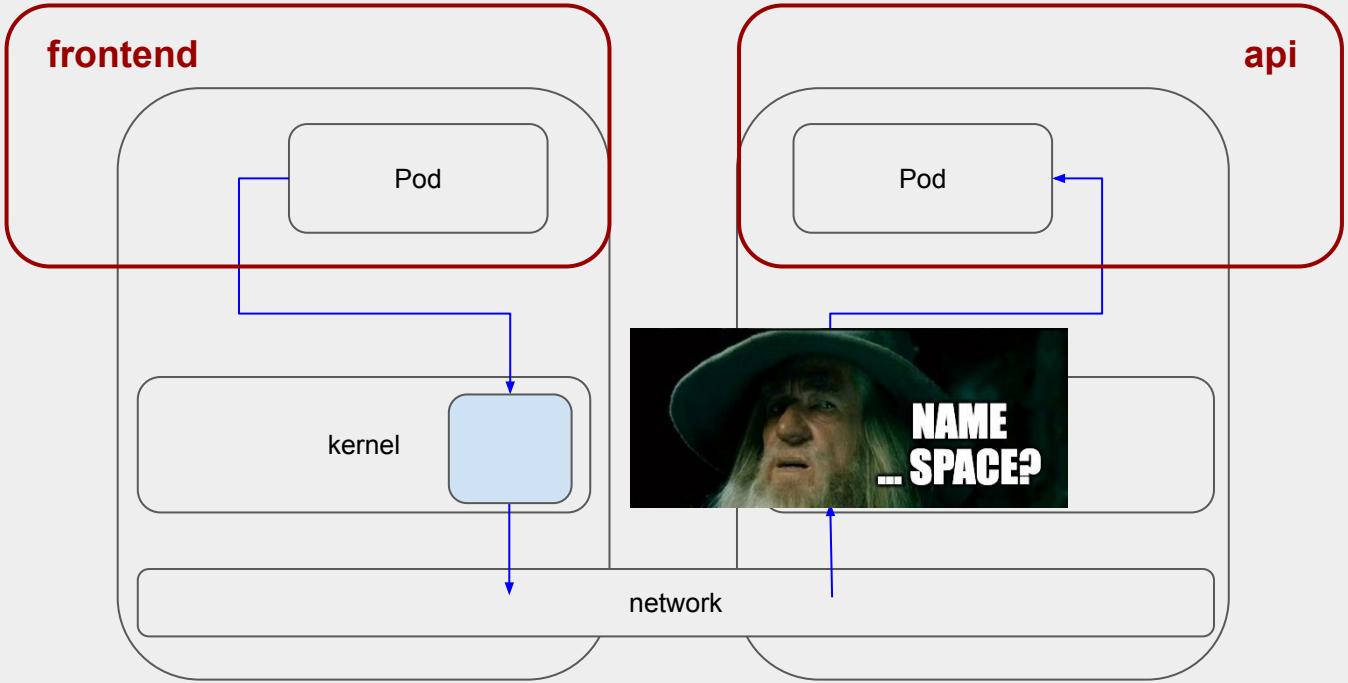


CNI - how is it enforced?



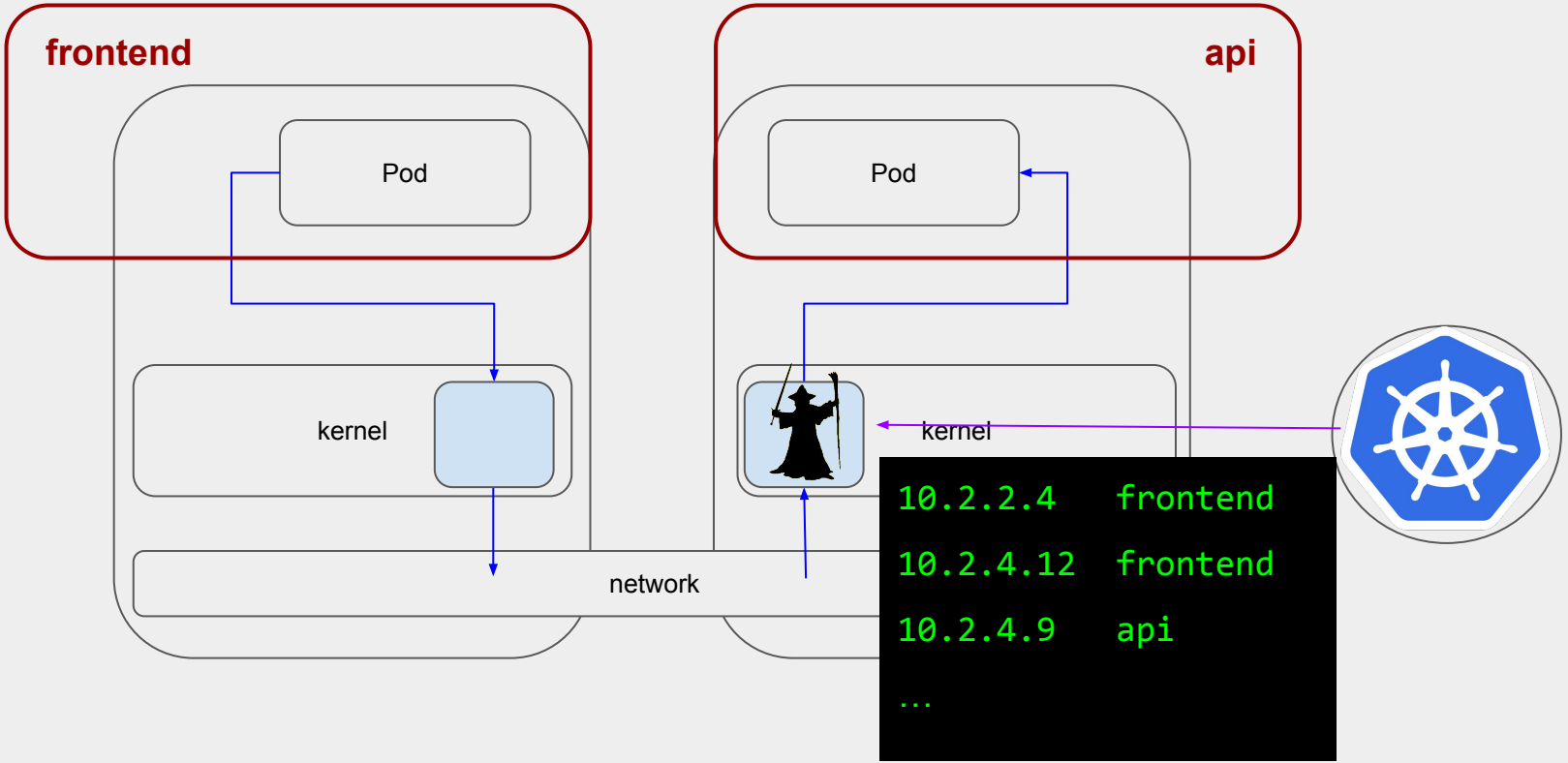

```
$ dear CNI: [pods in frontend] => [pods in api] == OK!
```

CNI - how is it enforced?

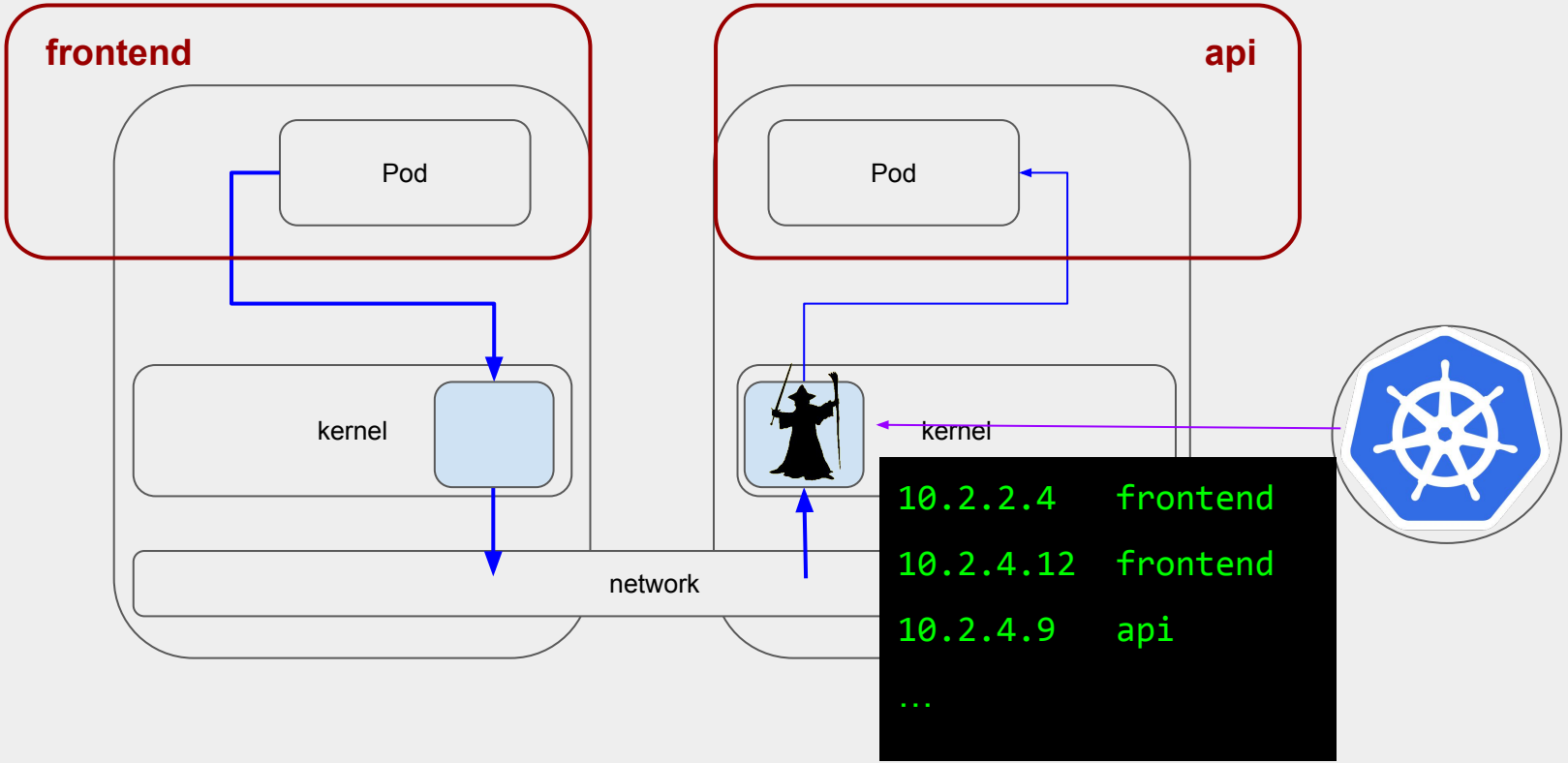



```
$ dear CNI: [pods in frontend] => [pods in api] == OK!
```

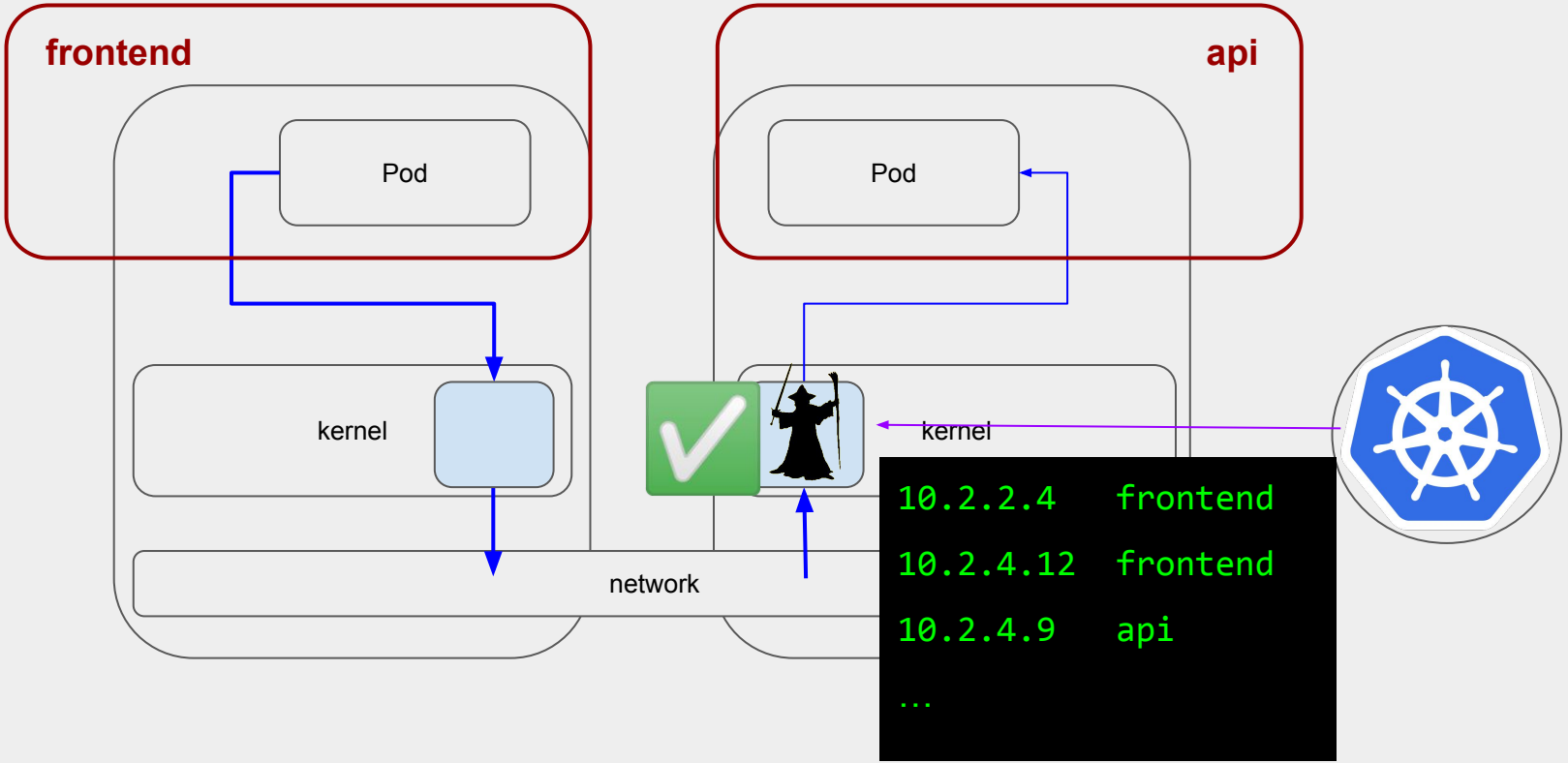
CNI - how is it enforced?



CNI - how is it enforced?

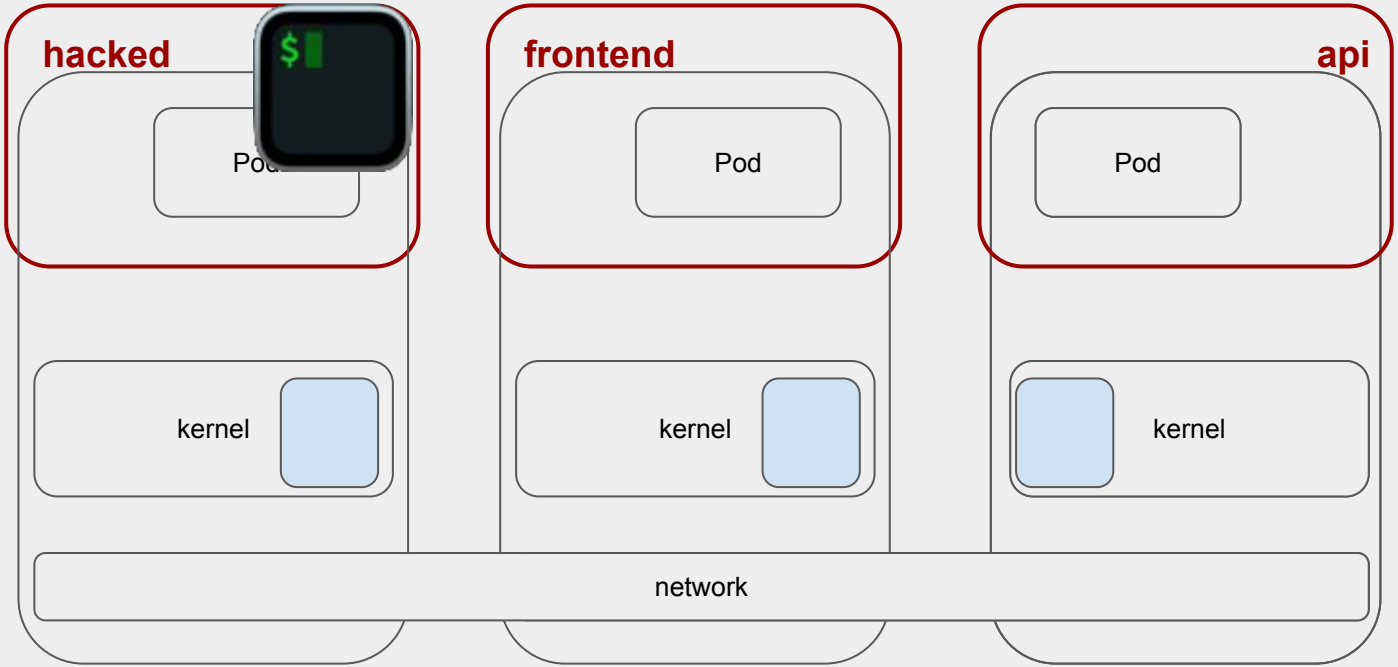


CNI - how is it enforced?



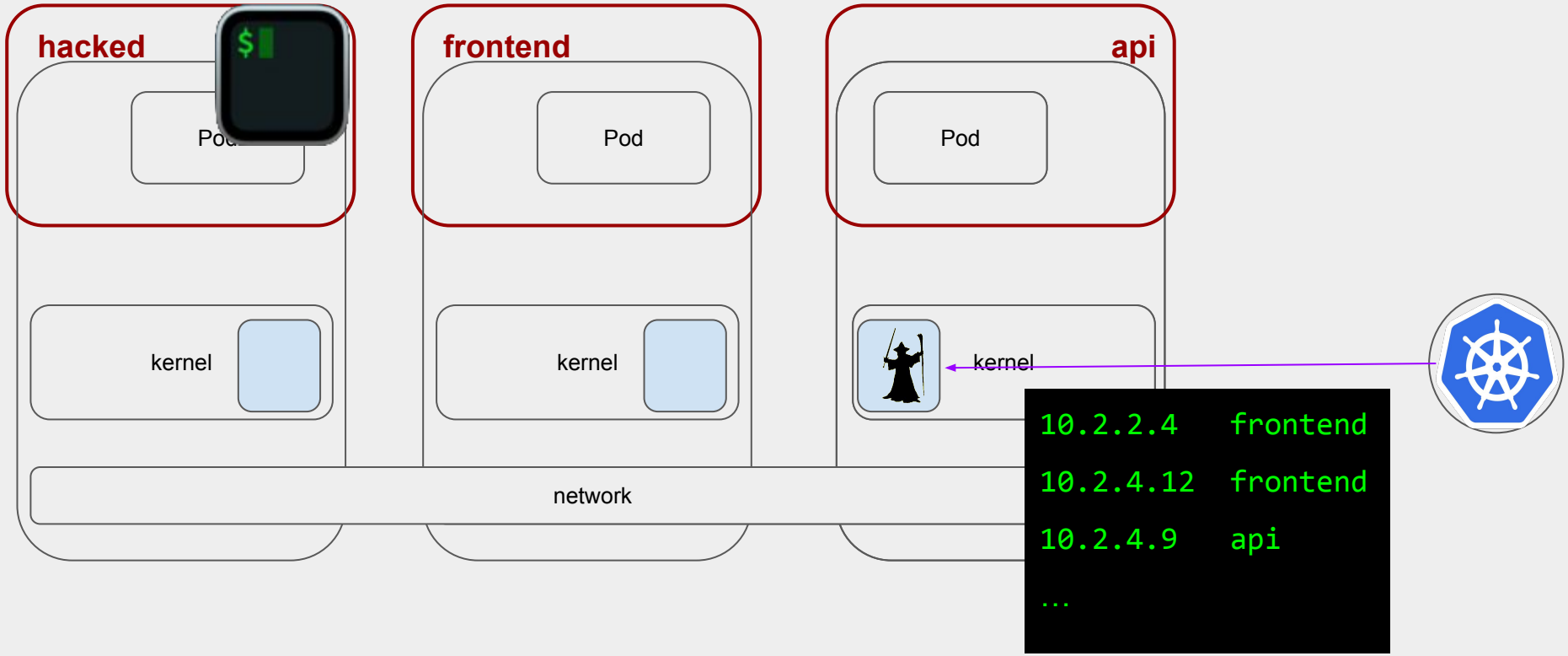
It’s time for a contrived scenario!

CNI - contrived scenario

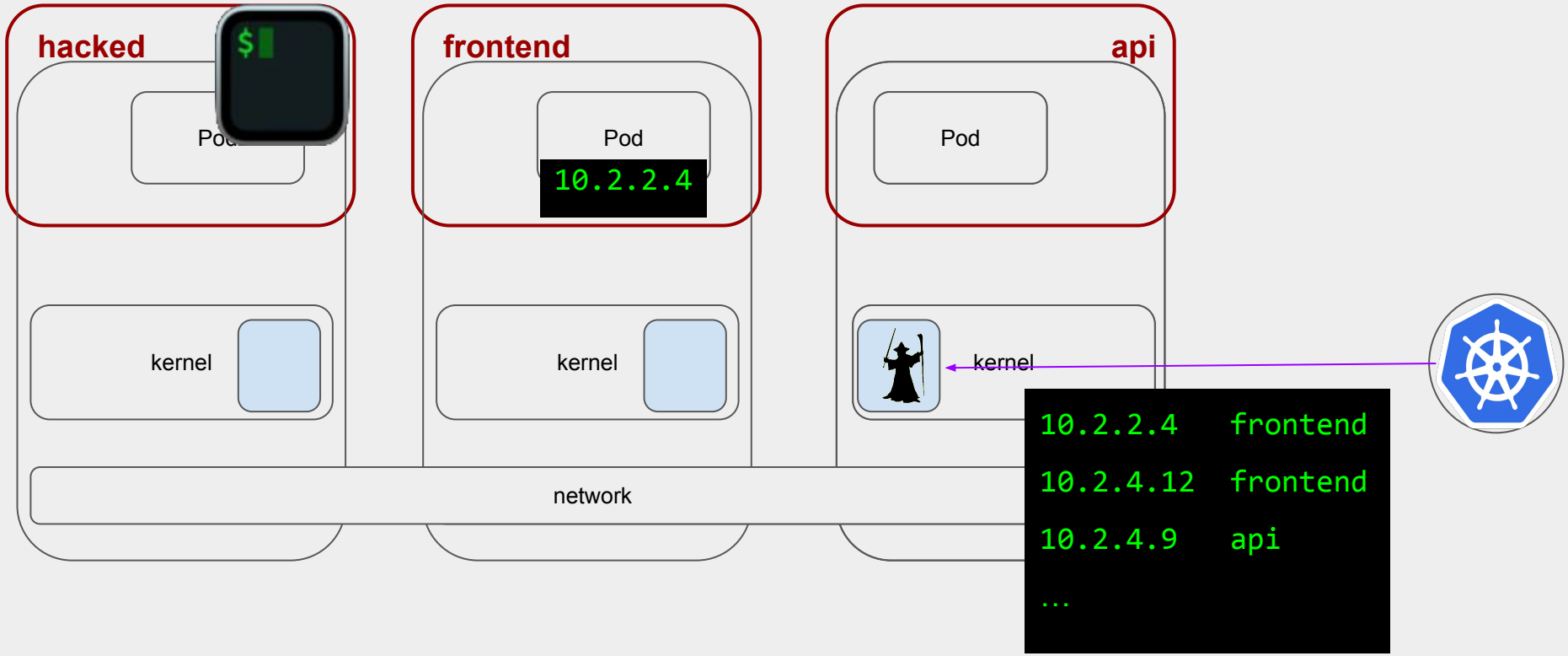



```
$ dear CNI: [pods in frontend] => [pods in api] == OK!
```

