Improving your Kubernetes Workload Security with Hardware Virtualization

Fabian Deutsch, fdeutsch@redhat.com, Red Hat
Samuel Ortiz, samuel.ortiz@intel.com, Intel
KubeCon Europe, Copenhagen, 2018

Agenda

- Event link
  - Wednesday May 2, 2018 16:25 - 17:00
  - C1-M3
  - Security+Identity+Signing, Intermediate
- Total: 35min
  - 3min Intro
  - 11min Kata
  - 11min KubeVirt
  - 10min summary & QA

2 random maintainers

What kind of workloads can Kubernetes run?

Cloud native apps

Cloud native apps

No legacy workloads

Cloud native apps

No legacy workloads

One kernel for all

Cloud native apps

No legacy workloads

One kernel for all

Software based isolation


FEAR NOT!
I'LL HELP
imgflip.com

Hardware Virtualization

a.k.a. virtual machines

a.k.a. virtual machines

CPU and Memory virtualization

a.k.a. virtual machines

CPU and Memory virtualization

Devices virtualization

a.k.a. virtual machines

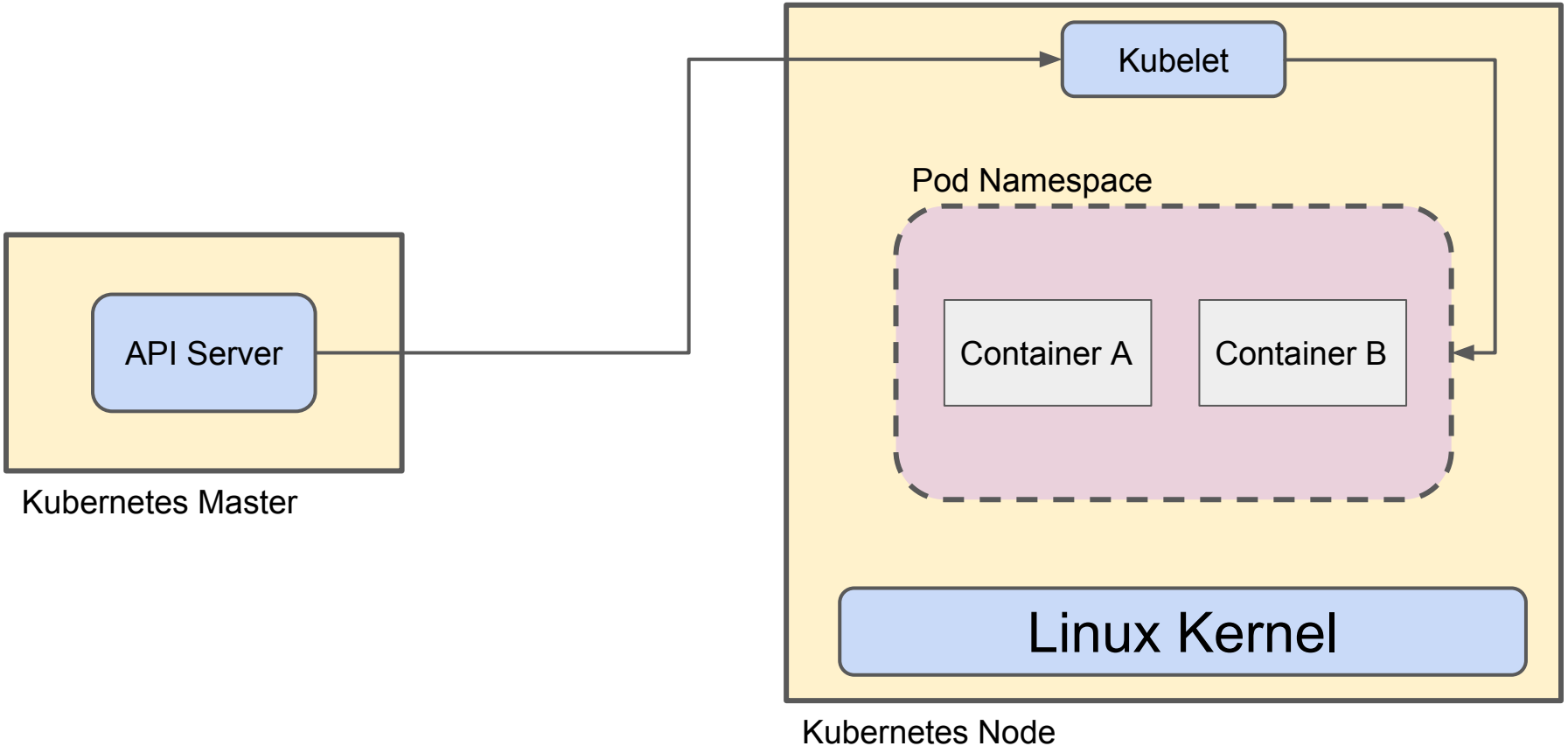
CPU and Memory virtualization

Devices virtualization

Transparent to Kubernetes

Cloud Native Applications

Kubernetes

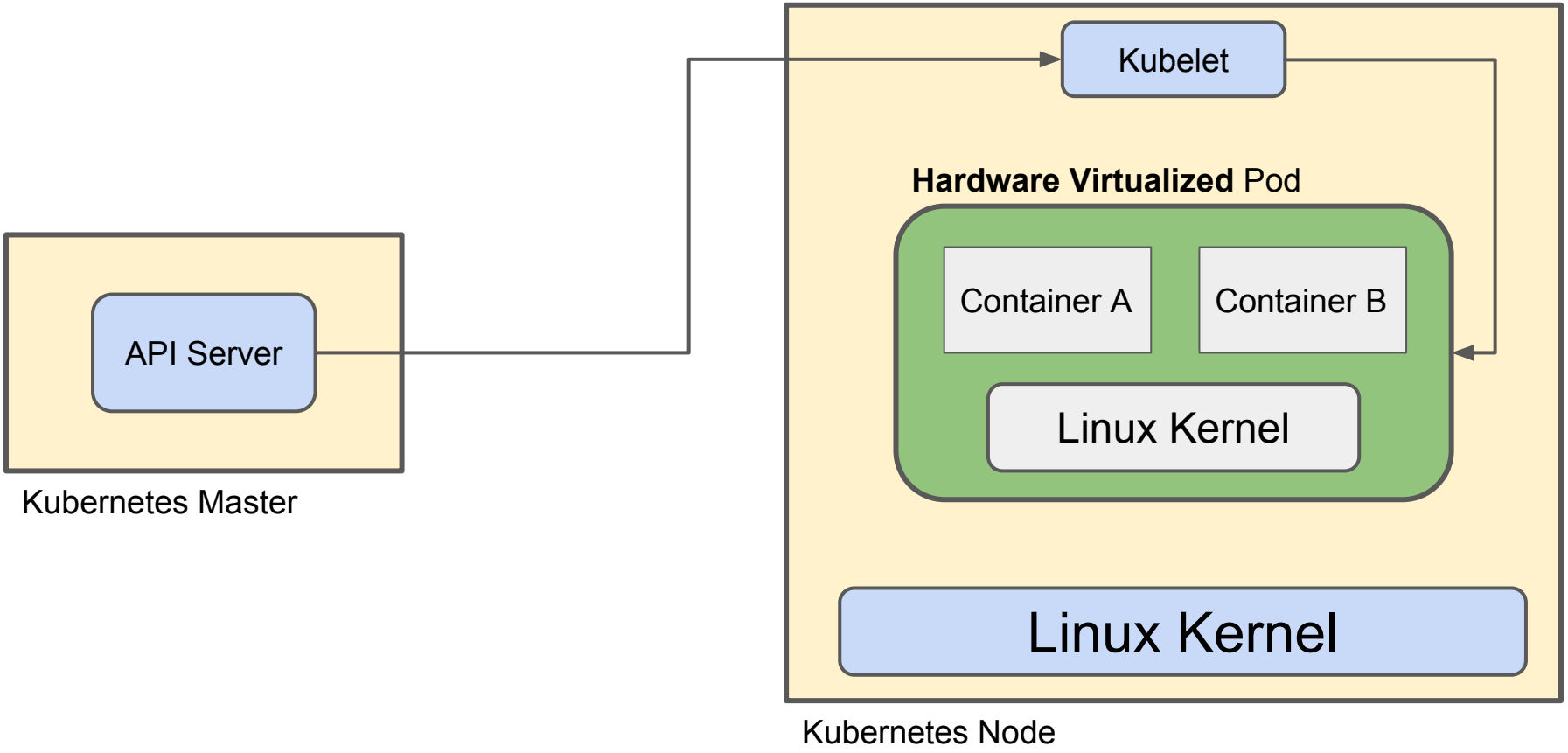

Kubelet
Pod Namespace
API Server
Container A
Container B
Linux Kernel
Kubernetes Master
Kubernetes Node

Adding a new isolation layer...