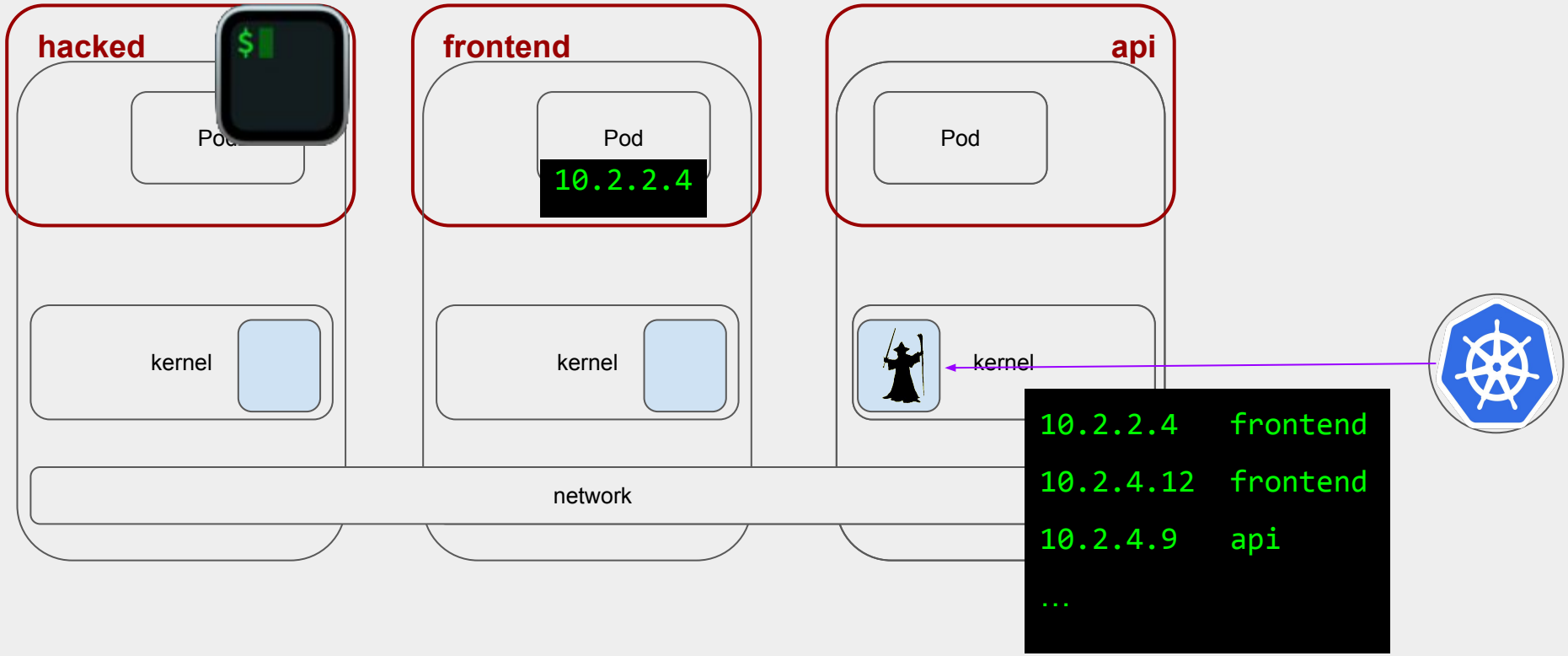
CNI - contrived scenario



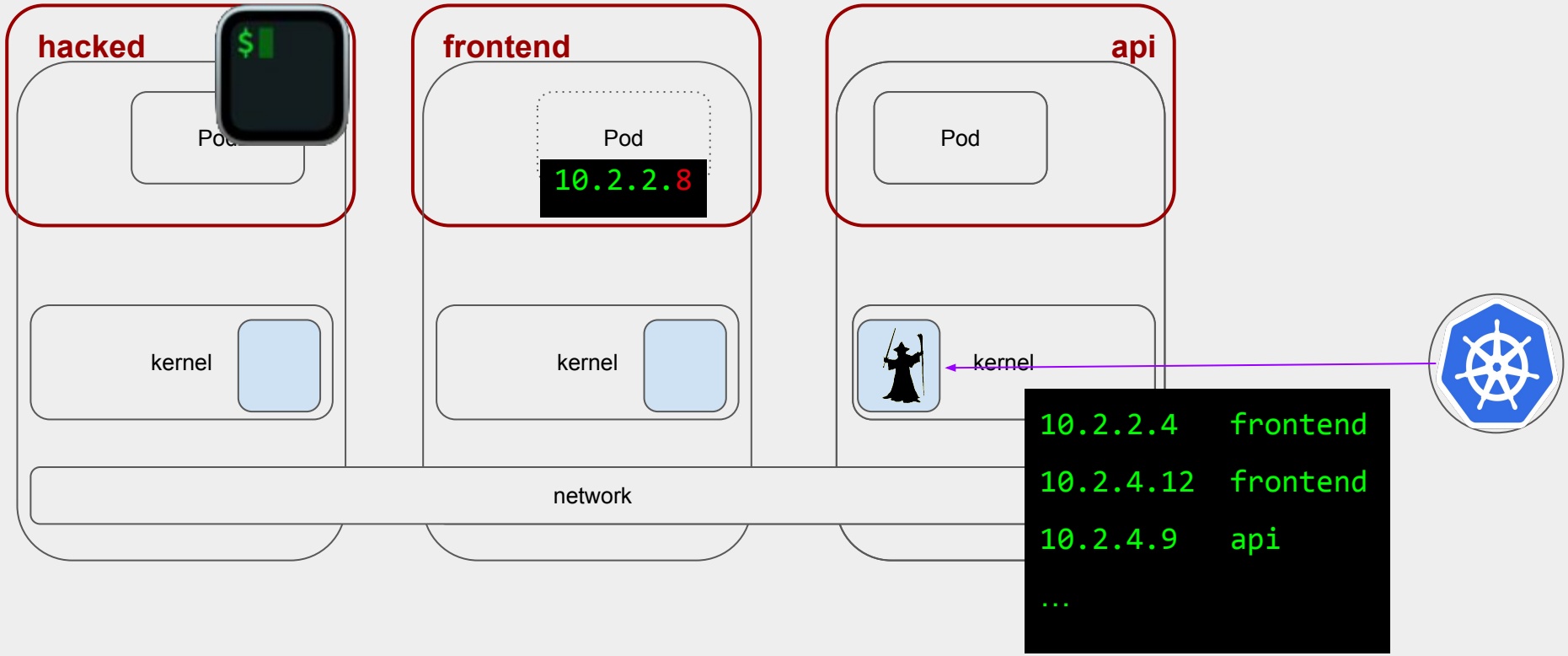
CNI - contrived scenario



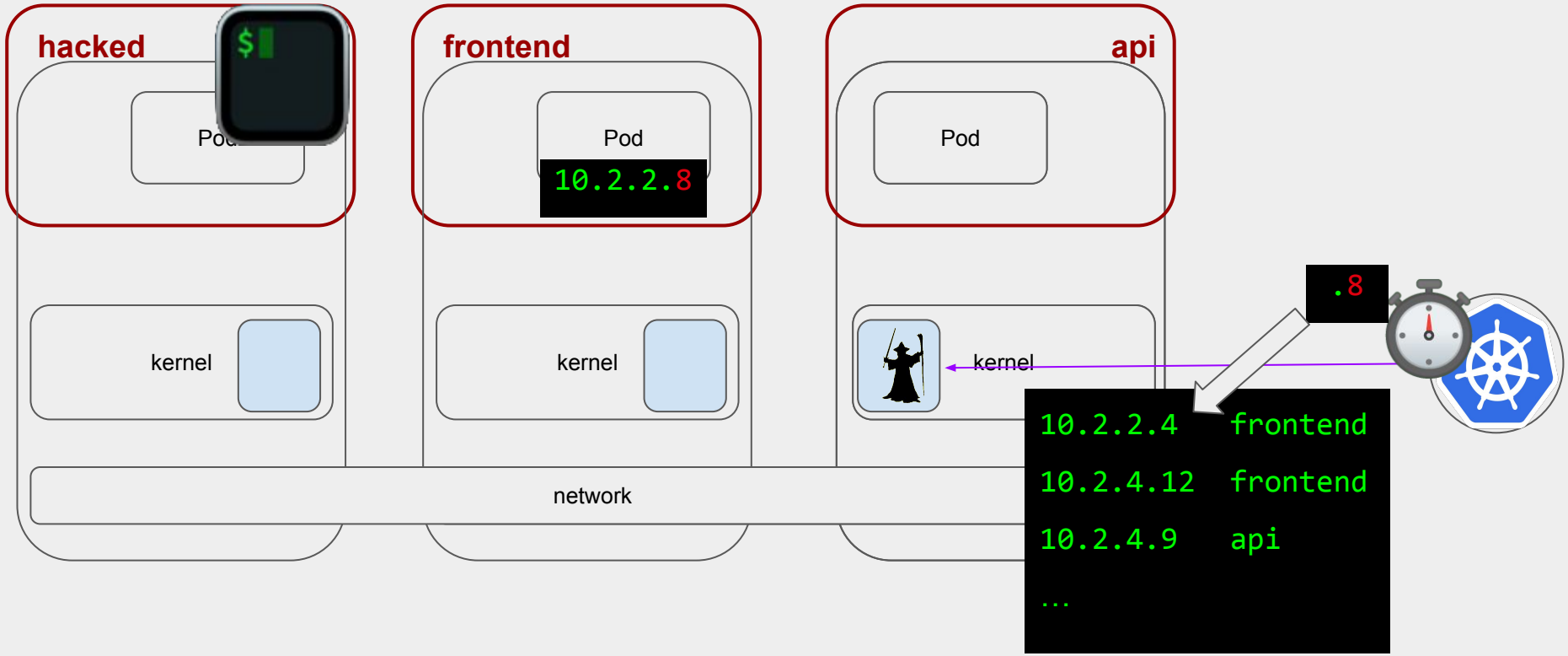
CNI - contrived scenario



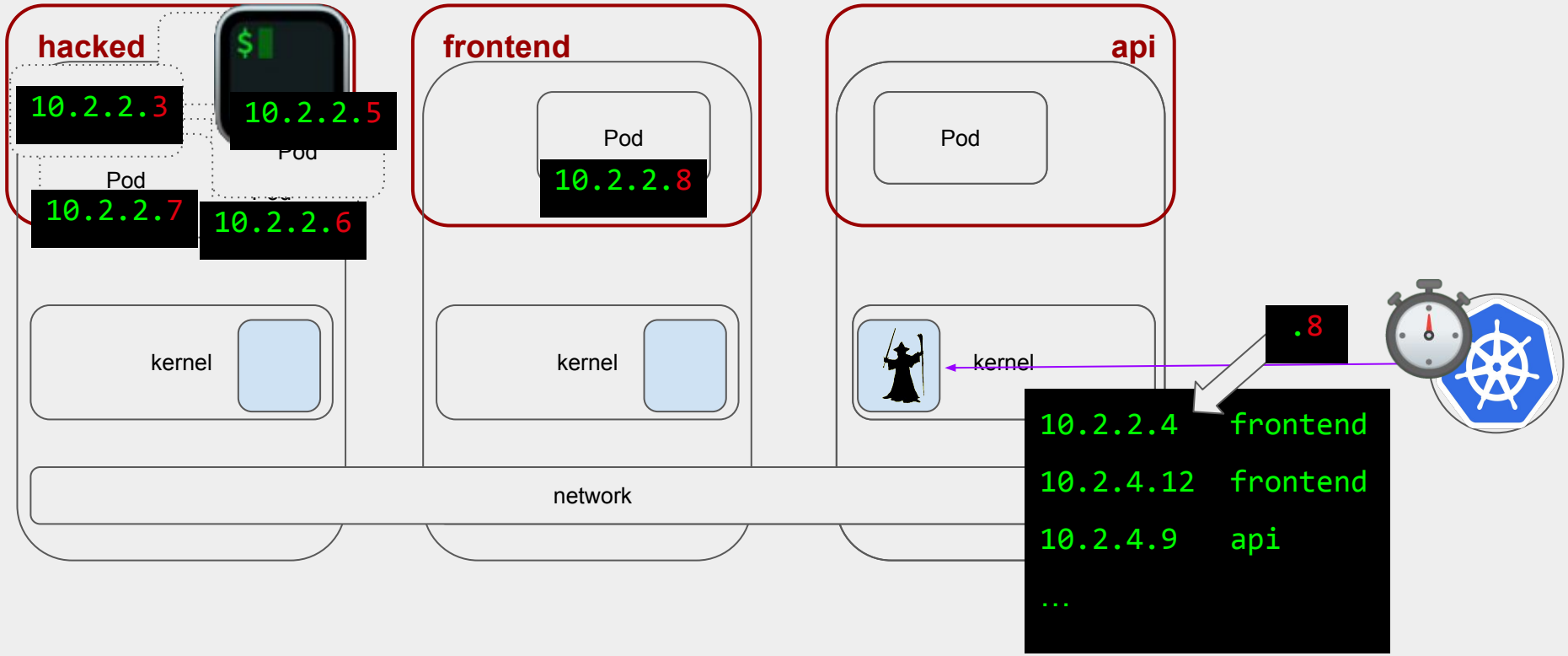
CNI - contrived scenario



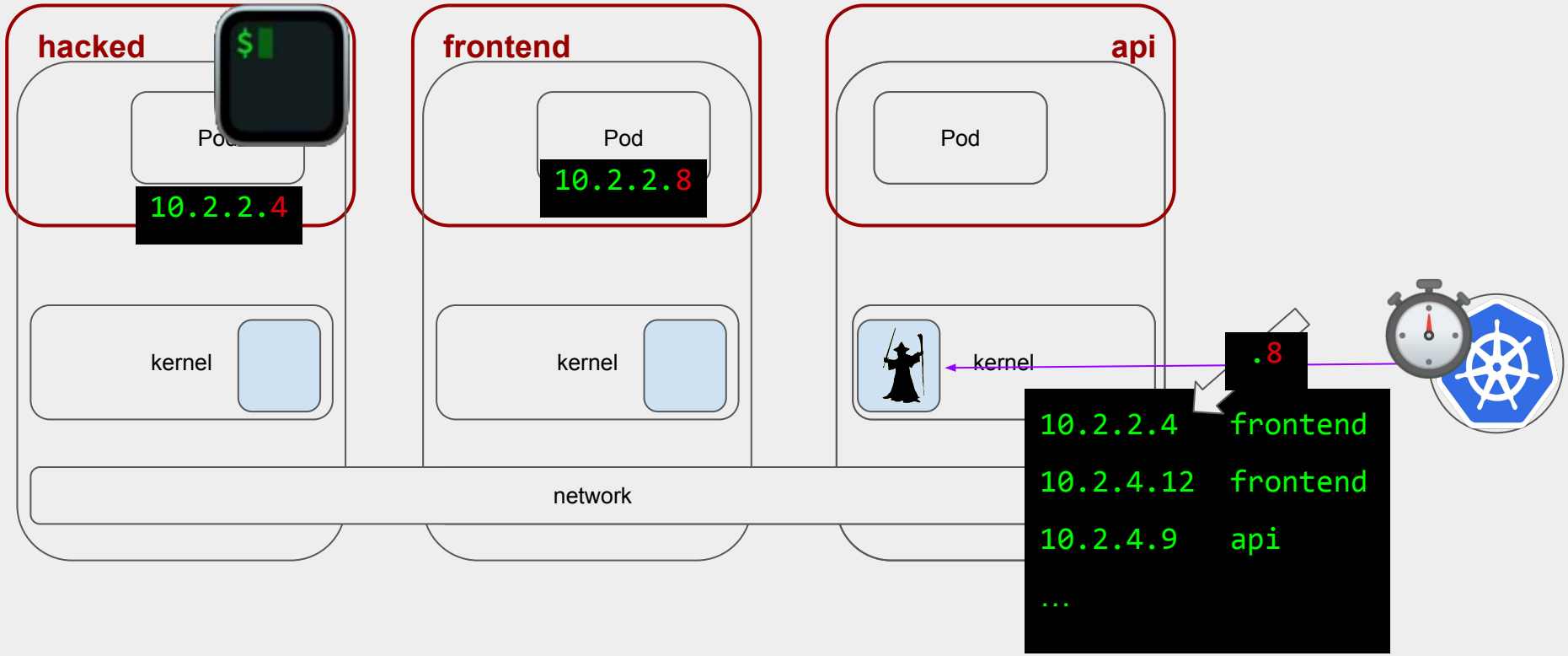
CNI - contrived scenario



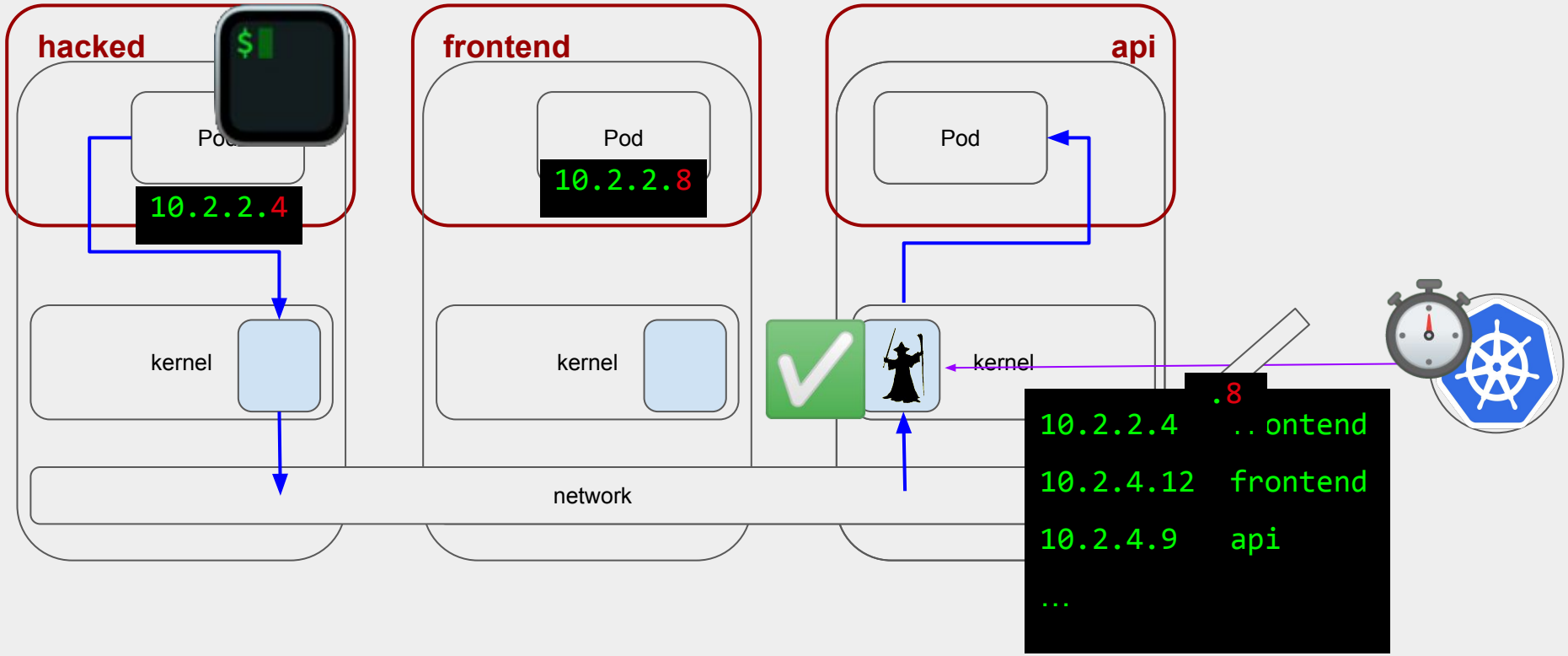
CNI - contrived scenario



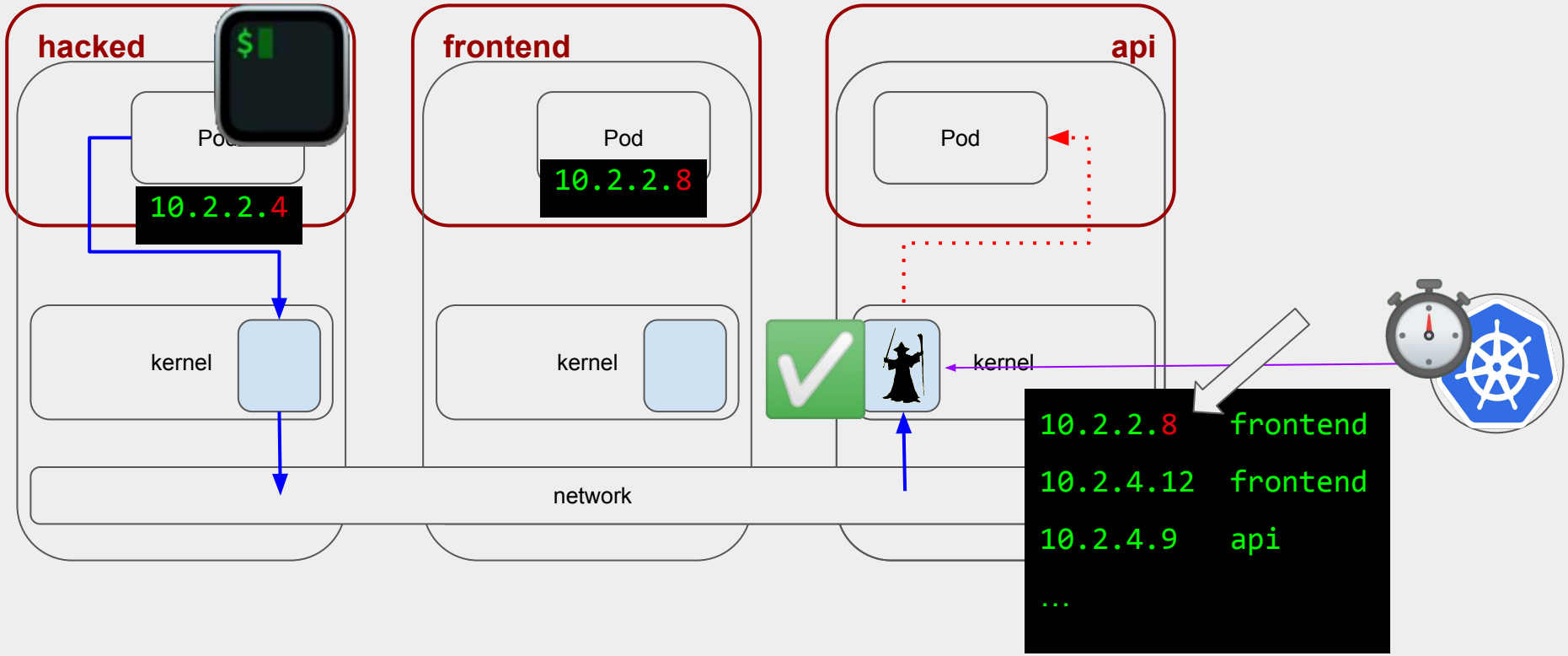
CNI - contrived scenario



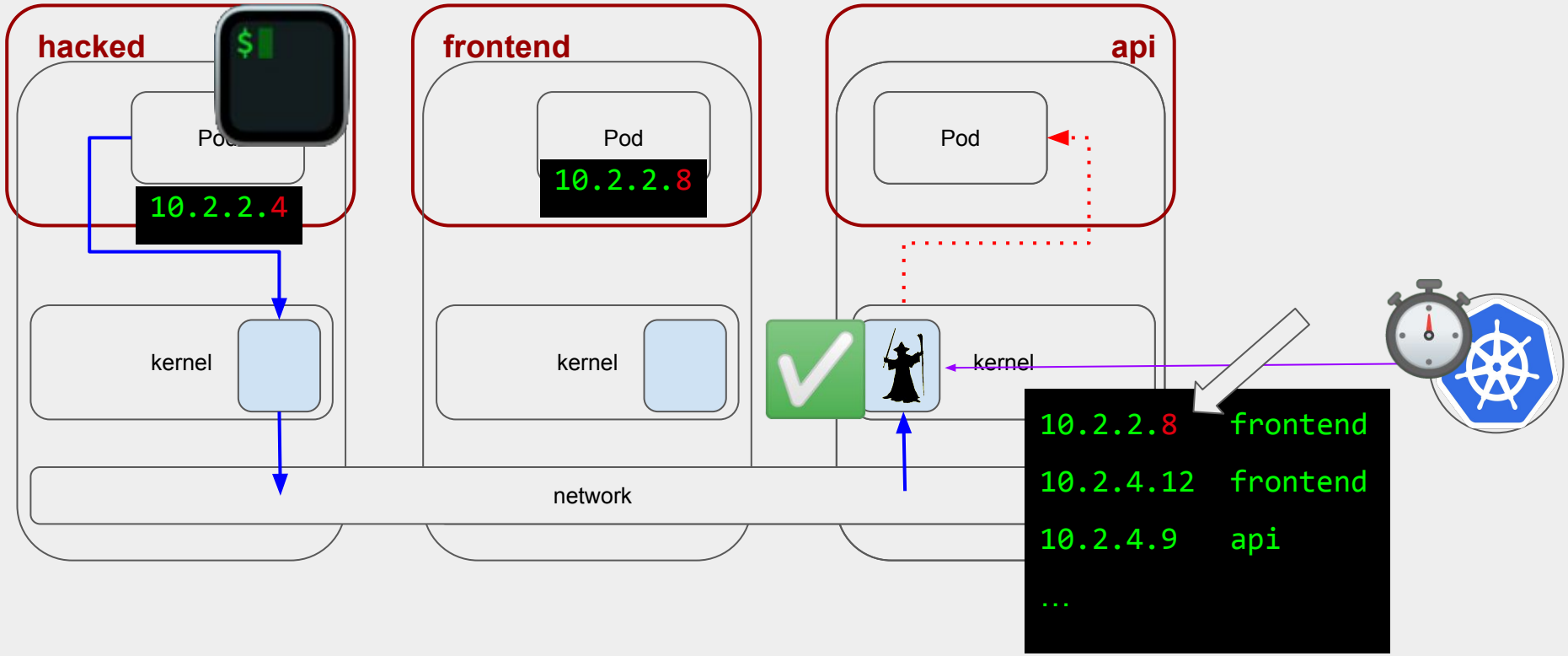
CNI - contrived scenario



CNI - contrived scenario

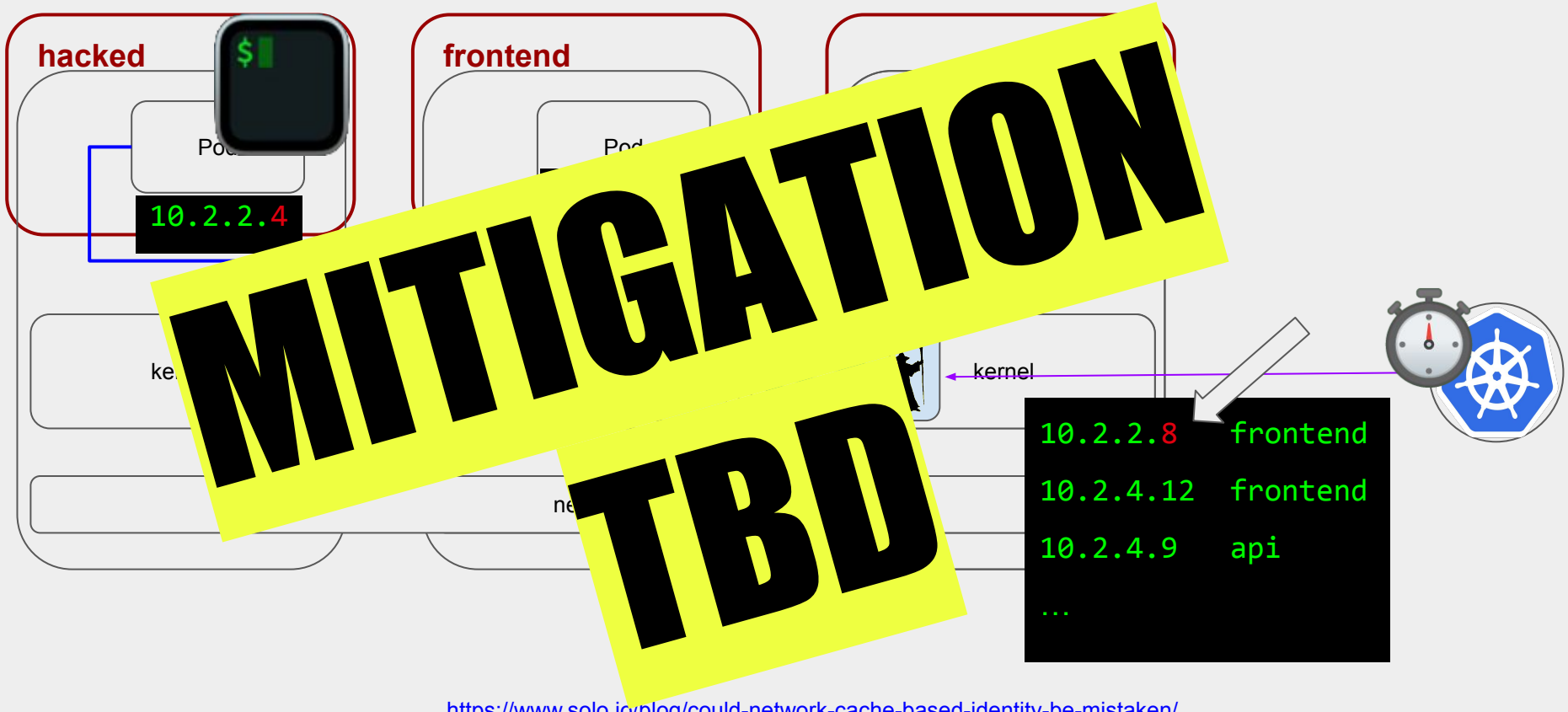


CNI - contrived scenario



https://www.solo.io/blog/could-network-cache-based-identity-be-mistaken/

CNI - contrived scenario



https://www.solo.io/blog/could-network-cache-based-identity-be-mistaken/

Svc Mesh - what is enforceable?

Svc Mesh - what is enforceable?

```
$ kubectl explain ${SERVICE_MESH_AUTH_POLICY}
```

- Allows you to apply policy on traffic which:
  - is going anywhere*
  - is going to a specific K8s Service
  - is going to a specific port(s)

* including loopback or host traffic

Svc Mesh - what is enforceable?

```
$ kubectl explain ${SERVICE_MESH_AUTH_POLICY}
```

- Allows you to apply policy on traffic which:
  - is going anywhere*
  - is going to a specific K8s Service
  - is going to a specific port(s)
  - Allows you to conditionally block/permit requests based on:
    - source IP address or CIDR
    - source kubernetes namespace
    - source kubernetes service account

* including loopback or host traffic

Svc Mesh - what is enforceable?

```
$ kubectl explain ${SERVICE_MESH_AUTH_POLICY}
```

- Allows you to apply policy on traffic which:
  - is going anywhere*
  - is going to a specific K8s Service
  - is going to a specific port(s)
  - Allows you to conditionally block/permit requests based on HTTP properties:
    - specific Host / Authority
    - specific HTTP method
    - specific URI (or prefix)
    - specific header is present or set to a specific value
    - JWT claims (Istio only)

* including loopback or host traffic

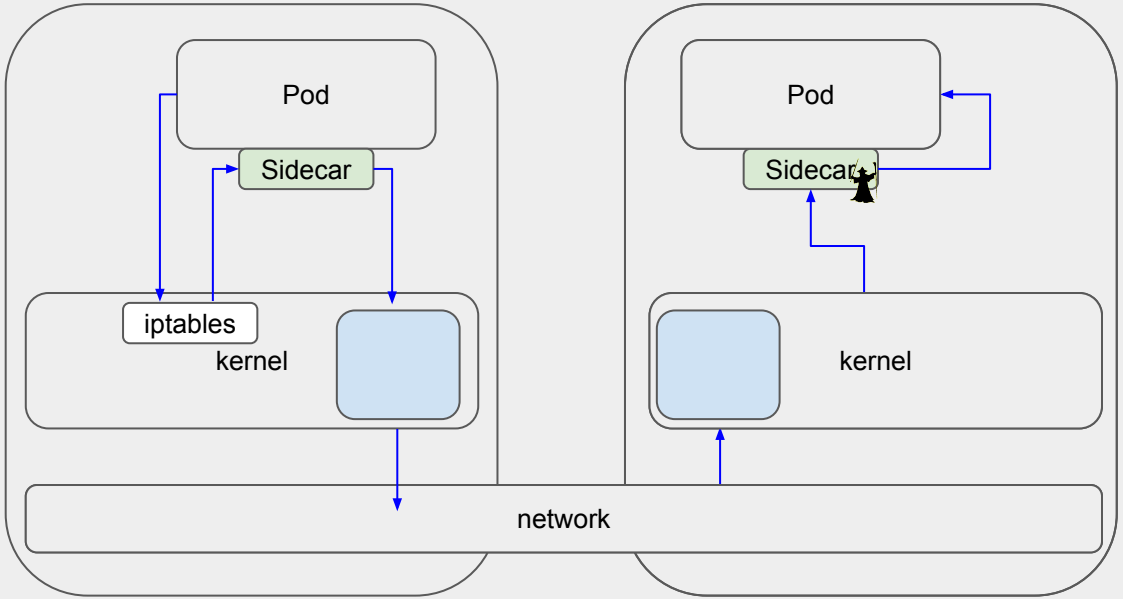
Svc Mesh - what is enforceable?

```
$ kubectl explain ${SERVICE_MESH_AUTH_POLICY}
```

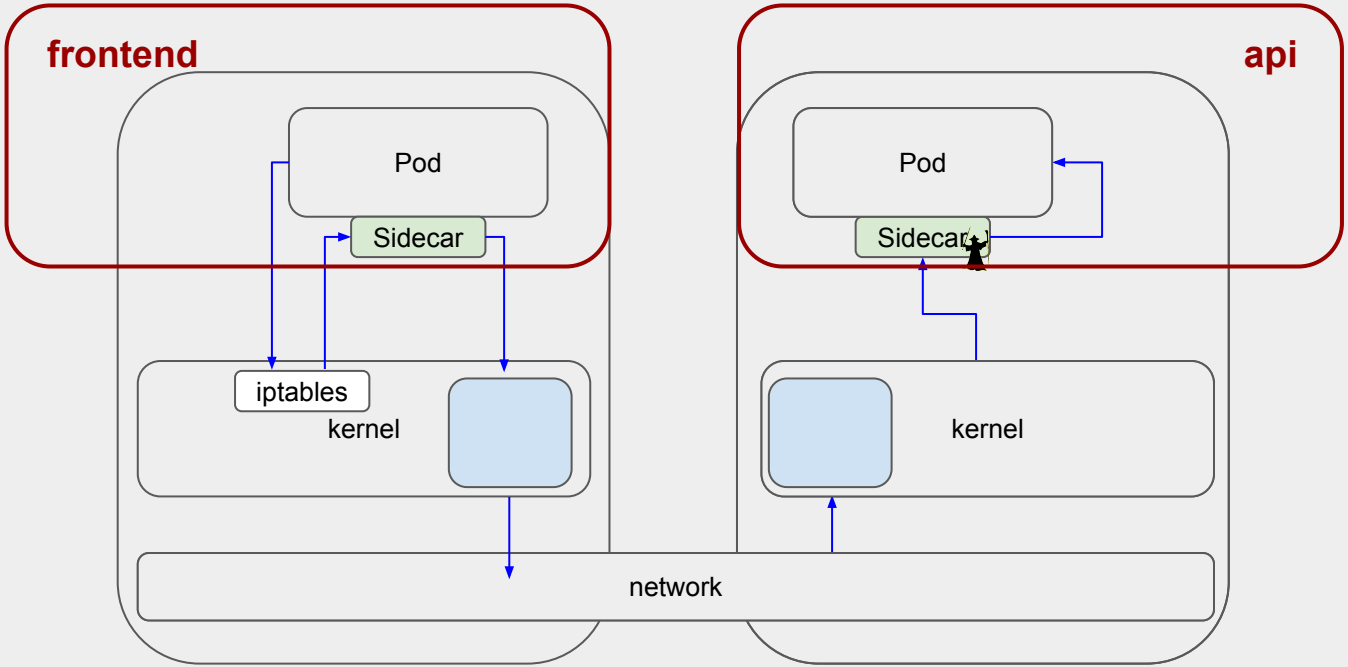
- Allows you to apply policy on traffic which:
  - is going anywhere*
  - is going to a specific K8s Service
  - is going to a specific port(s)
  - Allows you to conditionally block/permit requests based on:
    - source IP address or CIDR
    - source kubernetes namespace
    - source kubernetes service account

* including loopback or host traffic

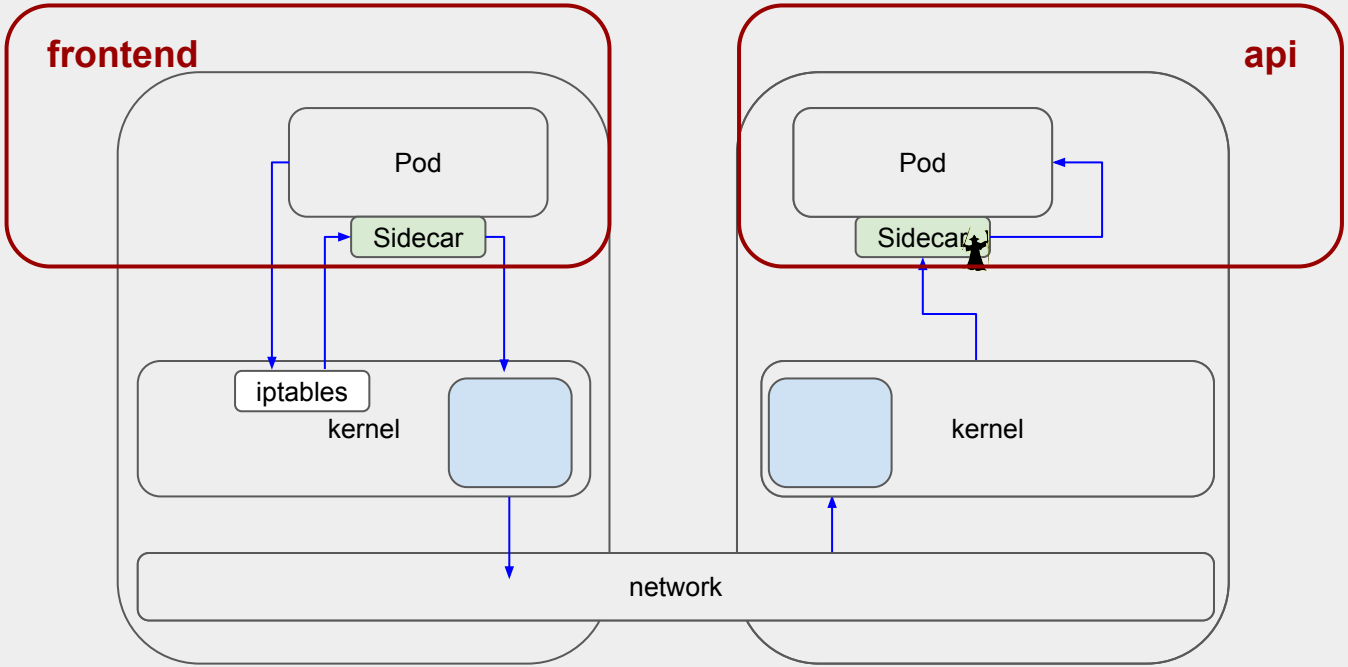
Svc Mesh - how is it enforced?



Svc Mesh - how is it enforced?



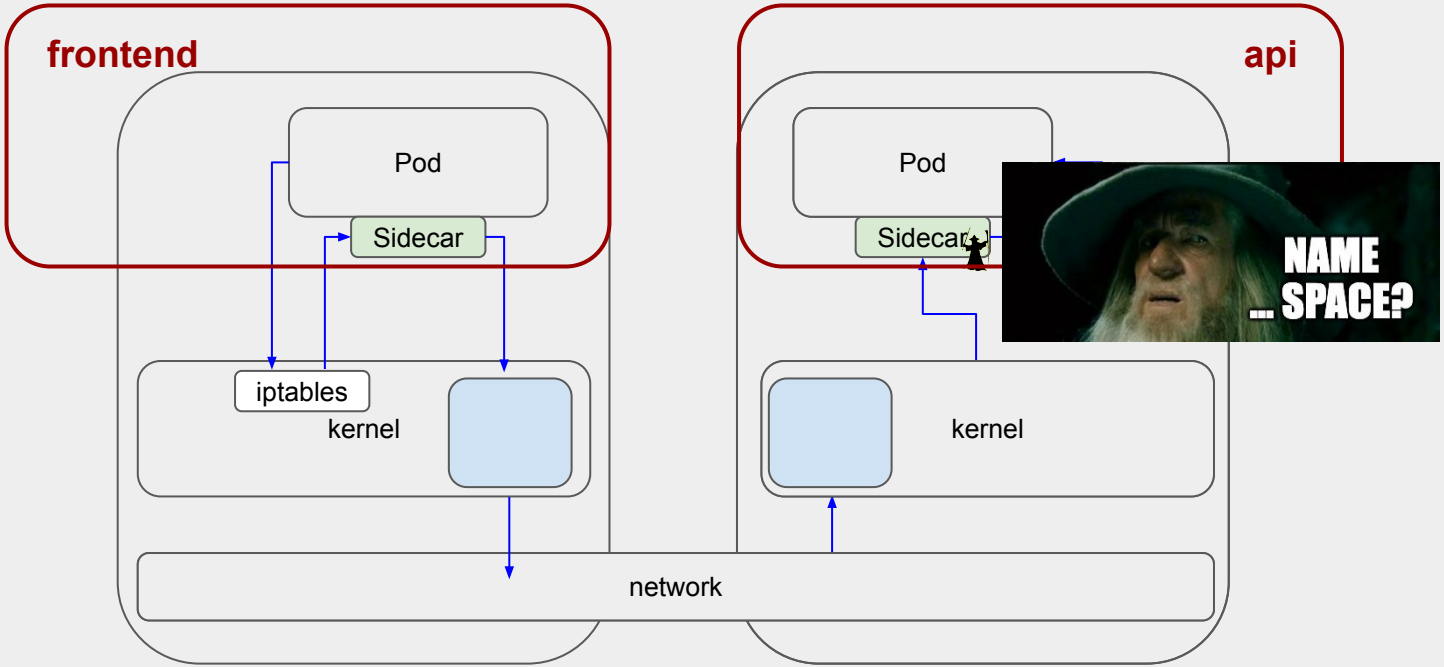
Svc Mesh - how is it enforced?




```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

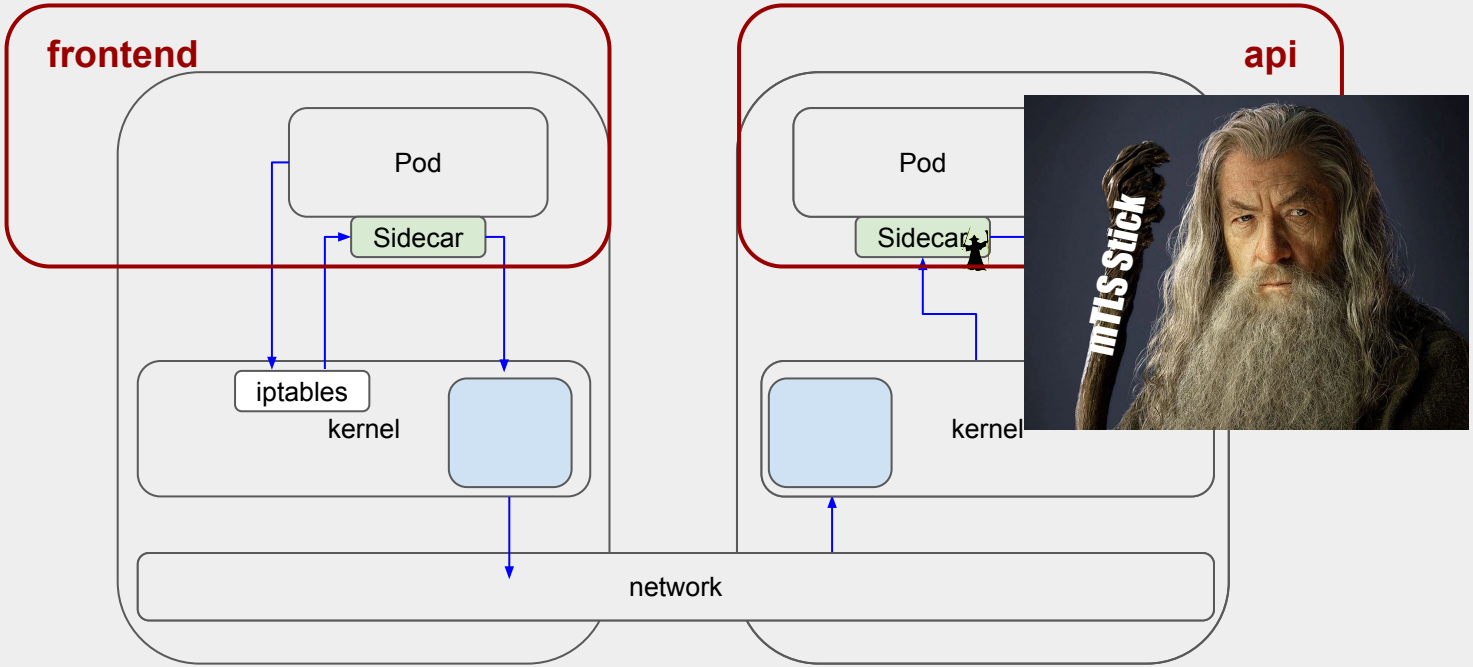
data

Svc Mesh - how is it enforced?




```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

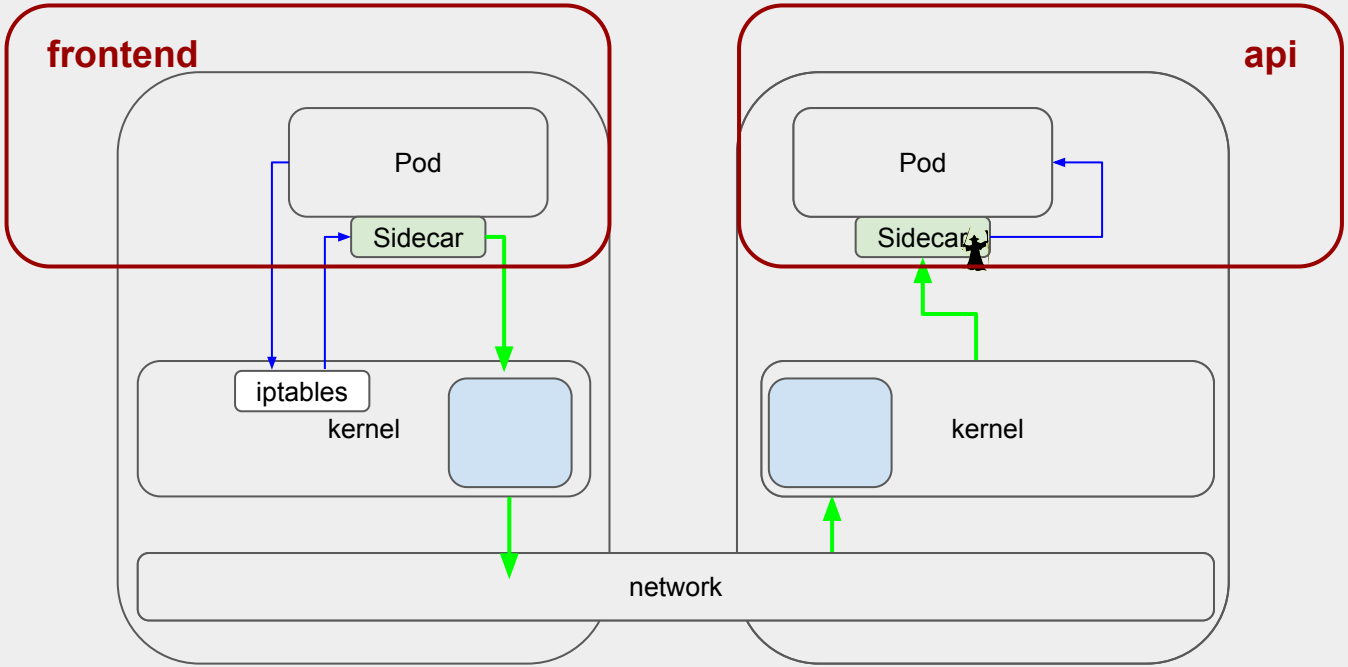
Svc Mesh - how is it enforced?



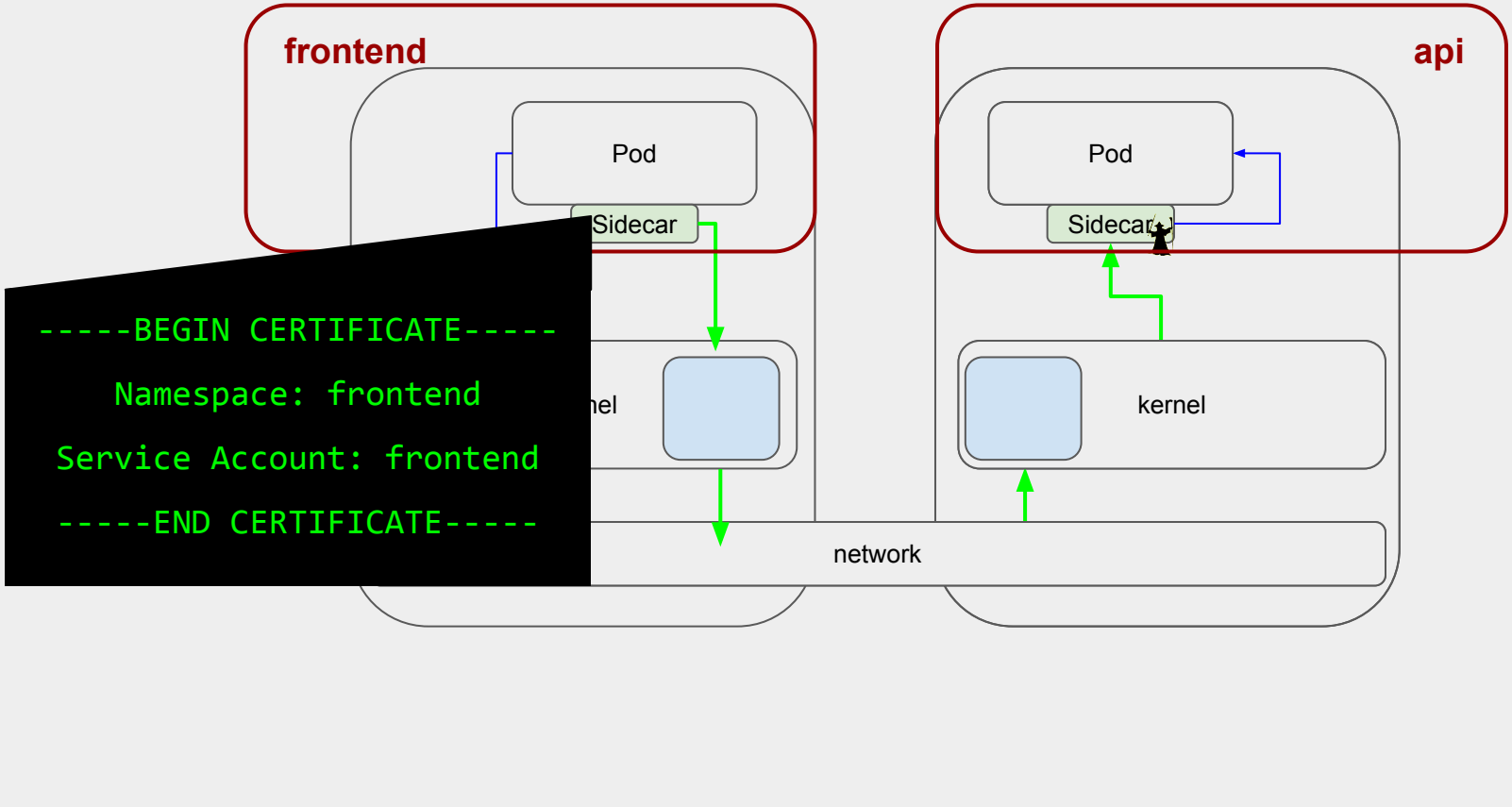

```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

data

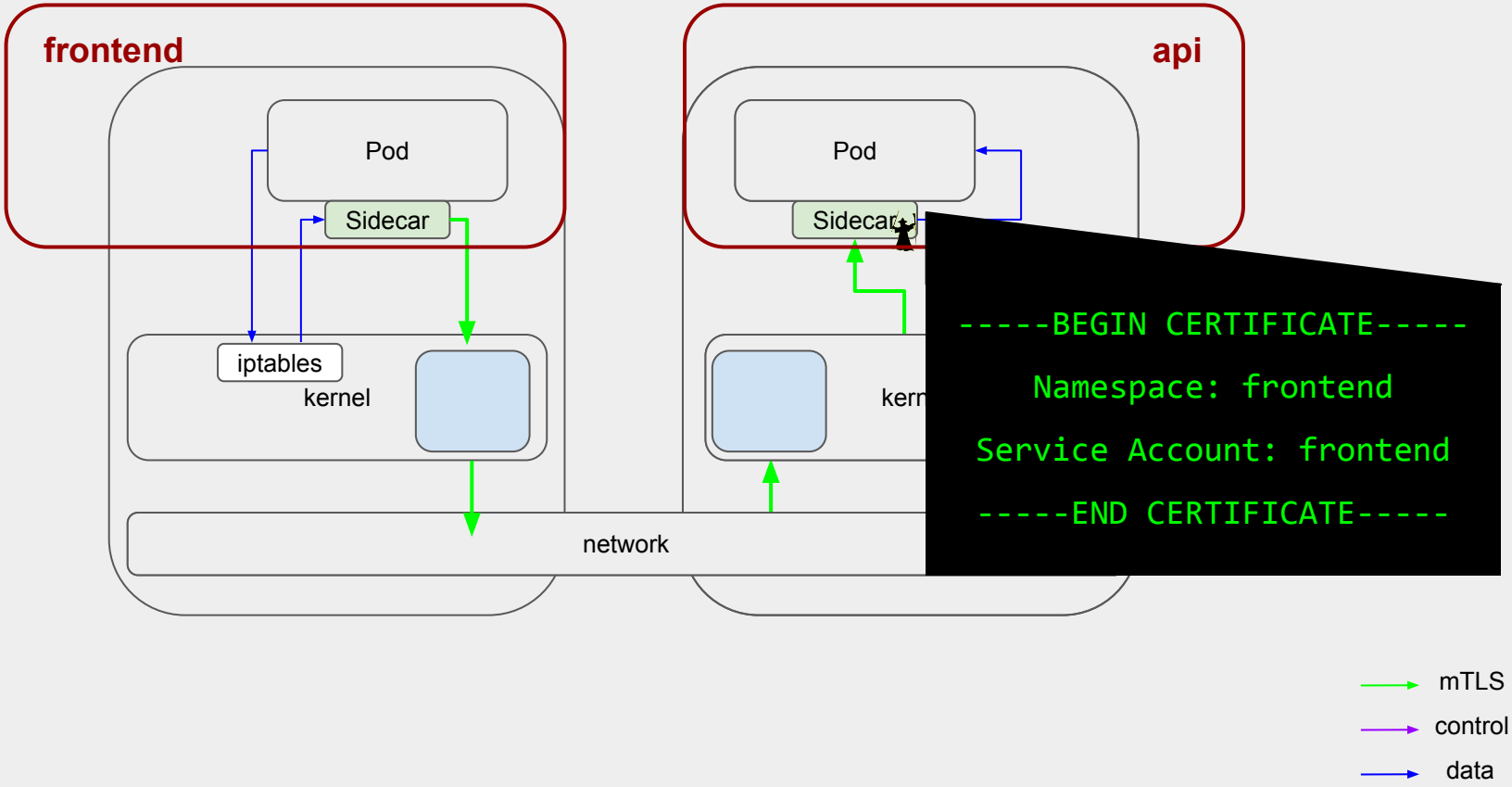
Svc Mesh - how is it enforced?



Svc Mesh - how is it enforced?



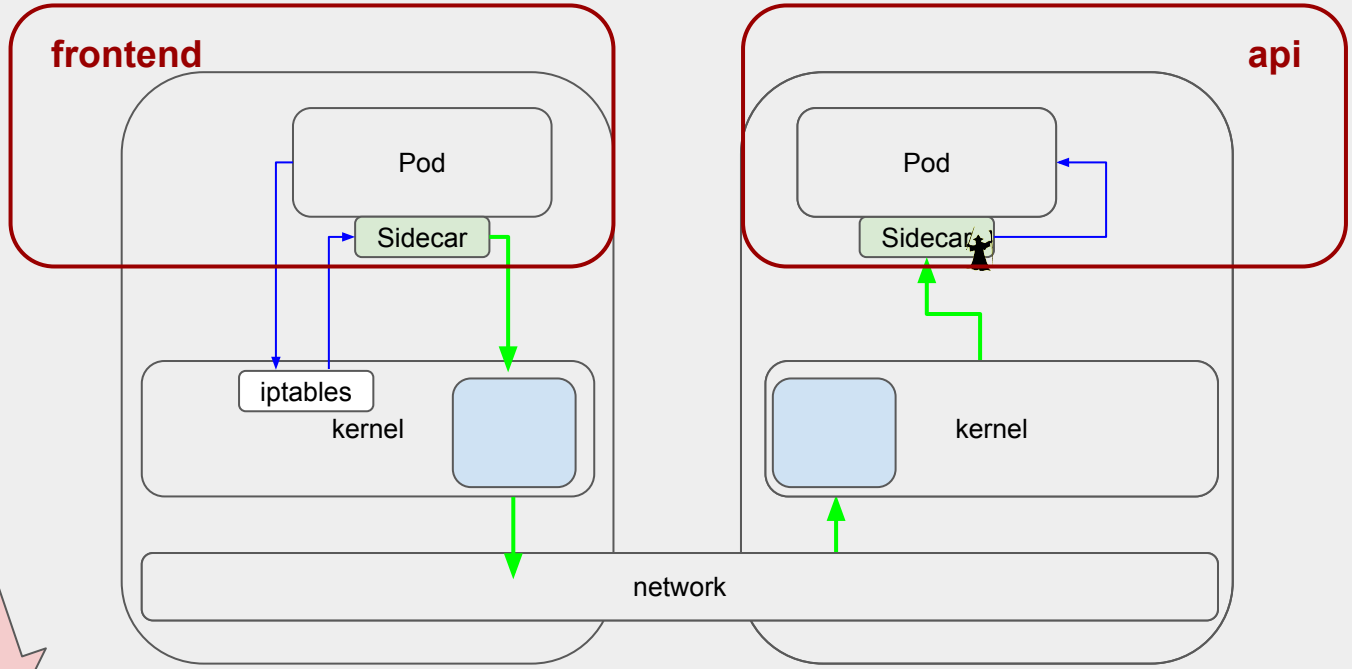
Svc Mesh - how is it enforced?



Svc Mesh - how is it enforced?

Svc Mesh - how is it enforced?

1

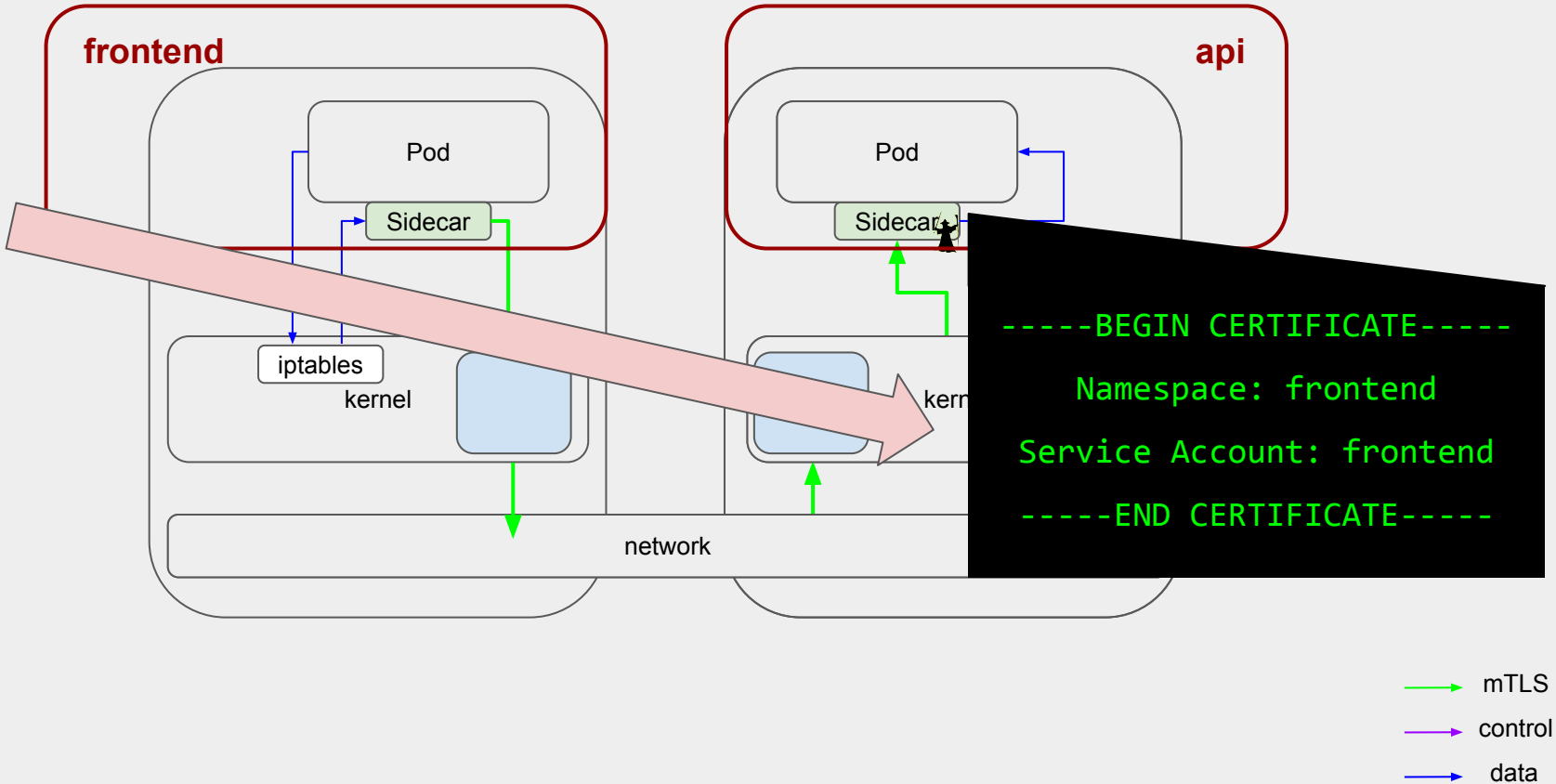



```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

data

Svc Mesh - how is it enforced?

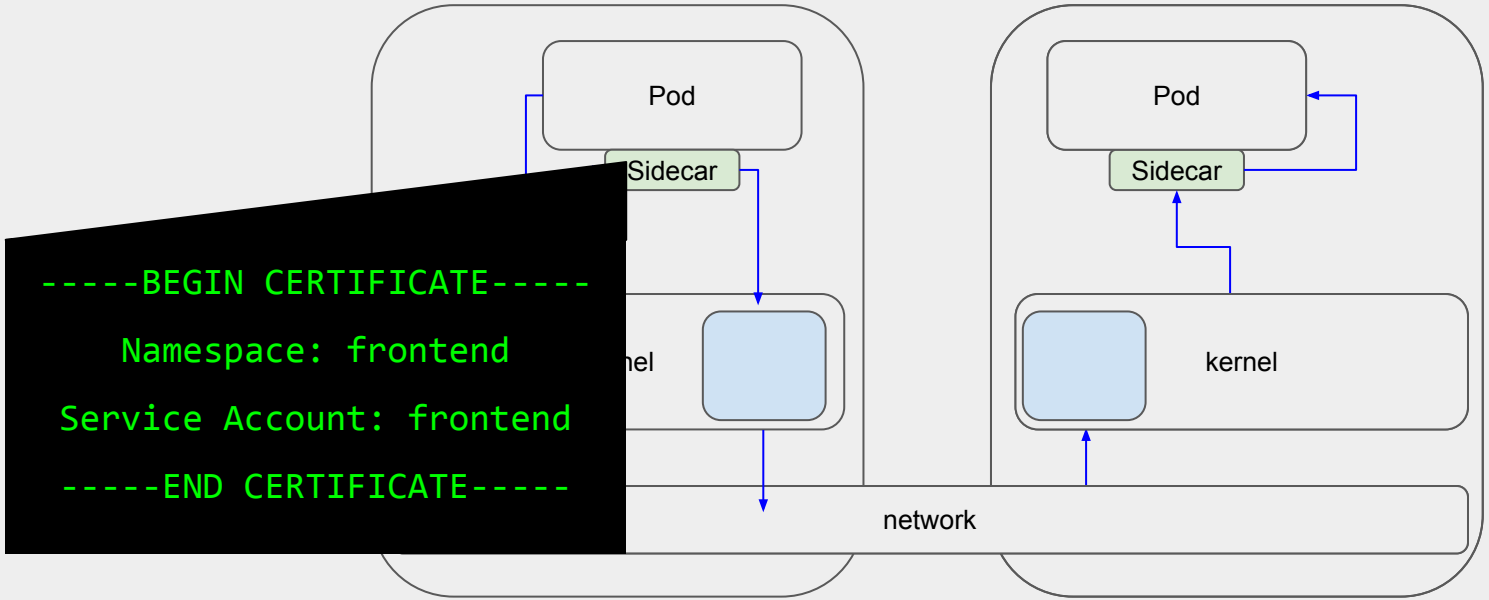
2



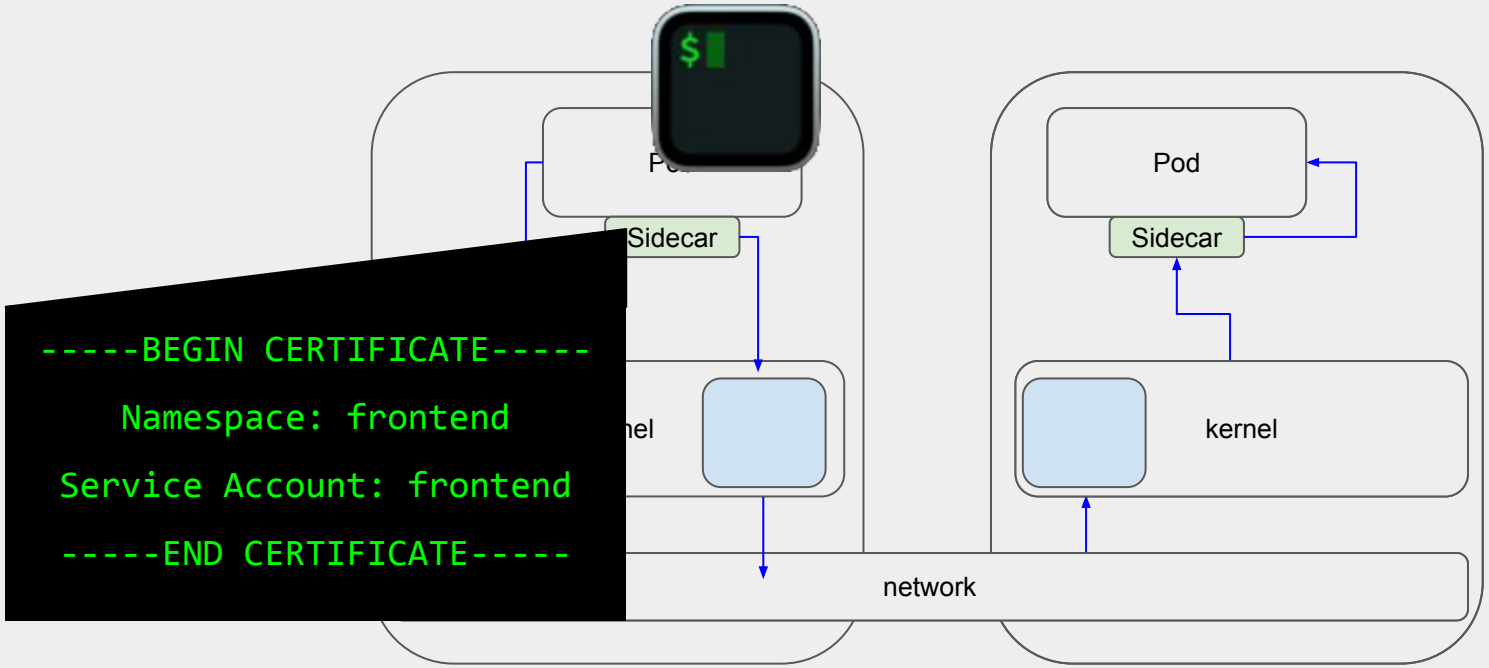
Client Cert - how is it secured?

```
-----BEGIN CERTIFICATE-----
    Namespace: frontend
 Service Account: frontend
 -----END CERTIFICATE-----
```

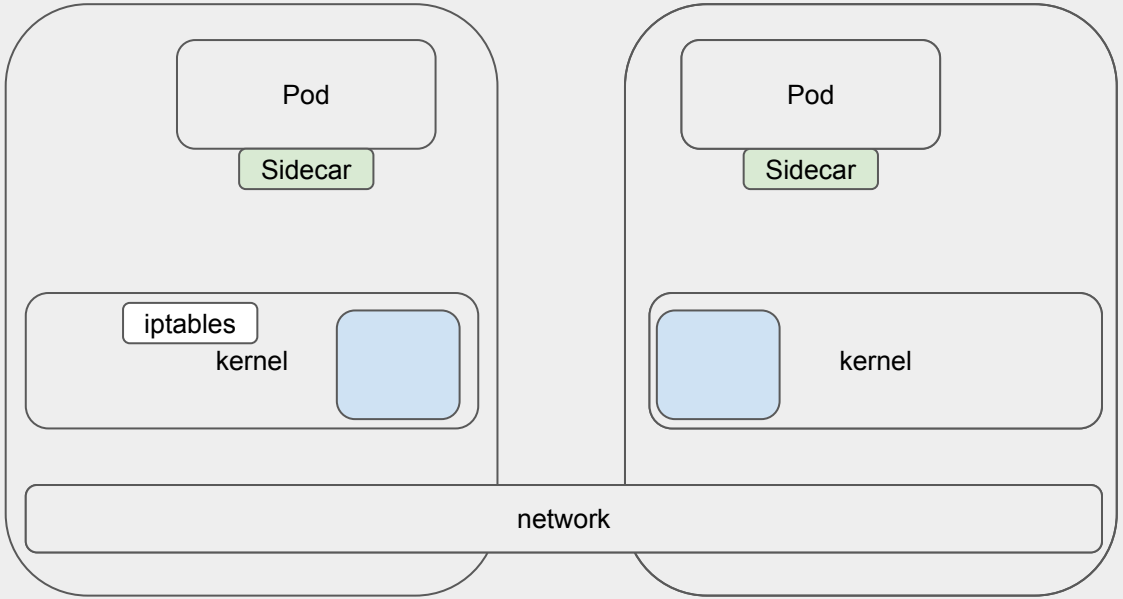
Client Cert - how is it secured?



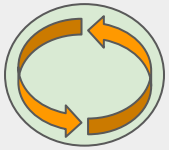
Client Cert - how is it secured?

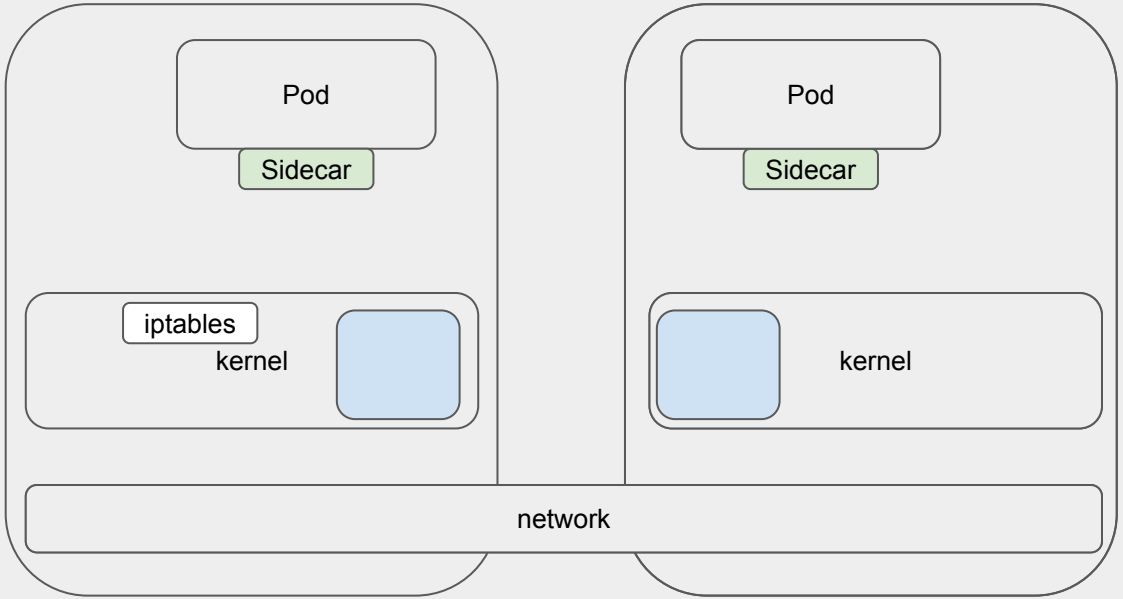


Client Cert - how is it issued?



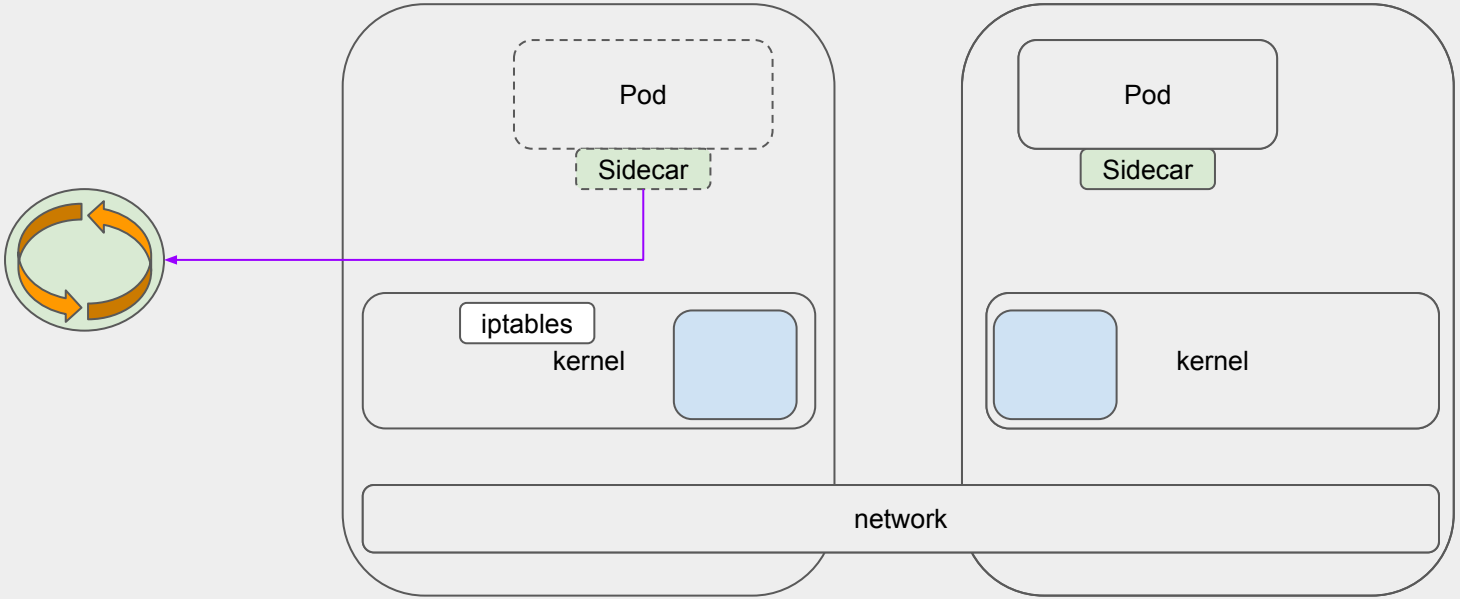
Client Cert - how is it issued?



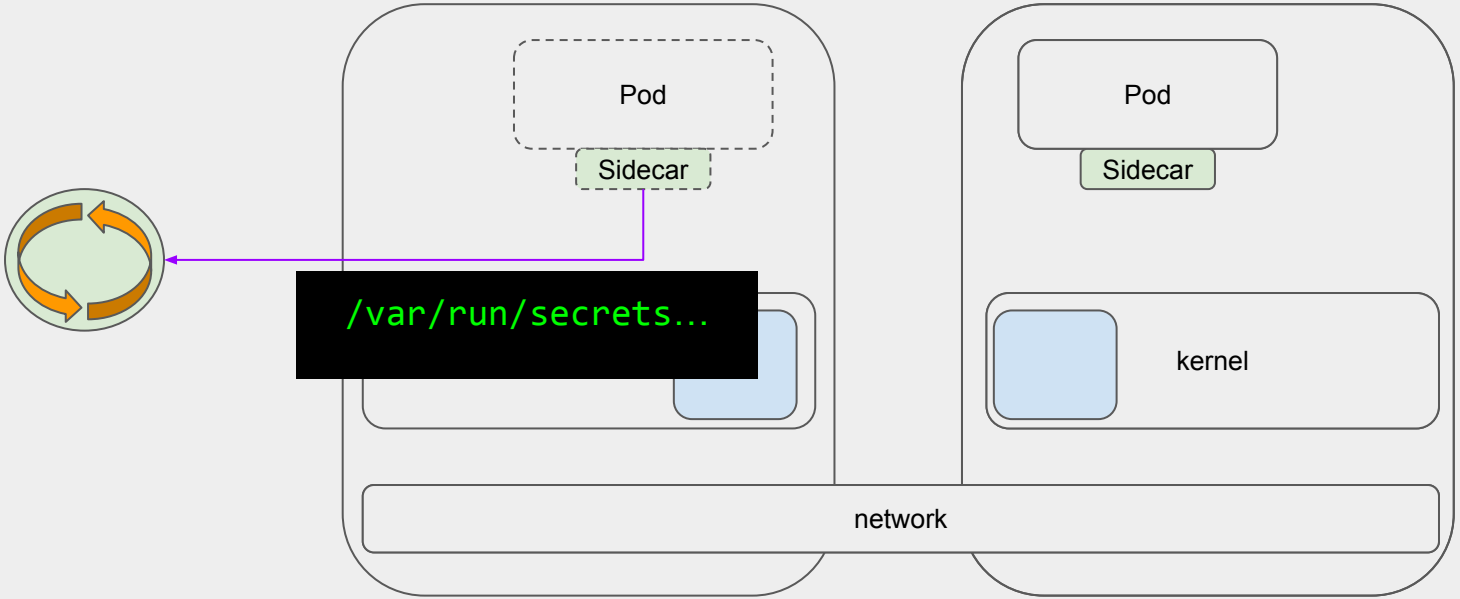


mTLS
control
data

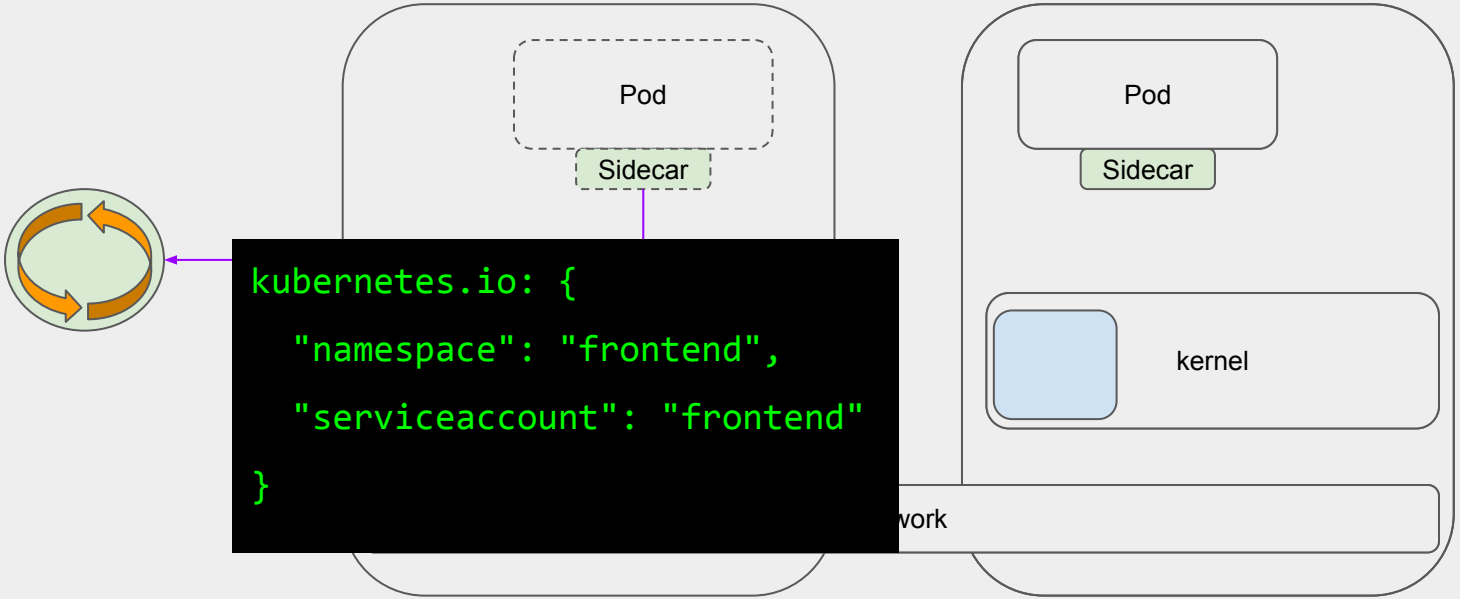
Client Cert - how is it issued?



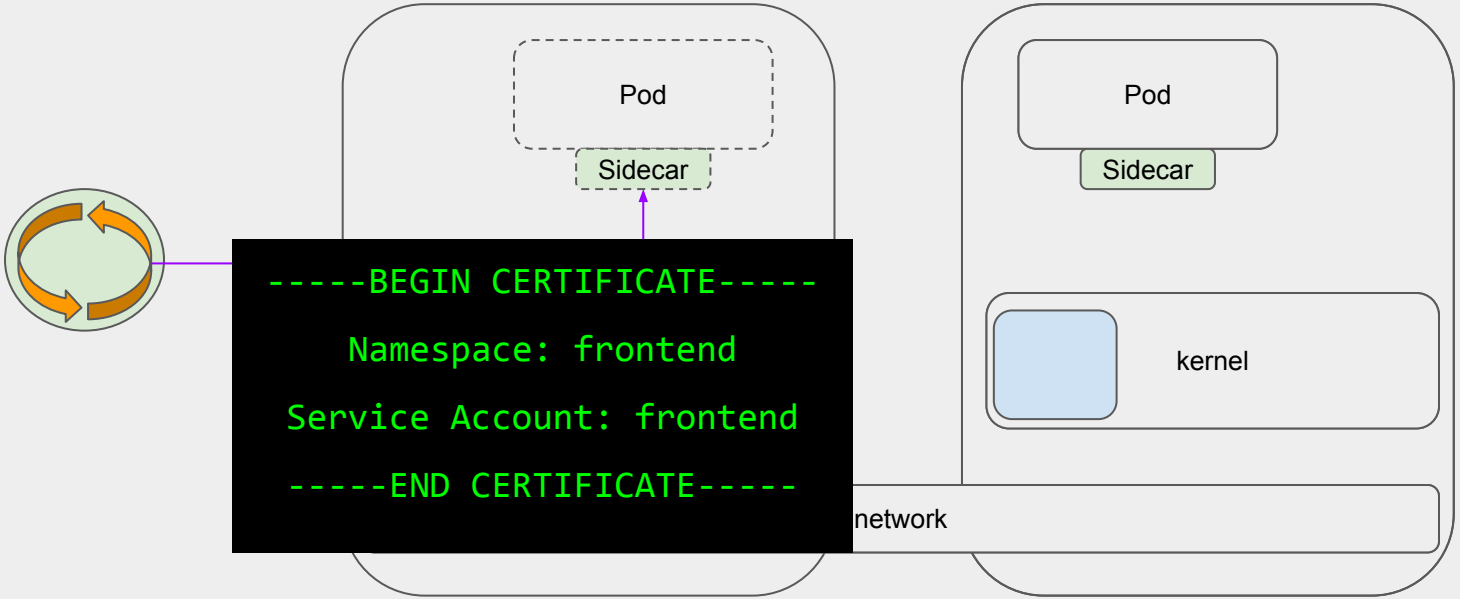
Client Cert - how is it issued?



Client Cert - how is it issued?

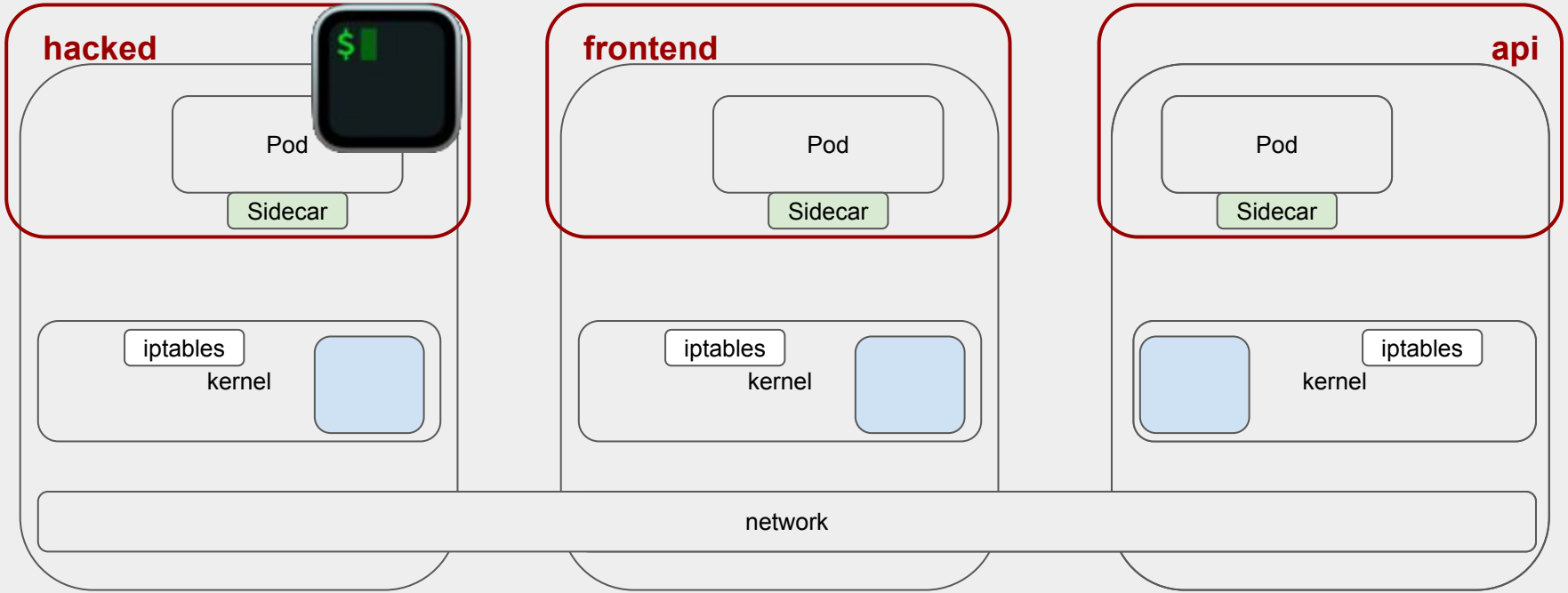


Client Cert - how is it issued?



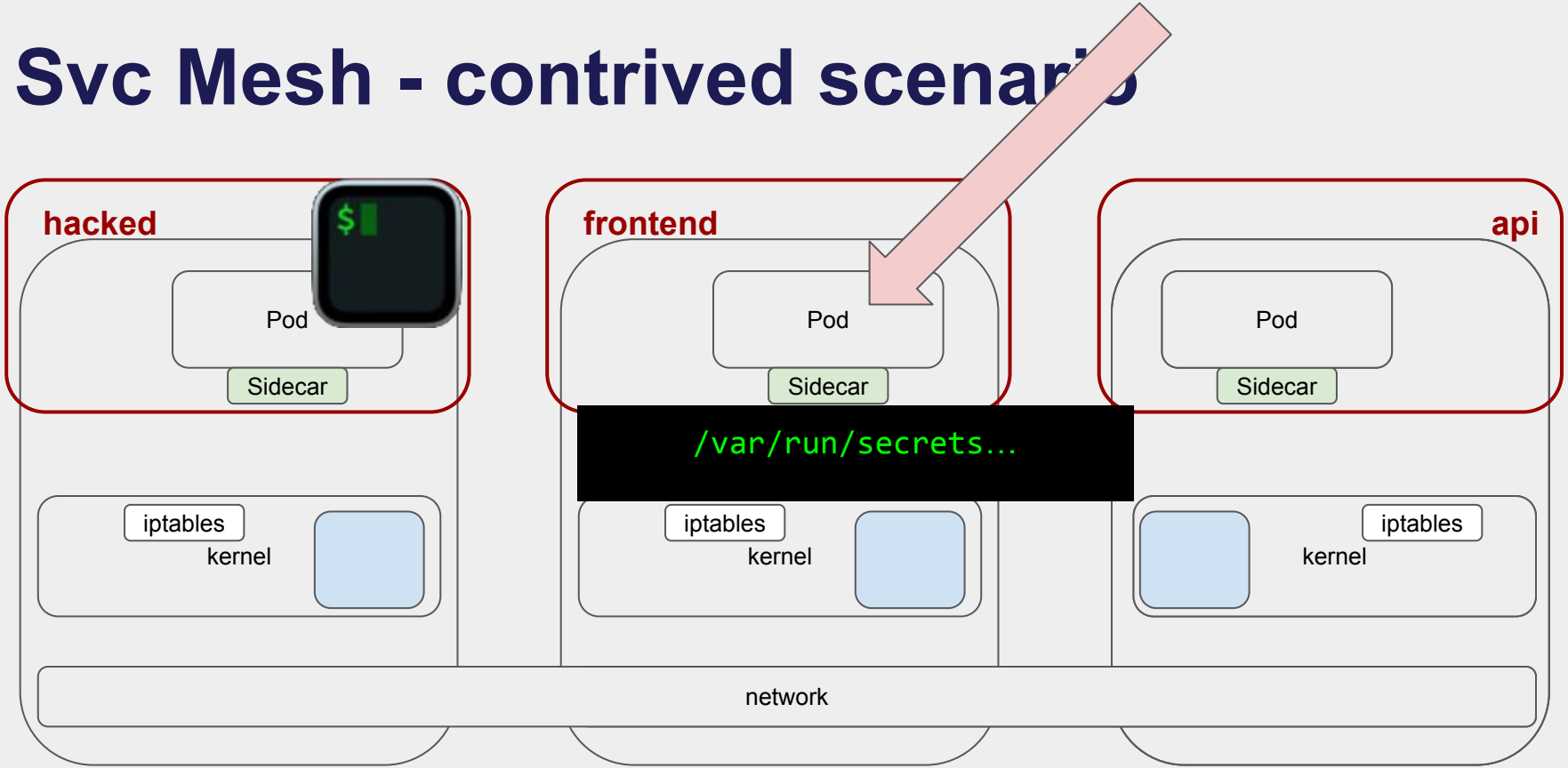
It’s time for ~~a~~ another contrived scenario!

Svc Mesh - contrived scenario



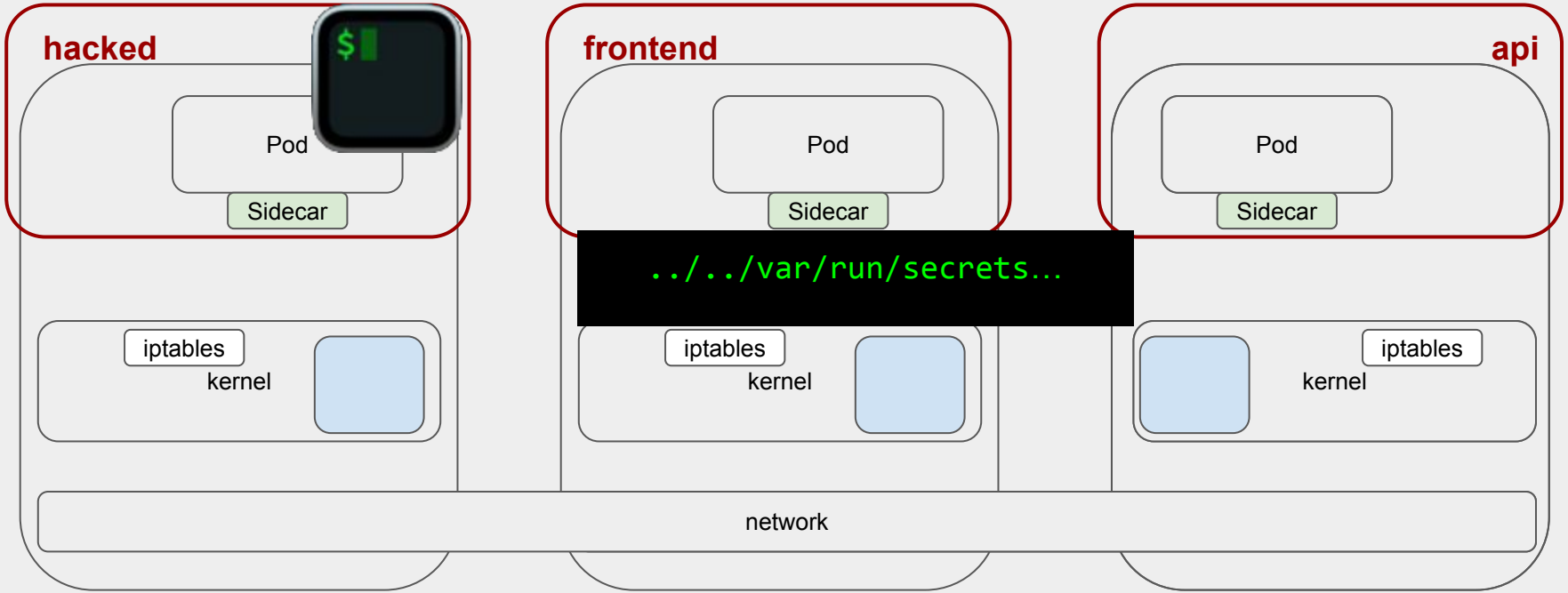

```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

Svc Mesh - contrived scenario



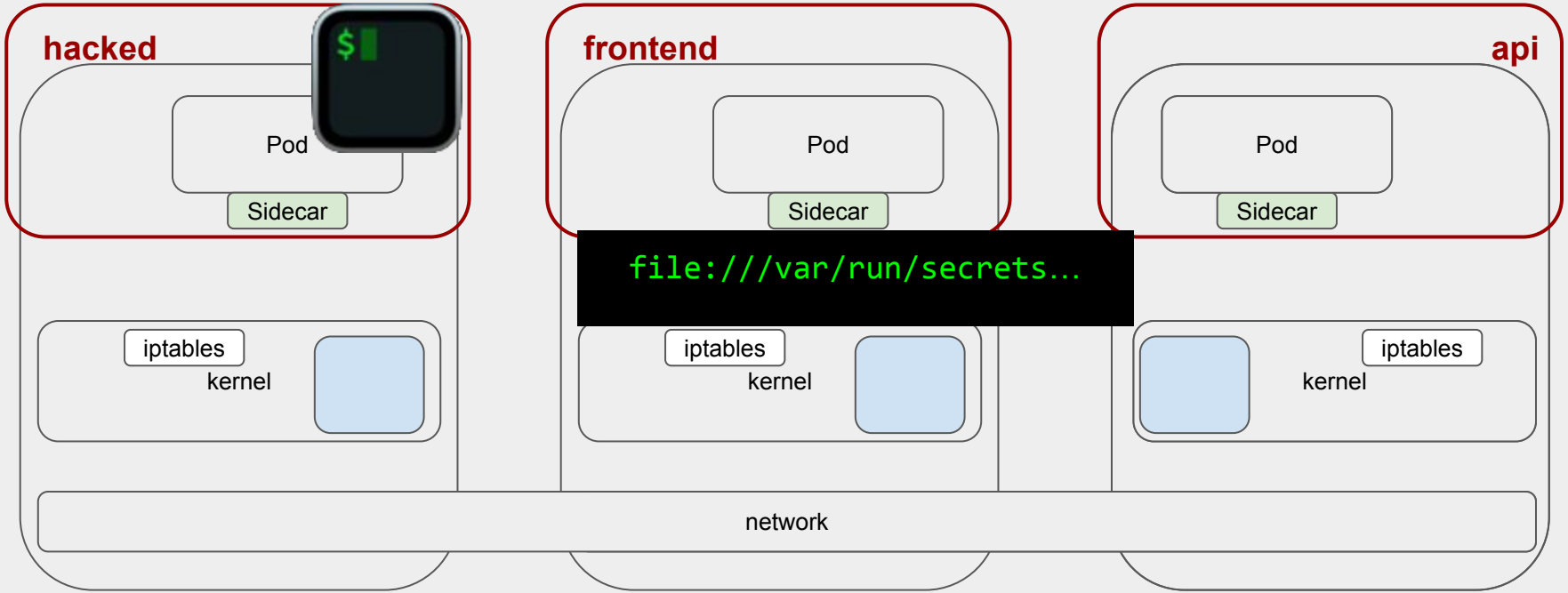

```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

Svc Mesh - contrived scenario



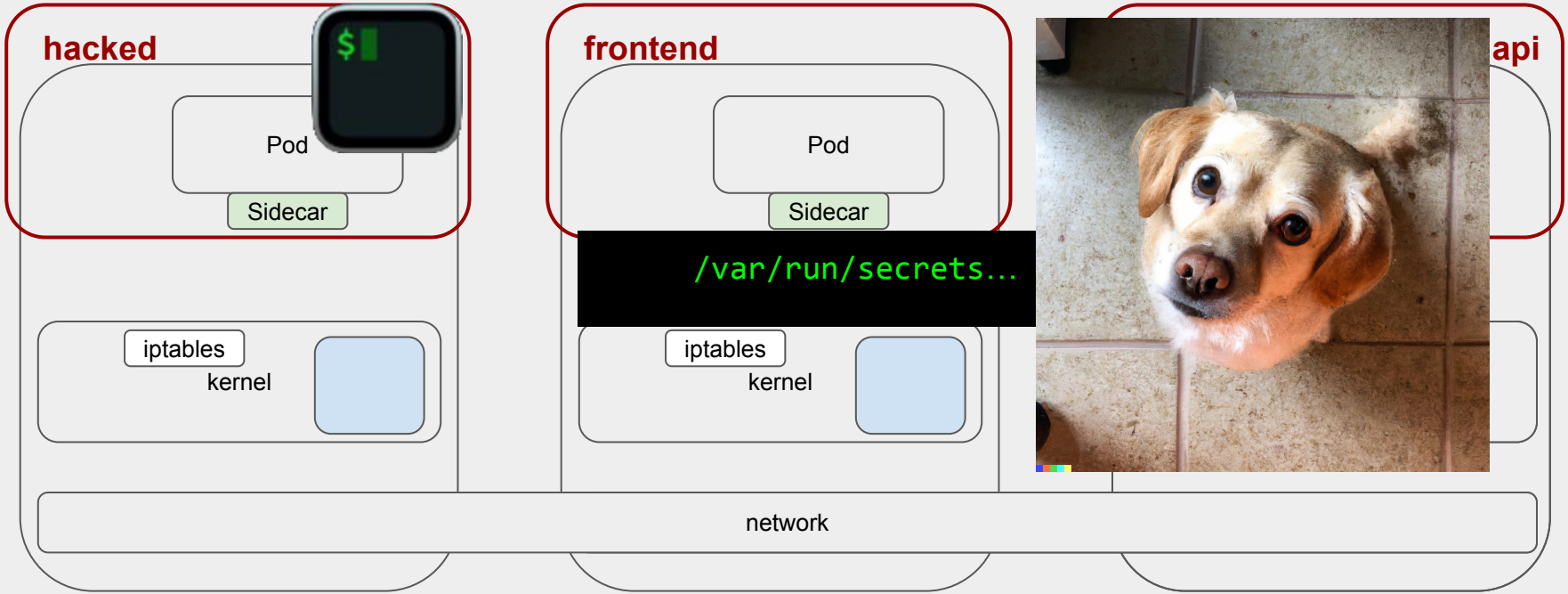

```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

Svc Mesh - contrived scenario



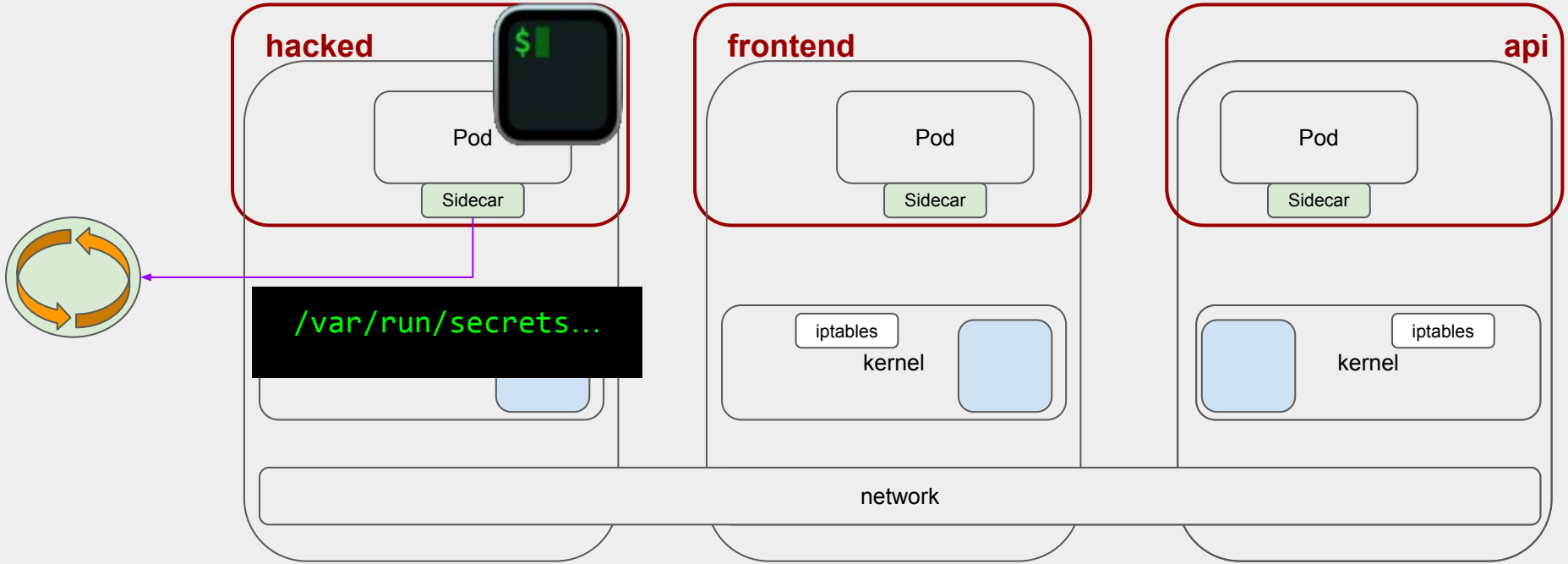

```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

Svc Mesh - contrived scenario



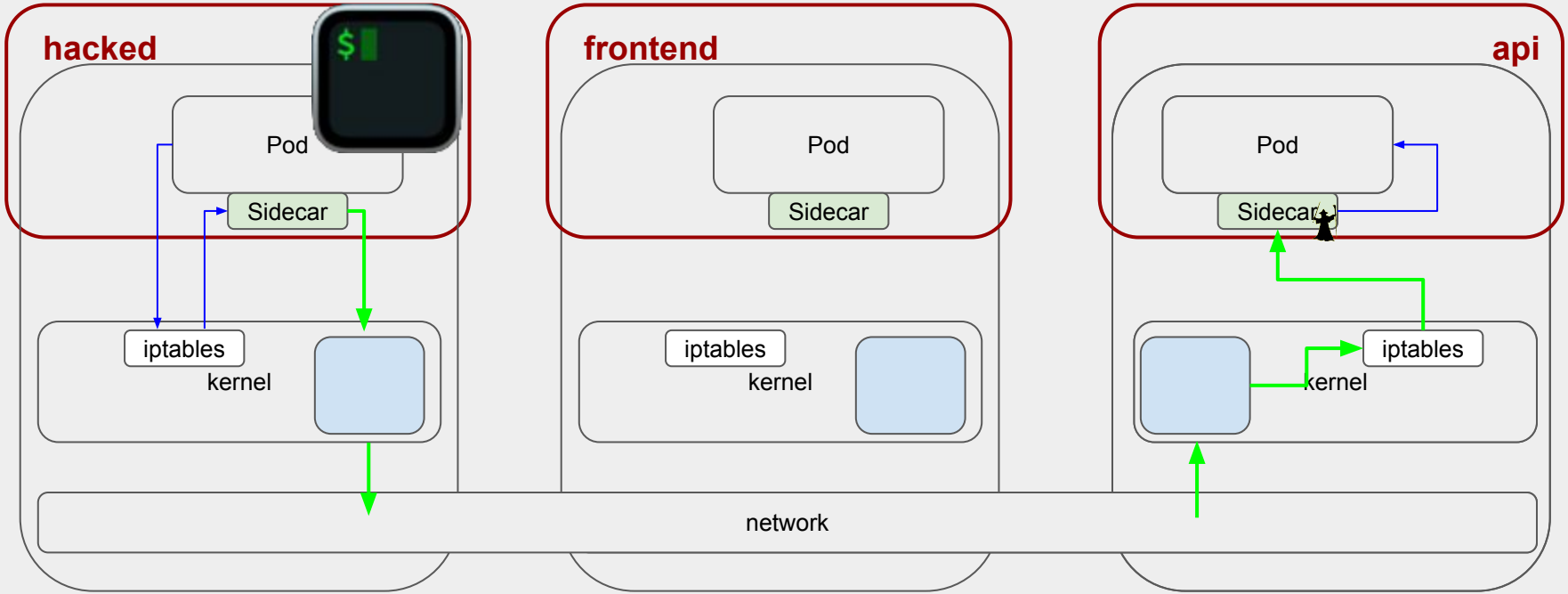

```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

Svc Mesh - contrived scenario



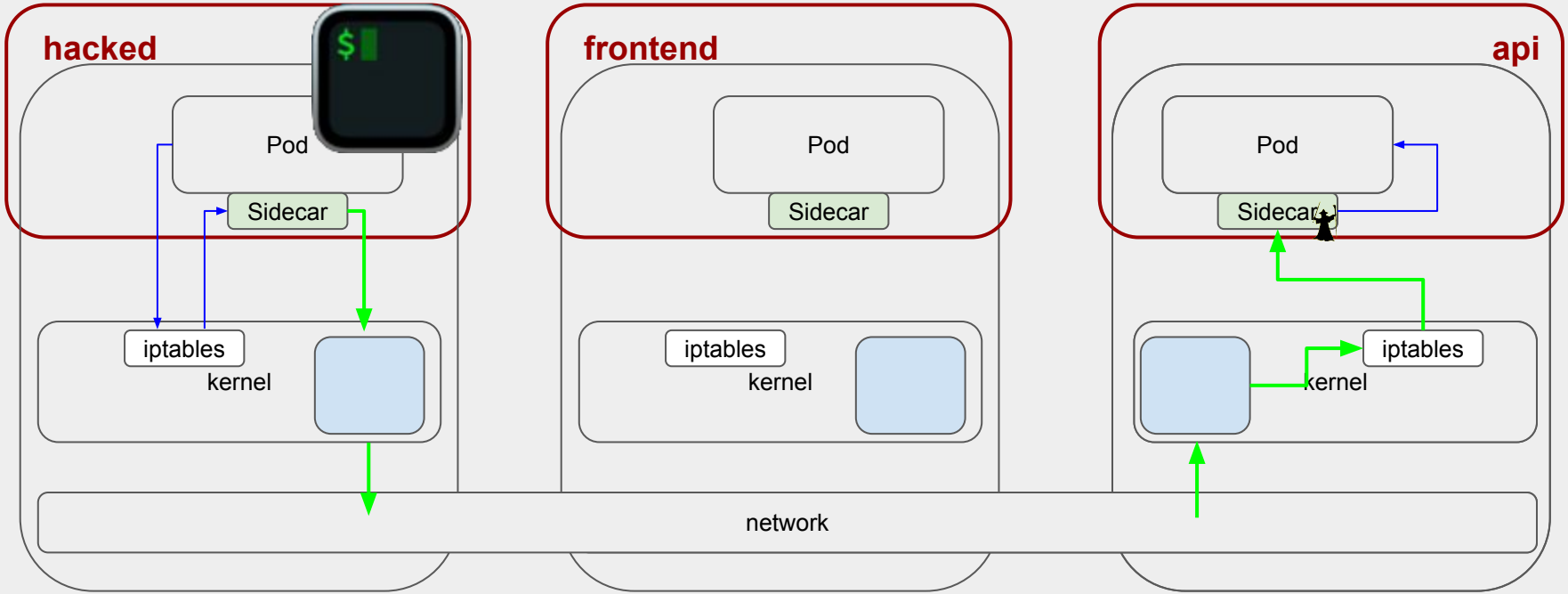

```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

Svc Mesh - contrived scenario




```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

Svc Mesh - contrived scenario

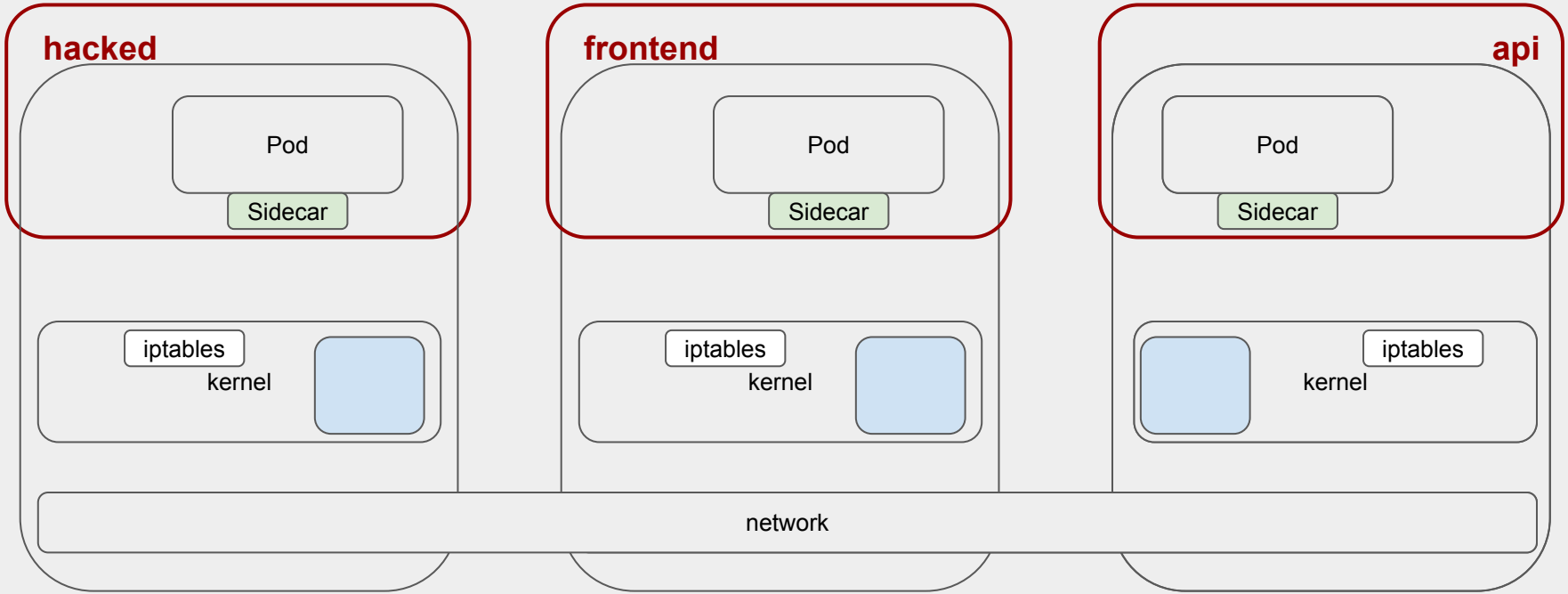



```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

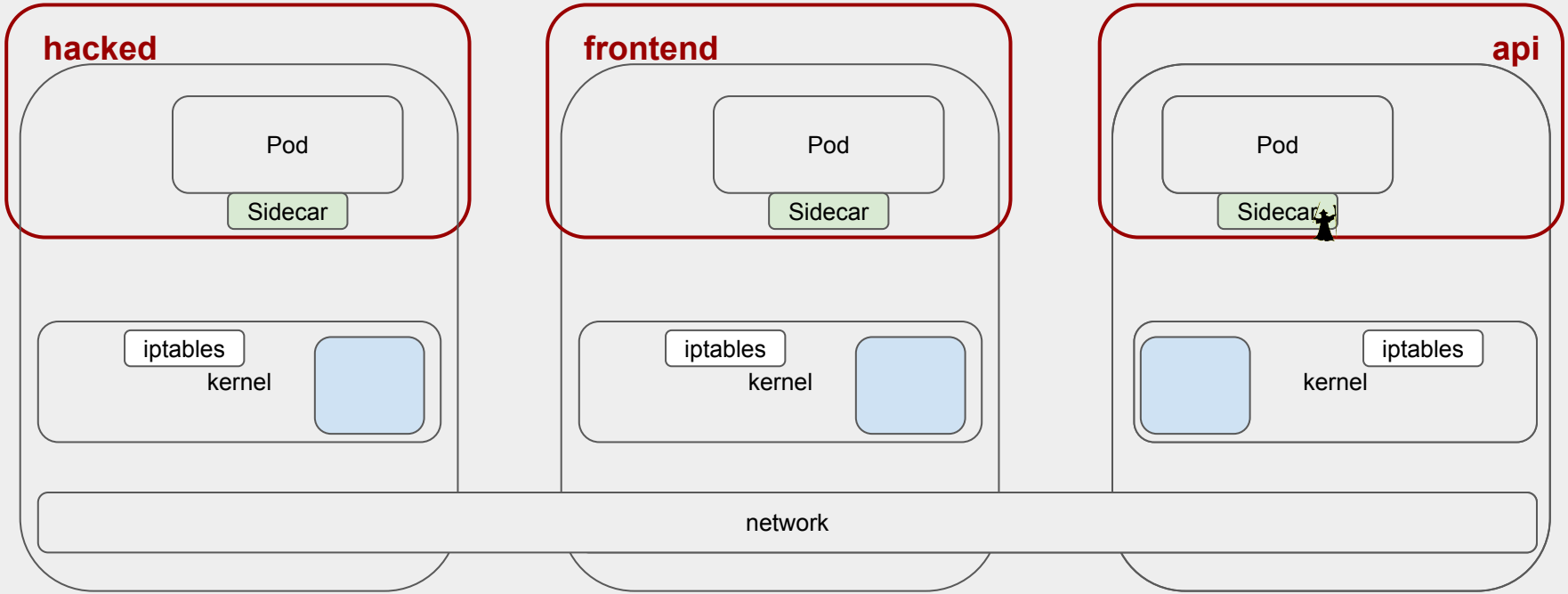
https://github.com/rsalmond/istio-mtls-client-cert-theft

2 mitigations

#1. Right ~~wizard~~ tool, right job.

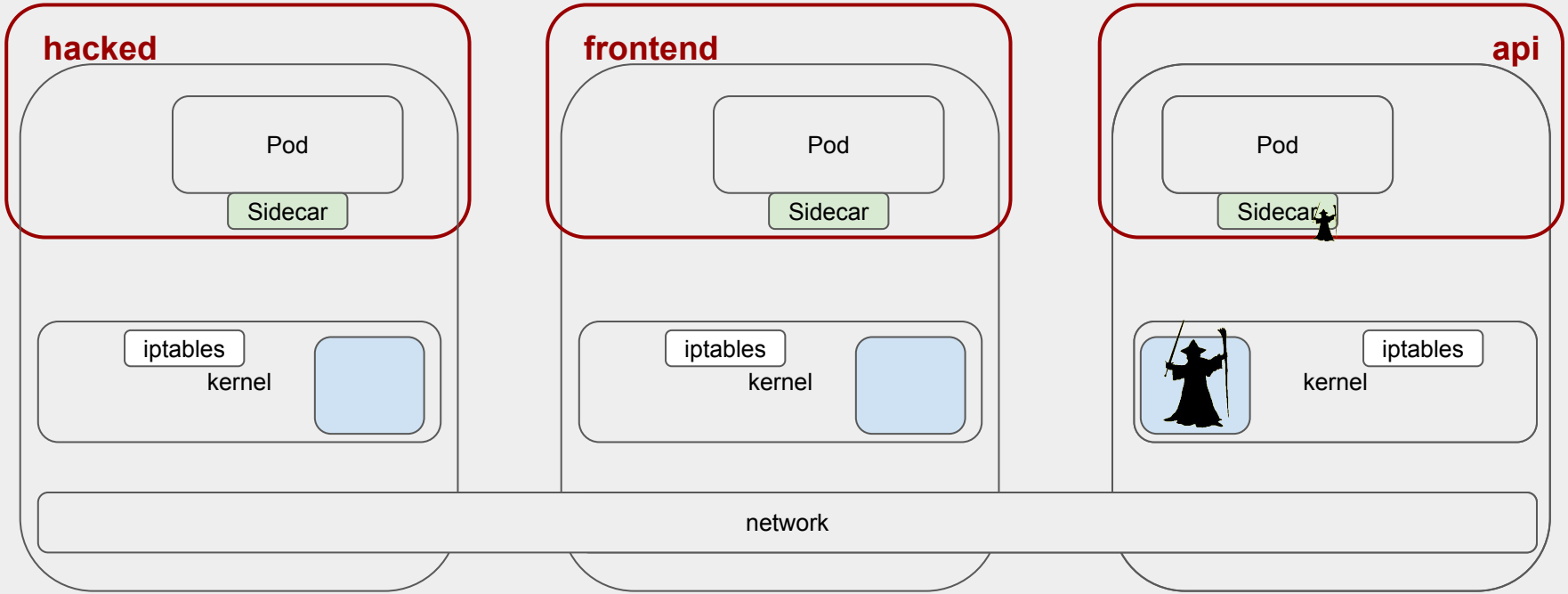


#1. Right ~~wizard~~ tool, right job.



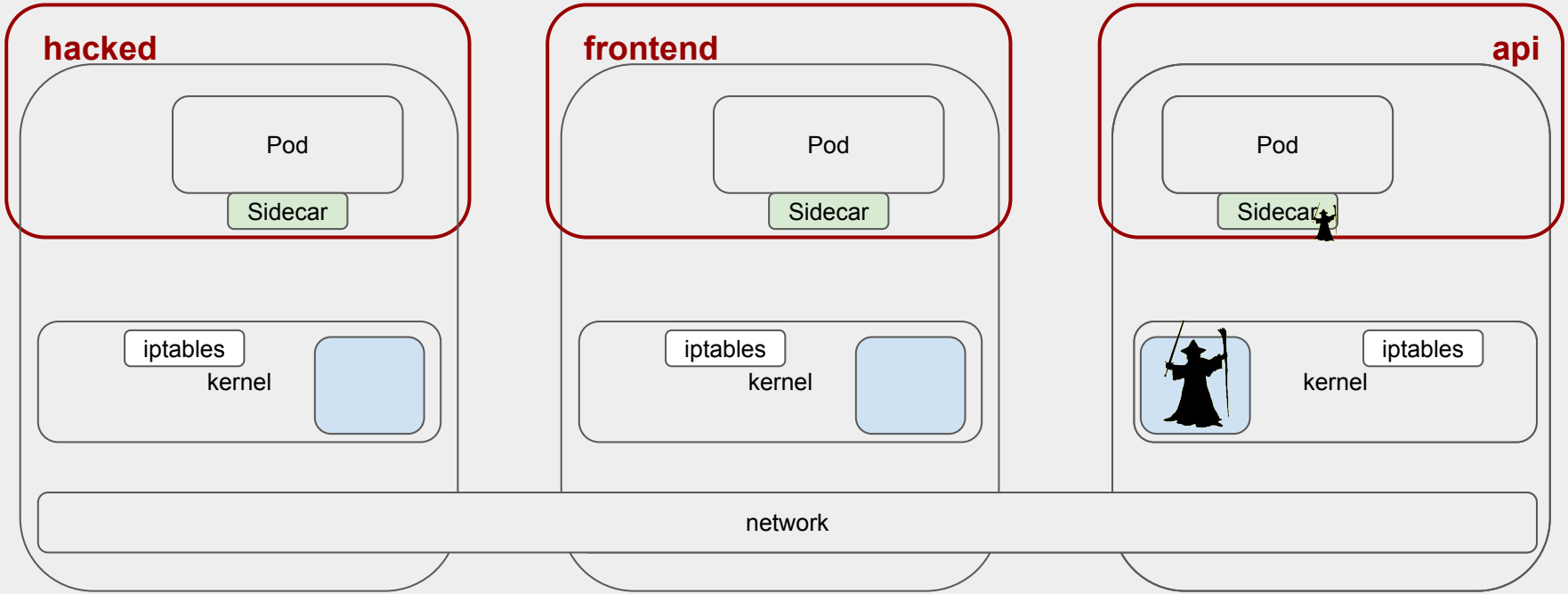

```
$ dear Service Mesh: [pods in frontend] => [pods in api] == OK!
```

#1. Right ~~wizard~~ tool, right job.

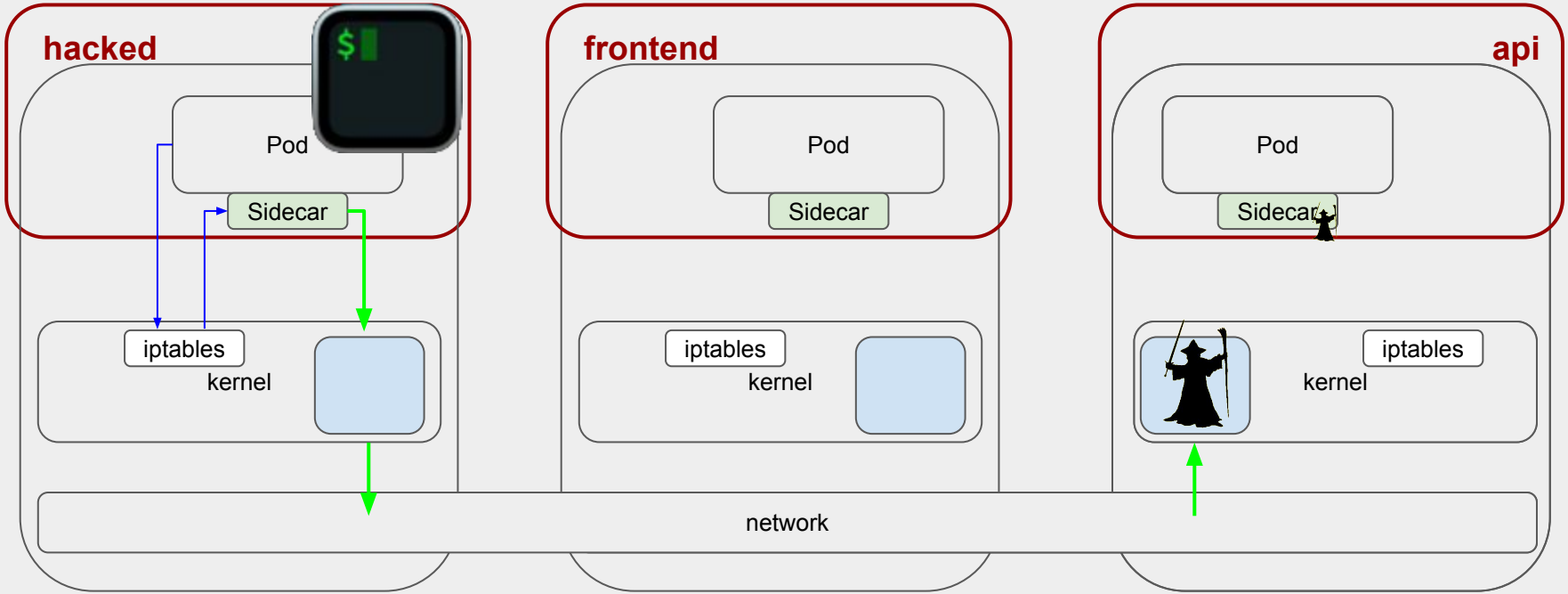



```
$ dear CNI: [pods in frontend] => [pods in api] == OK!
```

#1. Right ~~wizard~~ tool, right job.

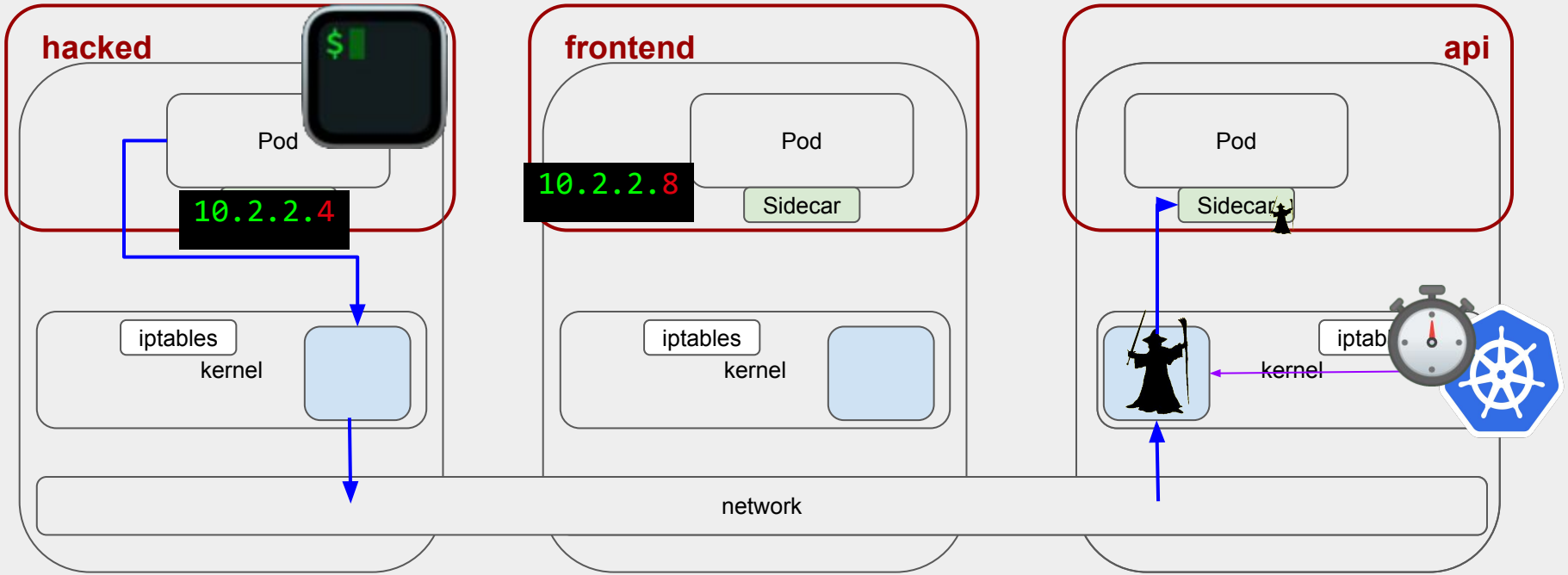


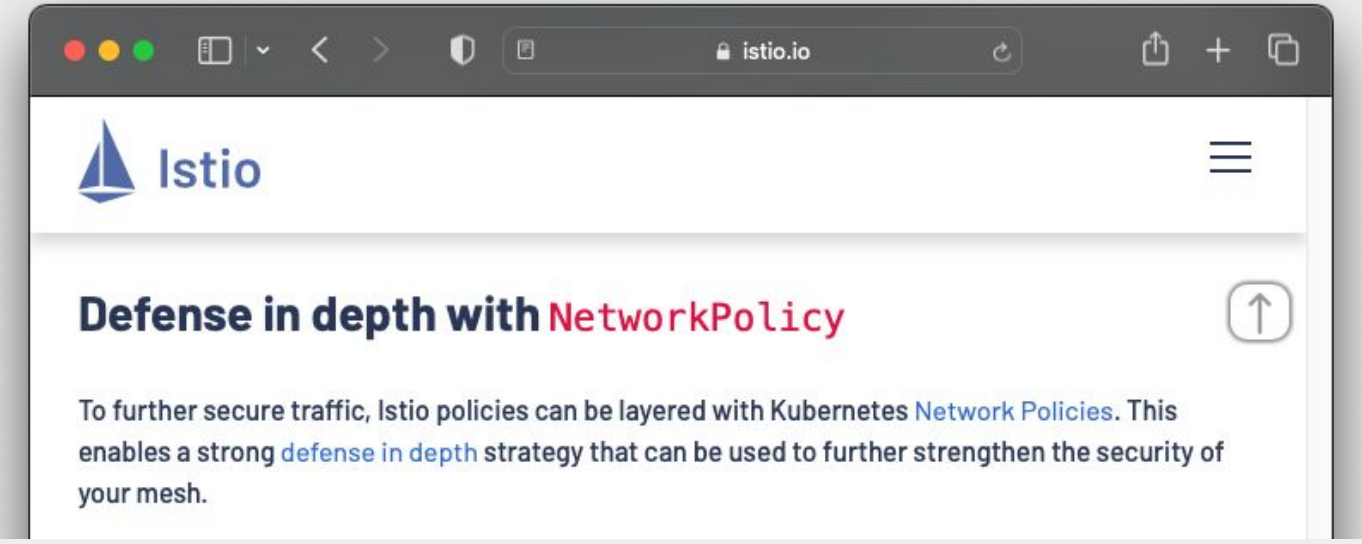
#1. Right ~~wizard~~ tool, right job.




```
$ dear CNI: [pods in frontend] => [pods in api] == OK!
```

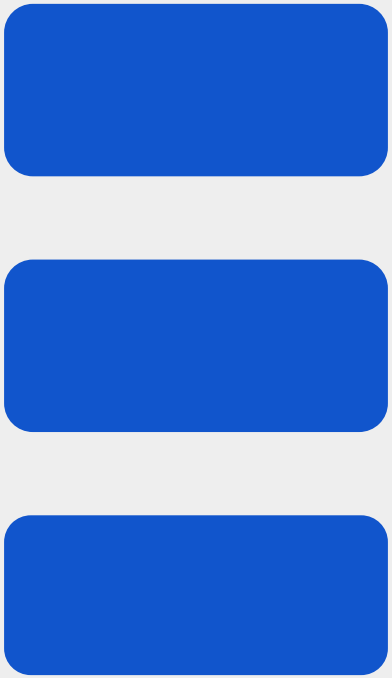
#1. Right ~~wizard~~ tool, right job.



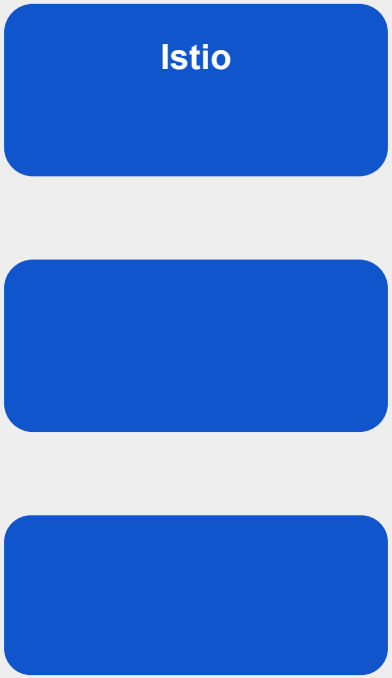

istio.io
Istio
Defense in depth with NetworkPolicy
To further secure traffic, Istio policies can be layered with Kubernetes Network Policies. This enables a strong defense in depth strategy that can be used to further strengthen the security of your mesh.

#2. Evolution.

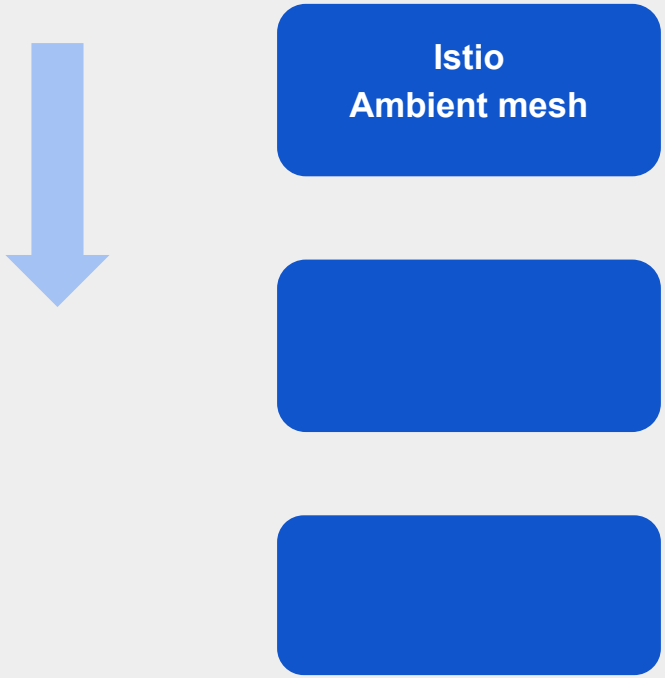
The landscape is changing 🔮



The landscape is changing 🔮



The landscape is changing 🔮



Ambient Mesh

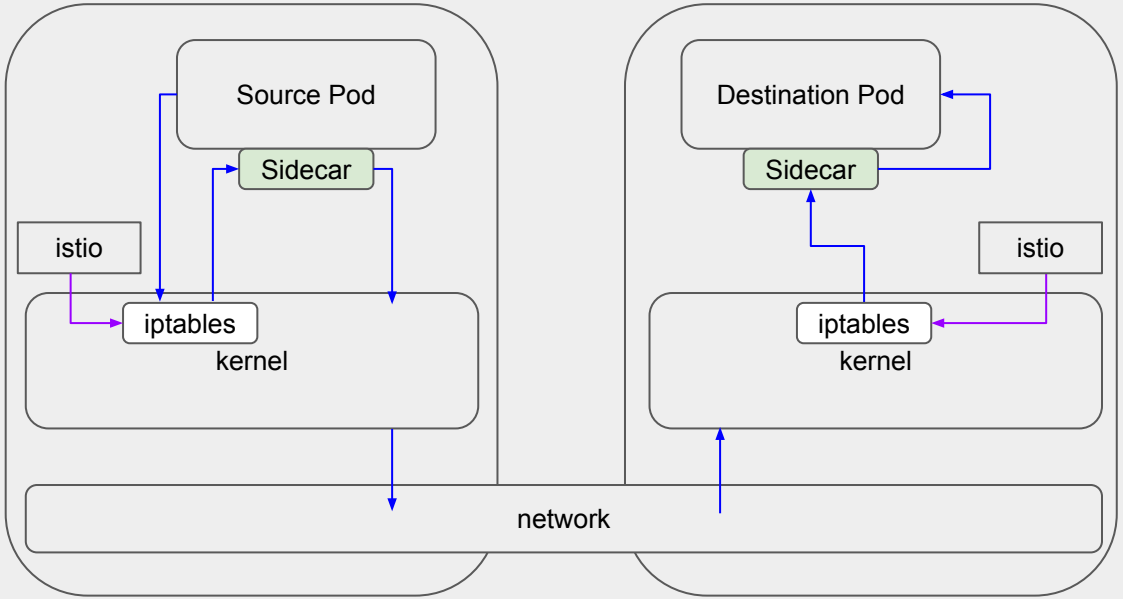
Ambient Mesh

L4 Secure Overlay

ztunnel

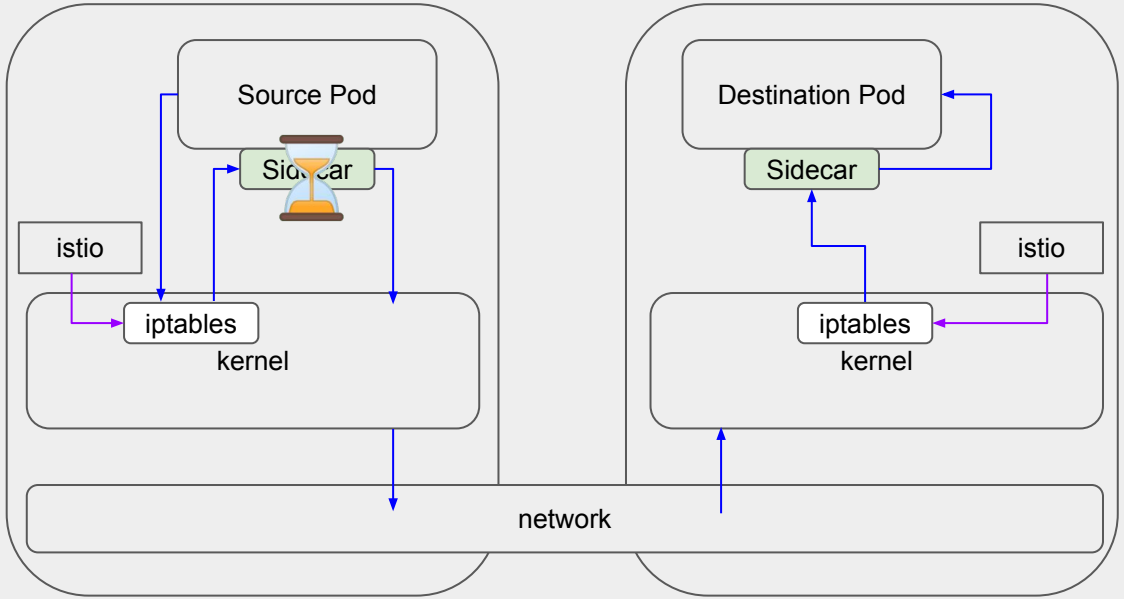
- mTLS,
- Svc-to-svc Authz Policies

Ambient Mesh



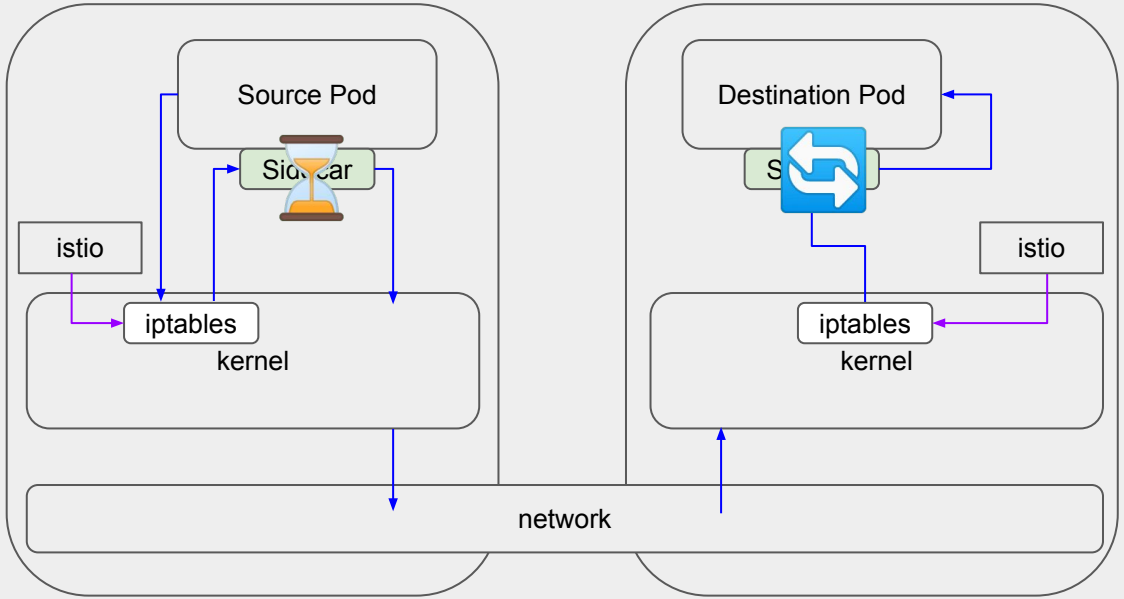
mTLS
control
data

Ambient Mesh



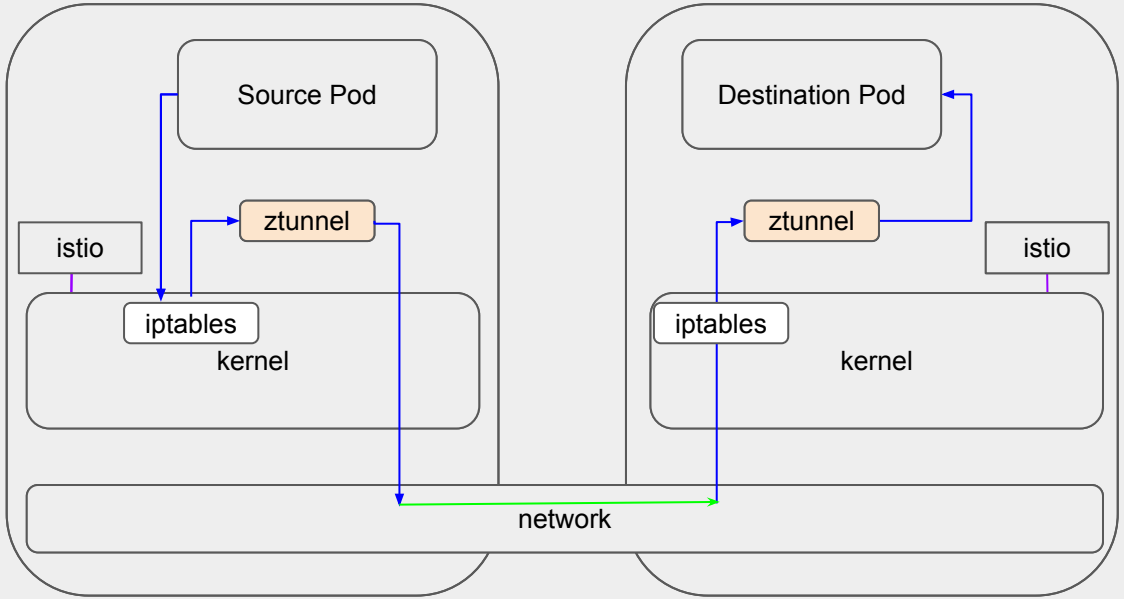
mTLS
control
data

Ambient Mesh



mTLS
control
data

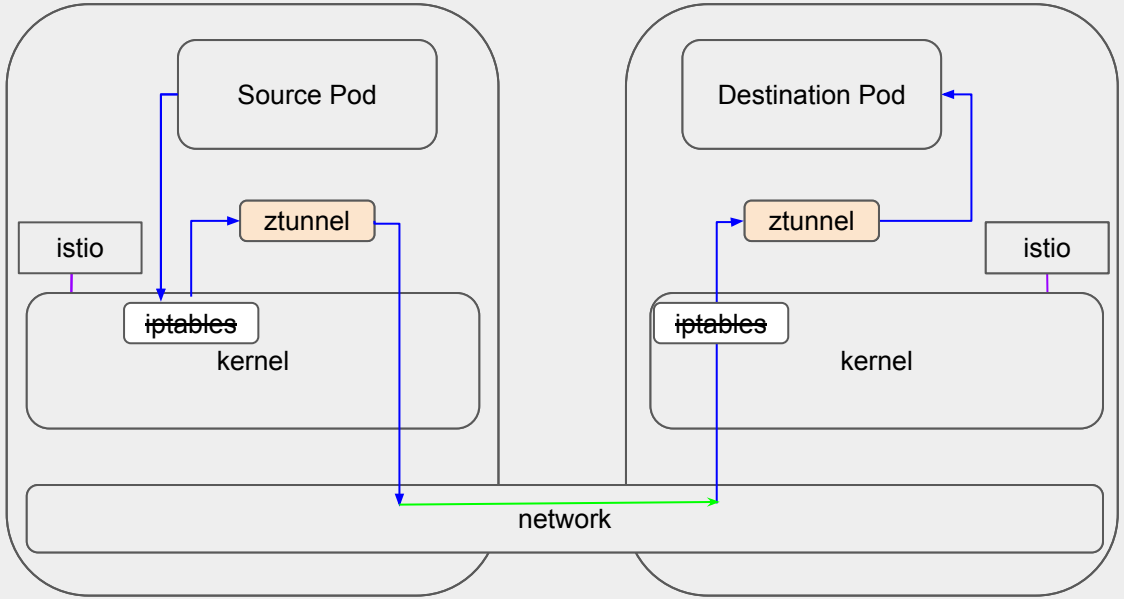
Ambient Mesh



mTLS
control
data

https://github.com/istio/istio/pull/42372

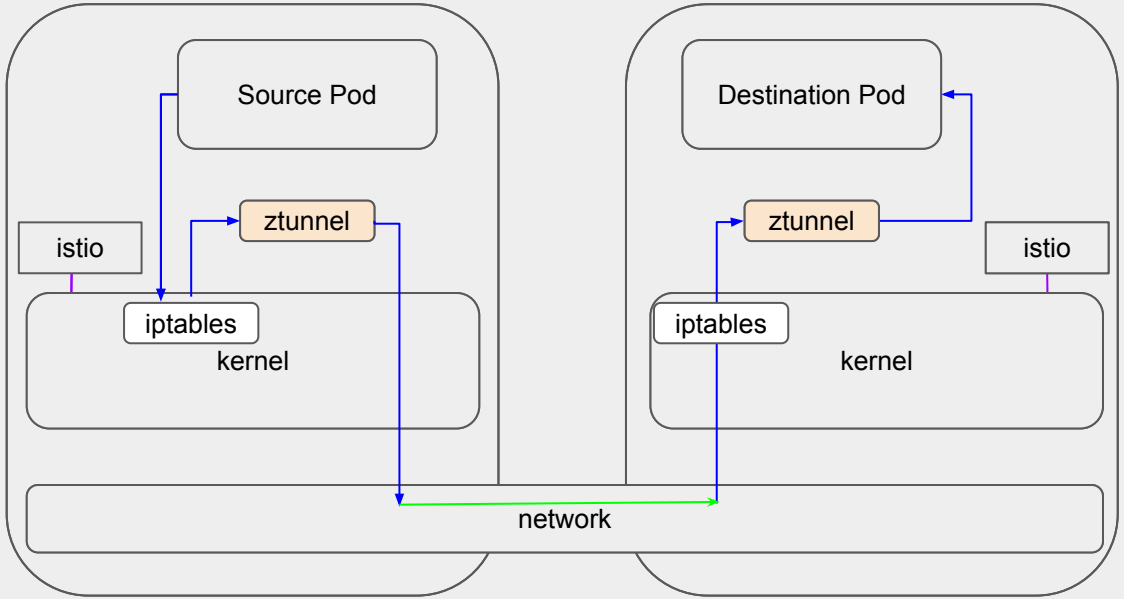
Ambient Mesh



mTLS
control
data

https://github.com/istio/istio/pull/42372

Ambient Mesh



mTLS
control
data

https://merbridge.io/

Ambient Mesh

L7 Processing Layer

Waypoint Proxy

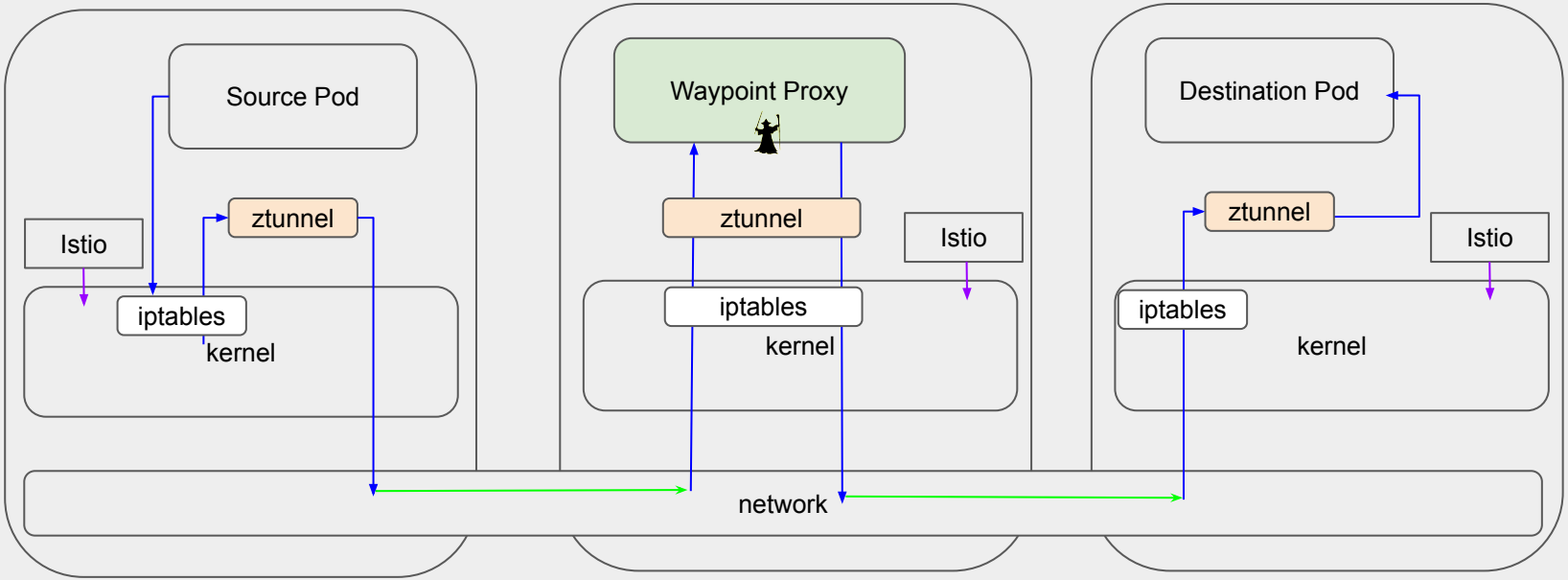
- Rich Authz Policies

L4 Secure Overlay

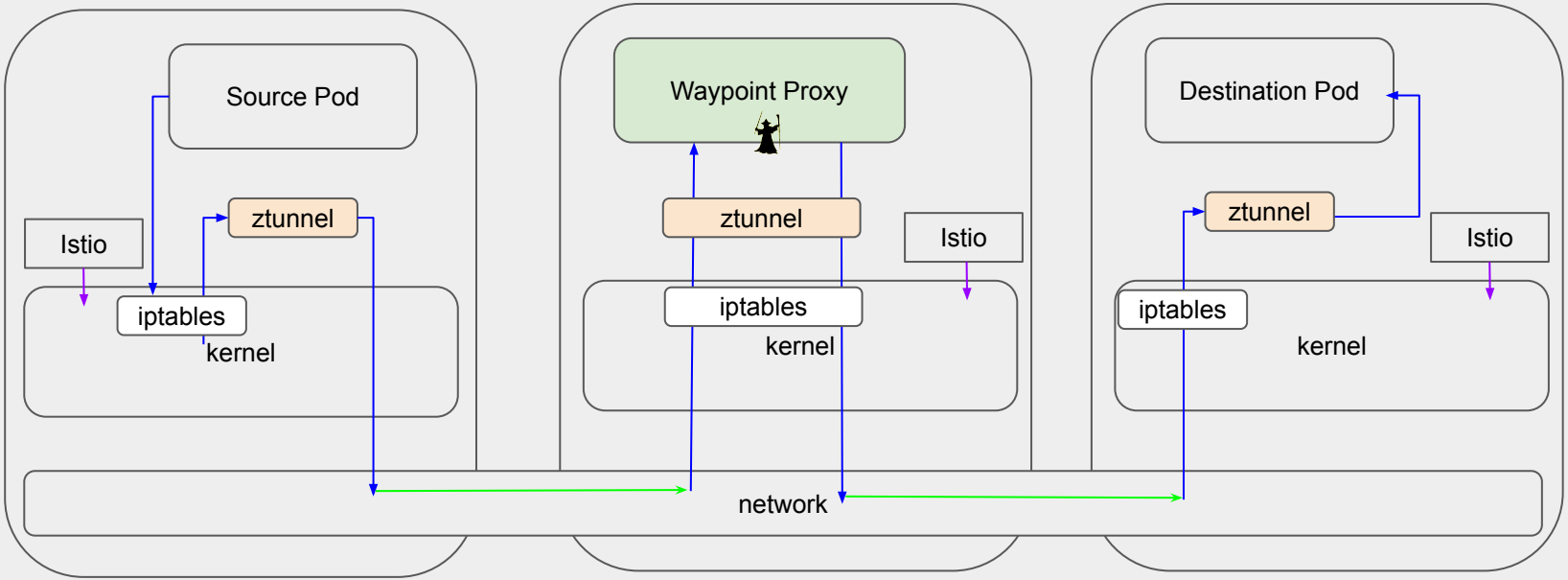
ztunnel

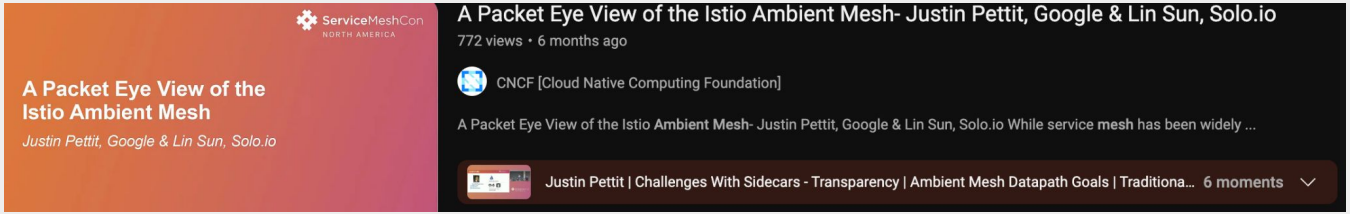
- mTLS,
- Svc-to-svc Authz Policies

Ambient Mesh

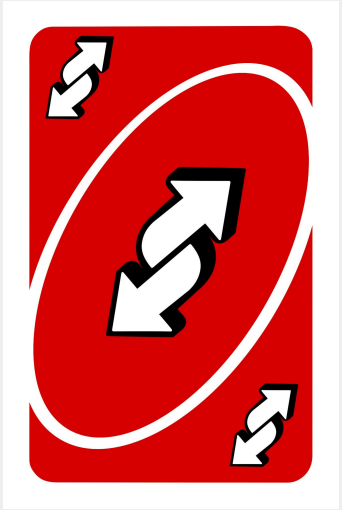


Ambient Mesh





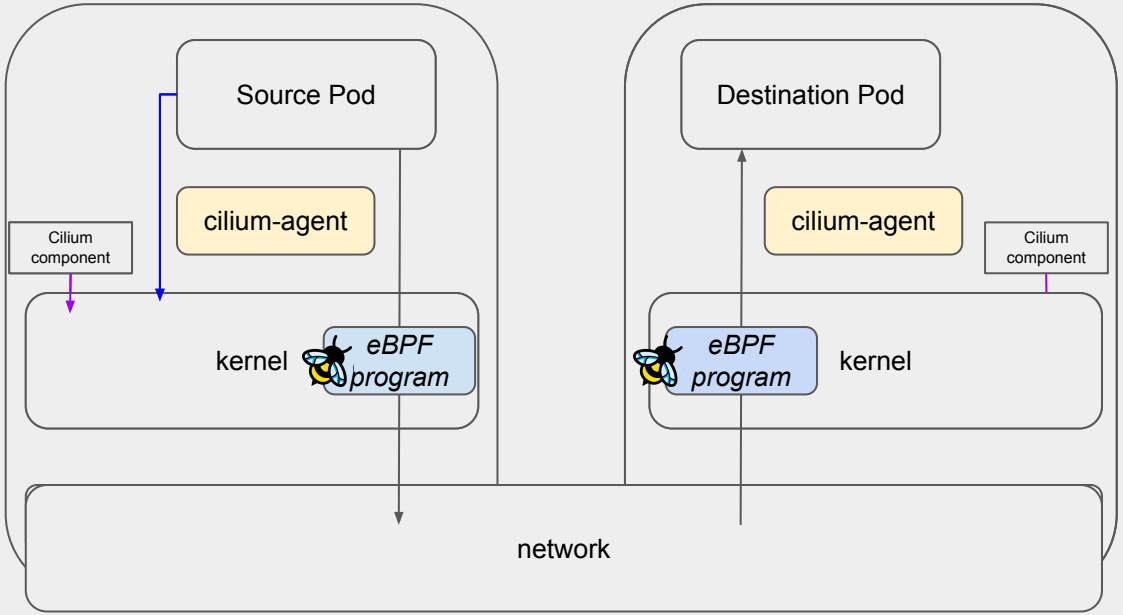
mTLS
control
data



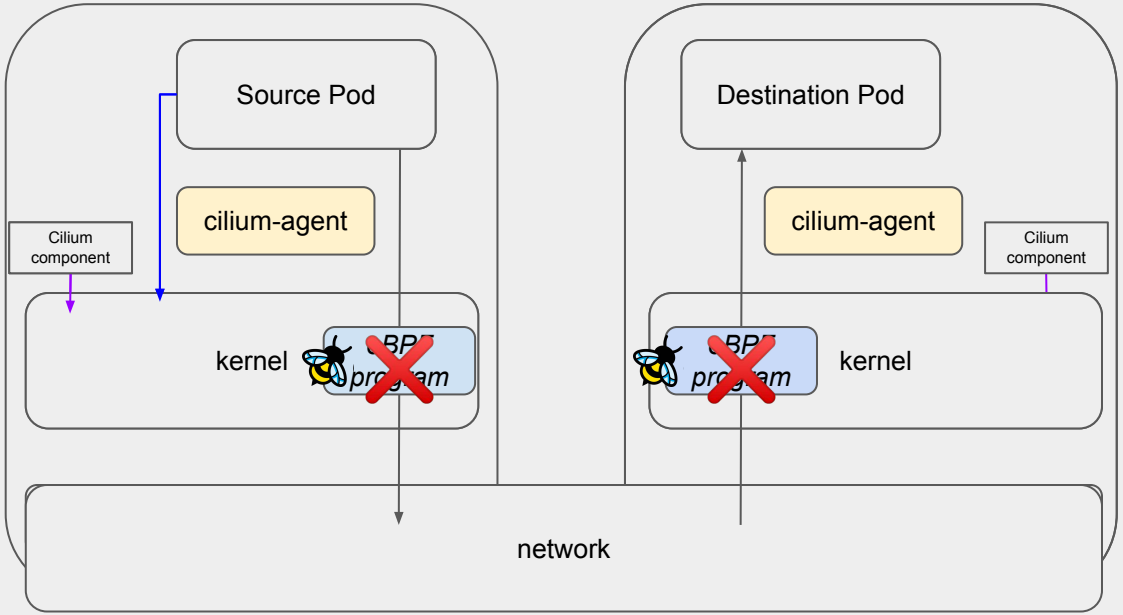
The landscape is changing 🔮

The landscape is changing 🔮

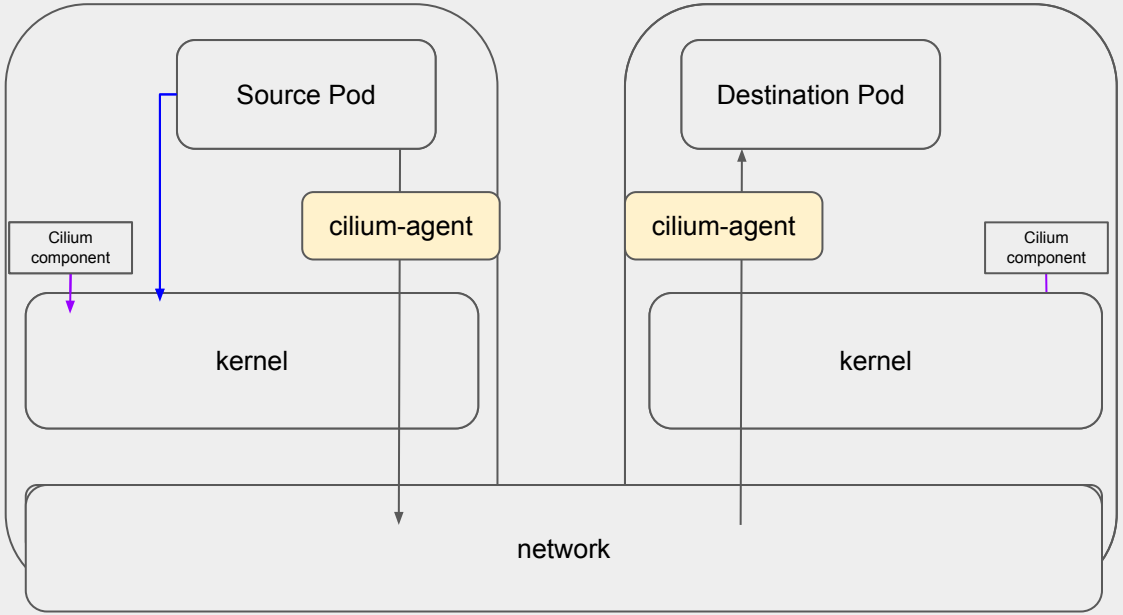
Cilium Service Mesh



Cilium Service Mesh

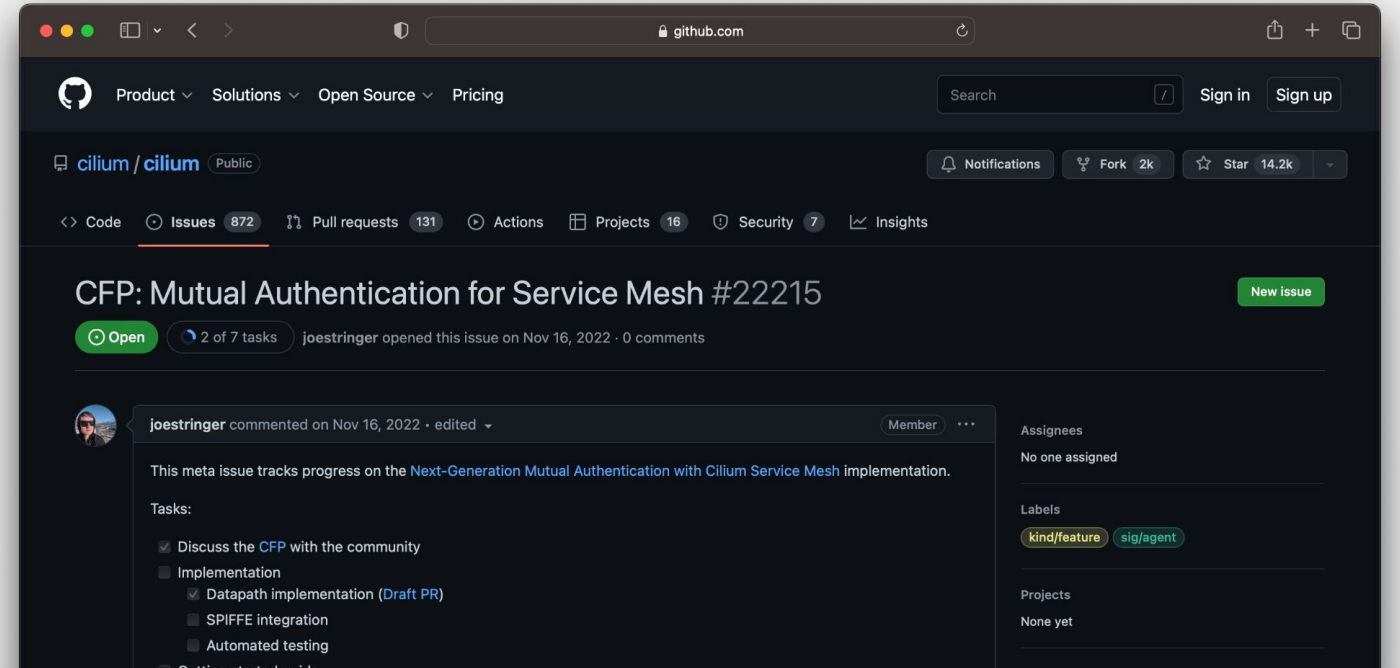


Cilium Service Mesh



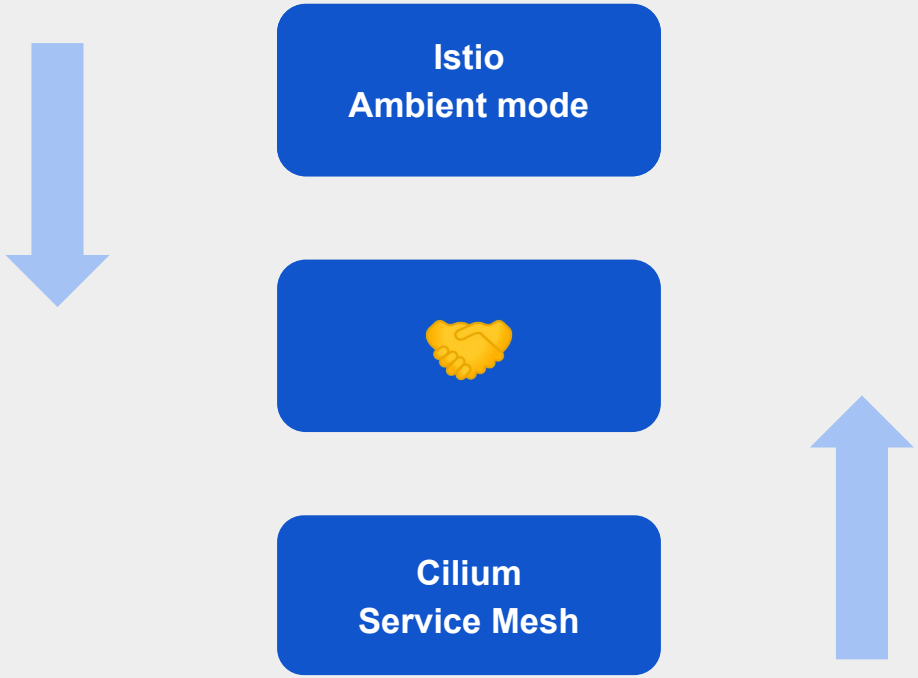
Cilium Service Mesh

- https://github.com/cilium/cilium/issues/22215



The landscape is changing 🔮

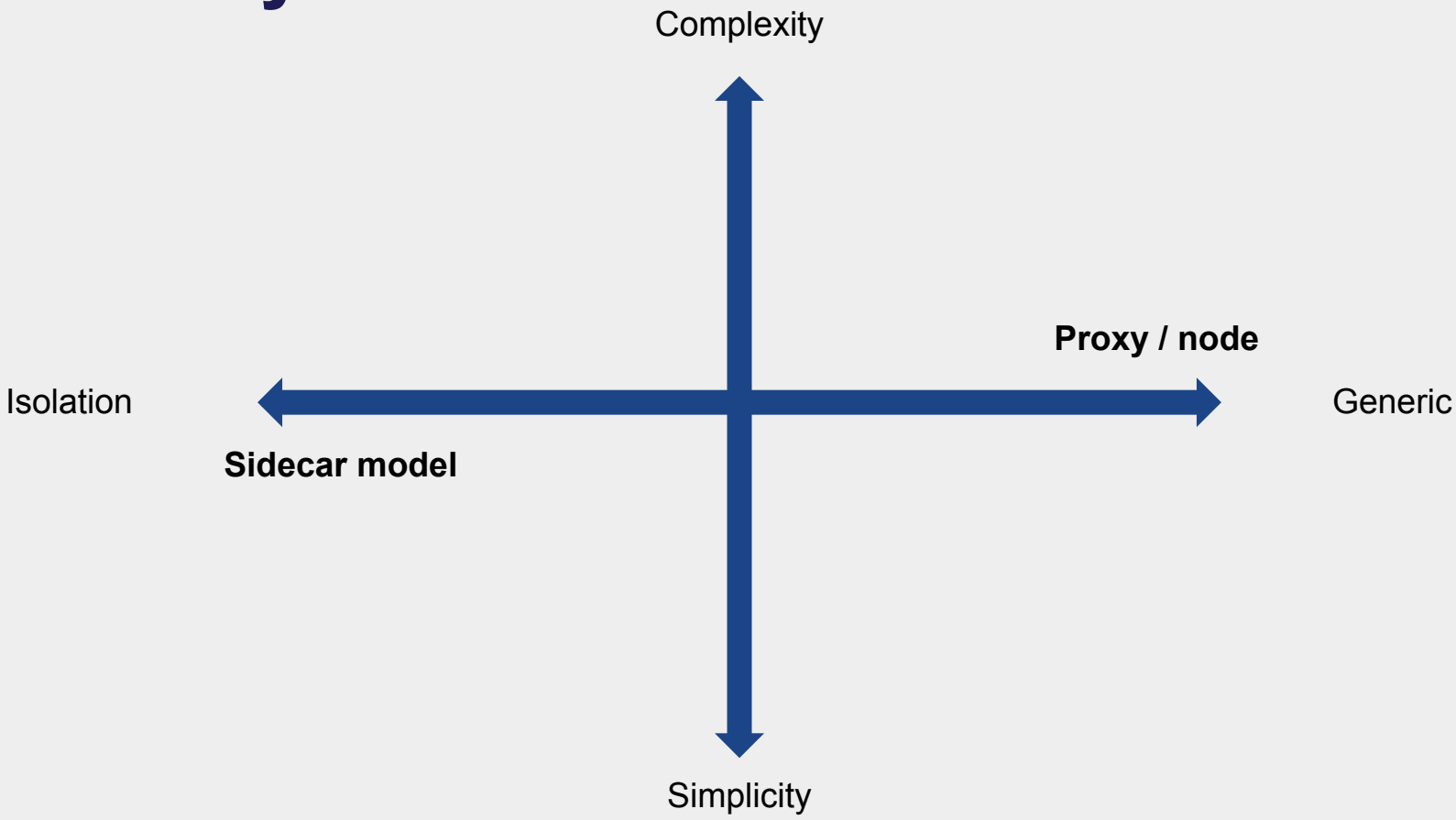
The landscape is ~~changing~~ converging



Takeaways

bit.ly/3YhFrRd

Takeaways


Complexity
Proxy / node
Isolation
Generic
Sidecar model
Simplicity

eBPF

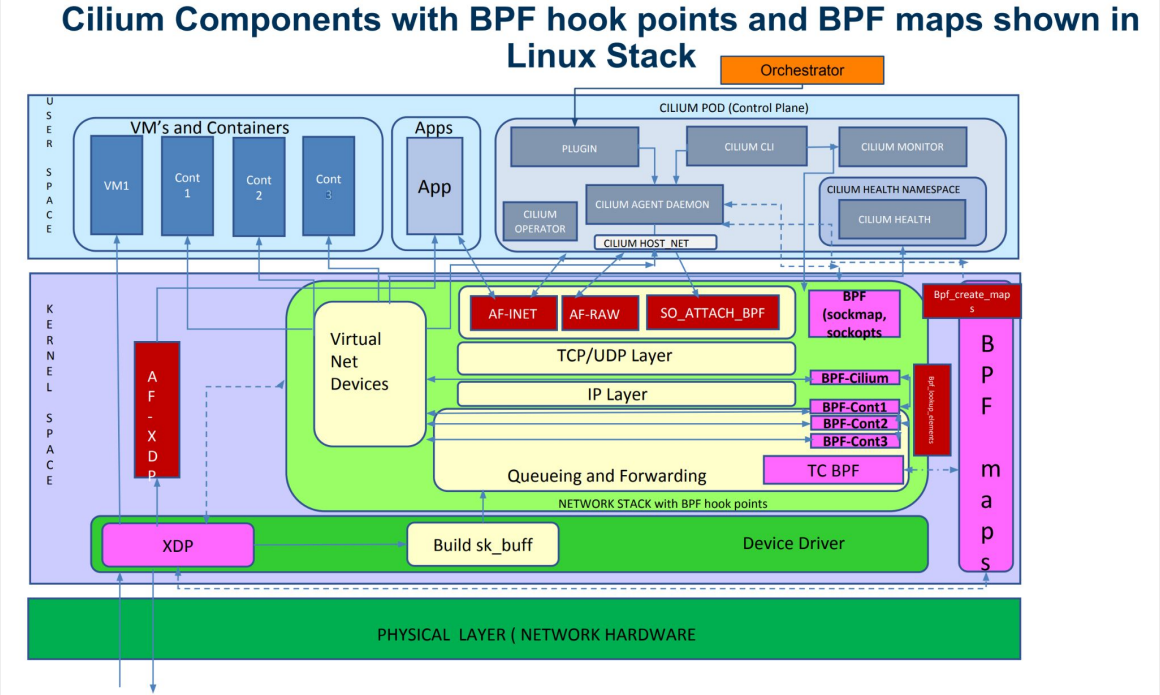
eBPF- CNI 

- L3 Observability
- L3 Routing
- L3 Network policy

*Replacing iptables with eBPF in Kubernetes with Cilium - **bit.ly/3DuNNgt***

eBPF

- L3 Observability
- L3 Routing
- L3 Network policy



*Replacing iptables with eBPF in Kubernetes with Cilium - **bit.ly/3DuNNgt***

eBPF


twitter.com
Explore
Settings
Thread
William Morgan
@wm
I'm thrilled to announce sidecar-free @Linkerd! In the next release, we will ship a fork of kubectl that uses #eBPF to remove references to linkerd-proxy from its output. This allows us to shift L7 processing "down" into underlying infrastructure, using the magic of eBPF! ✨
4:42 PM · Oct 3, 2022
52 Retweets
16 Quote Tweets
405 Likes
Search Twitter
New to Twitter?
Sign up now to get your own personalized timeline!
Sign up with Google
Sign up with Apple
Create account
By signing up, you agree to the Terms of Service and Privacy Policy, including Cookie Use.
Relevant people

Takeaways

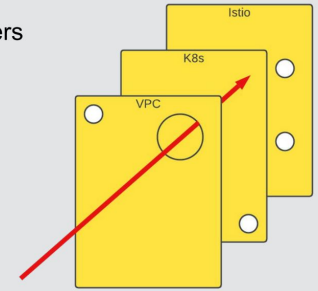
Takeaways

Takeaways

Takeaways

CLOUDNATIVE SECURITYCON NORTH AMERICA 2023

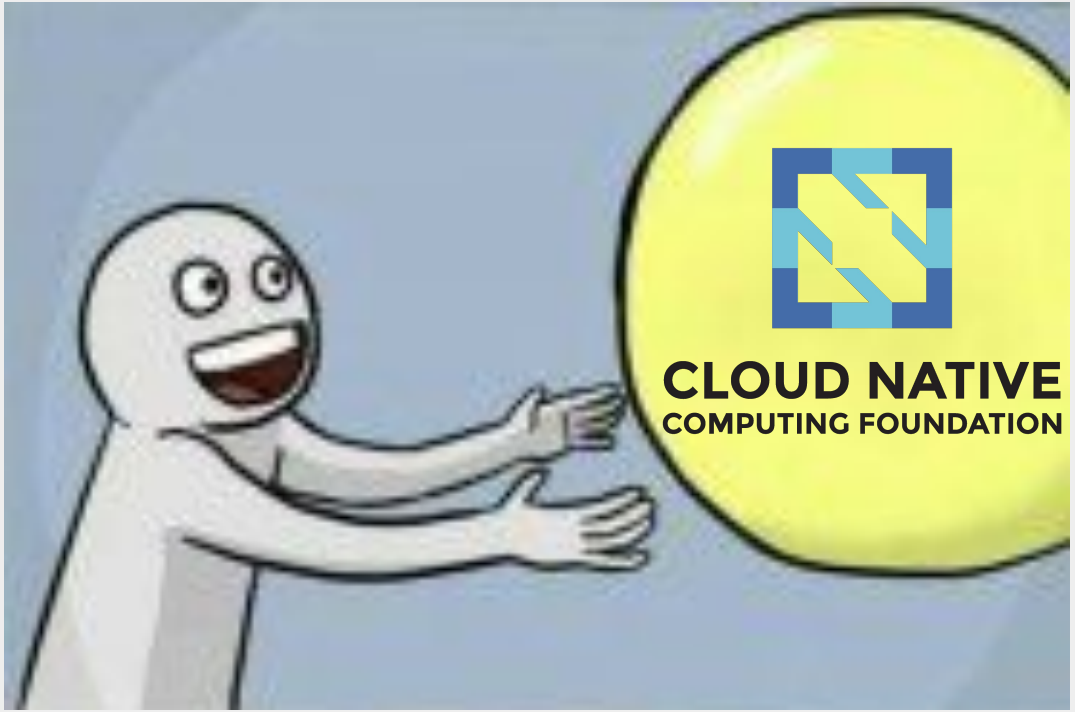
- **Self-service platforms are hard**
  - Safeguards to avoid users shooting themselves in the foot
  - Provide a Golden Path to avoid configuration errors
- **Defense in depth**: Add redundant security at all layers
- **Observability is key**
  - Help debug
  - Detect misconfiguration



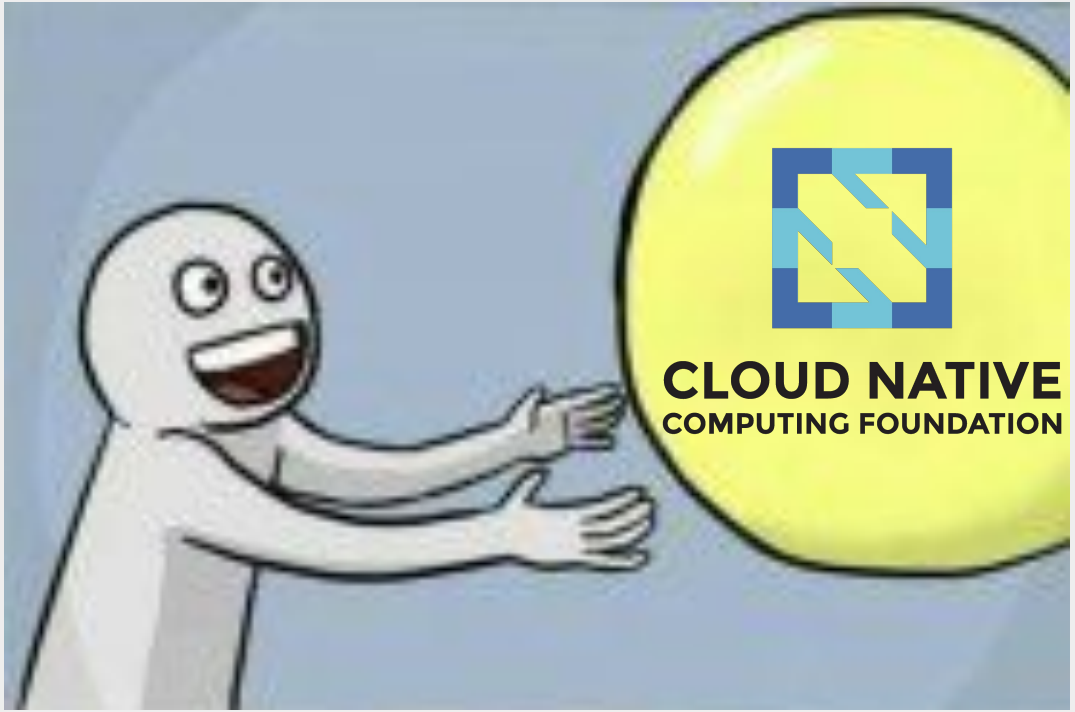
bit.ly/3Hy86uD

Call to action

Call to action

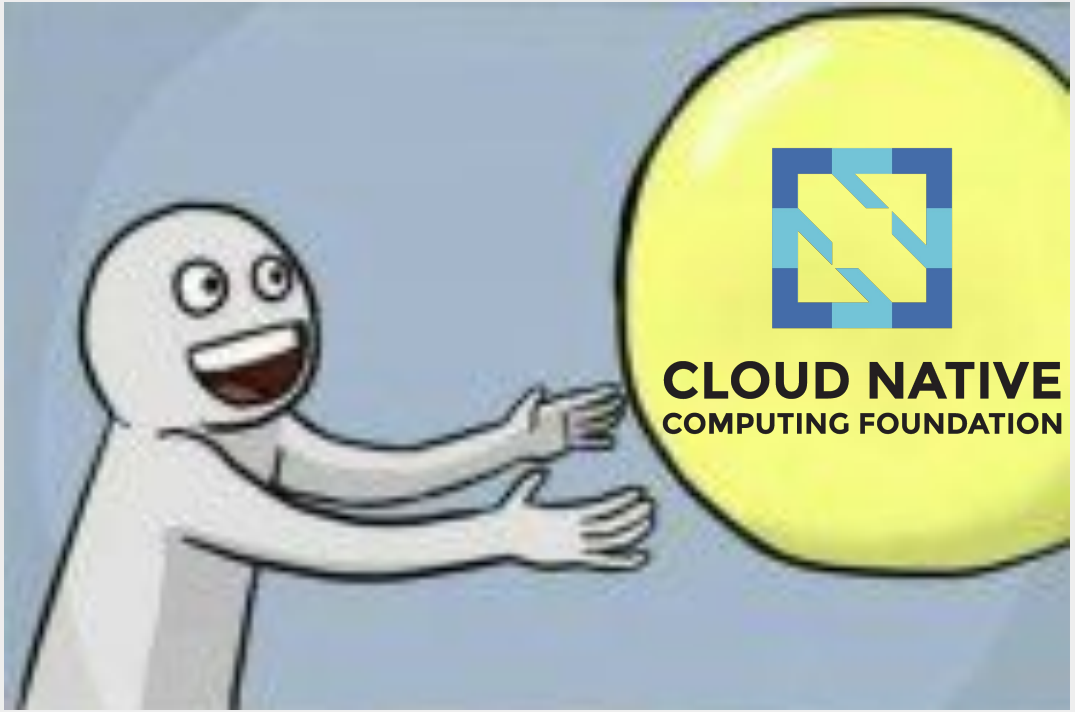


Call to action



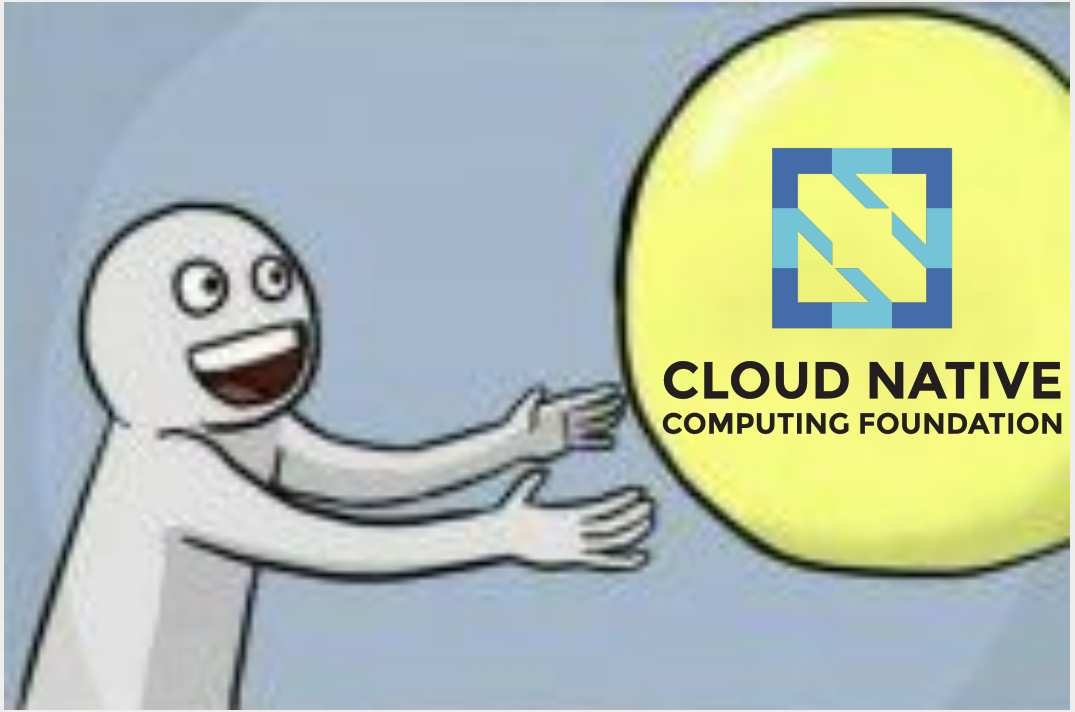
https://istio.io/latest/blog/2022/get-started-ambient/

Call to action

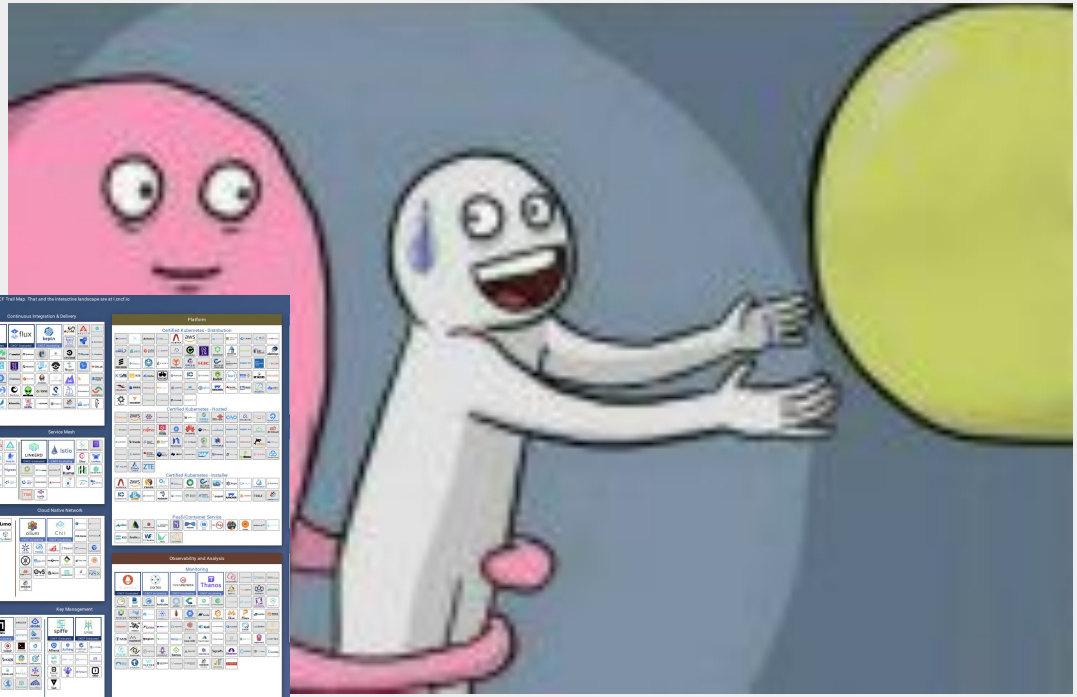


https://github.com/cilium/cilium/issues/22215

Call to action

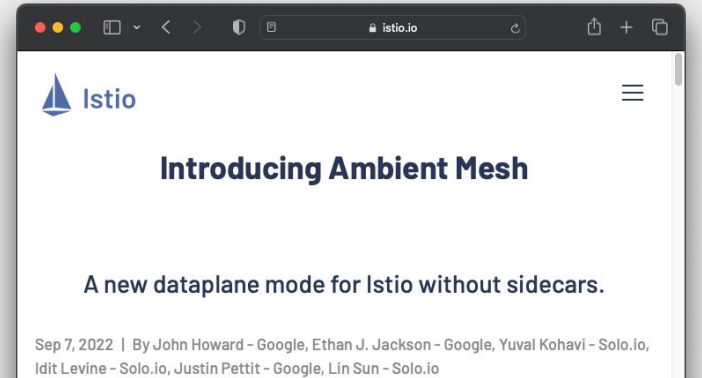


Call to action



cilium

Istio



Christine Kim - Isovalent

Rob Salmond - SuperOrbital