Adding a new isolation layer...

Cloud Native Apps With Stronger Isolation

Kubernetes and Kata Containers

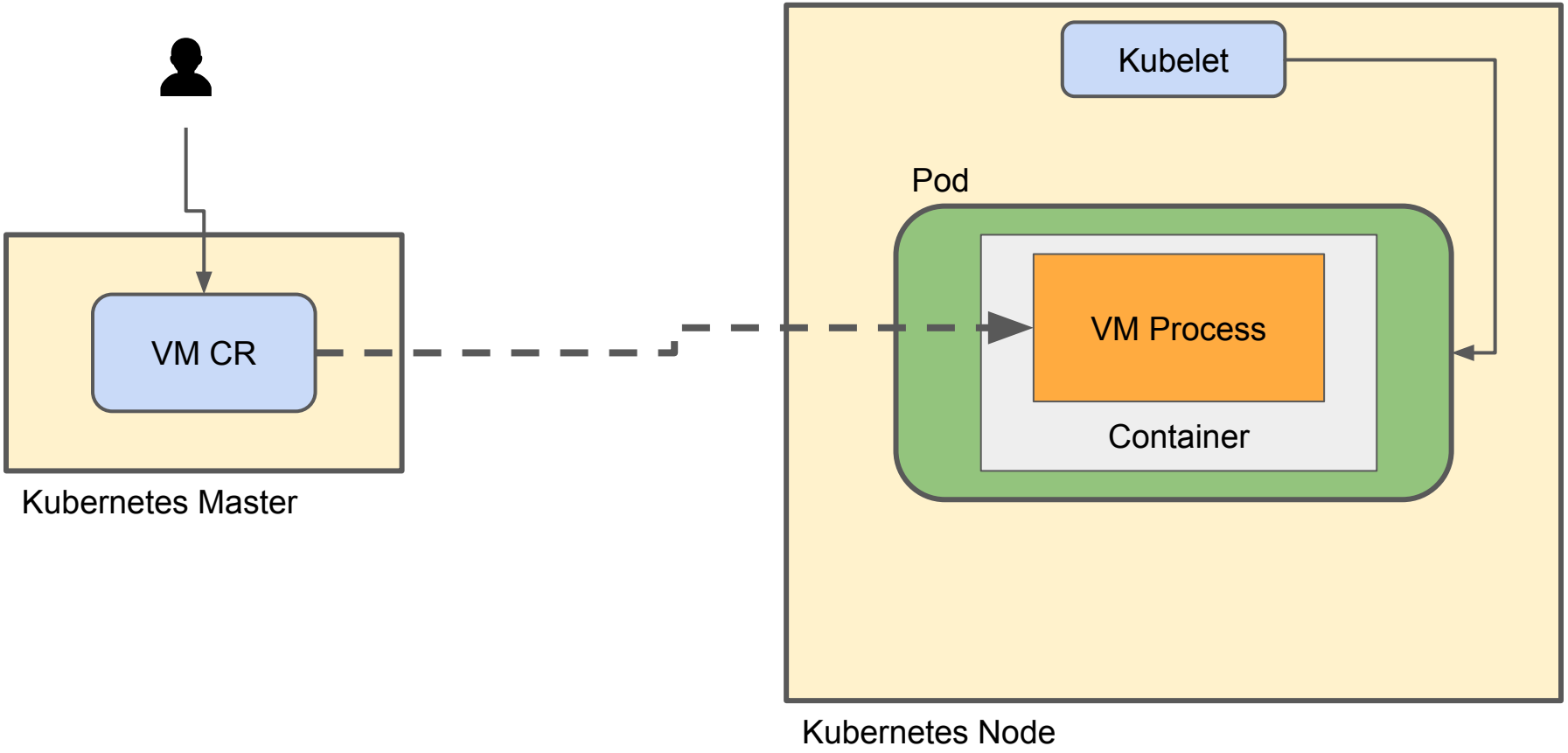


Adding a new Custom Resource...

Adding a new Custom Resource…

Isolation to run legacy applications

Kubernetes and KubeVirt



Strong Isolation for Cloud Native Workloads

Kata Containers

Kata Containers

OCI compatible runtime

Kata Containers

OCI compatible runtime

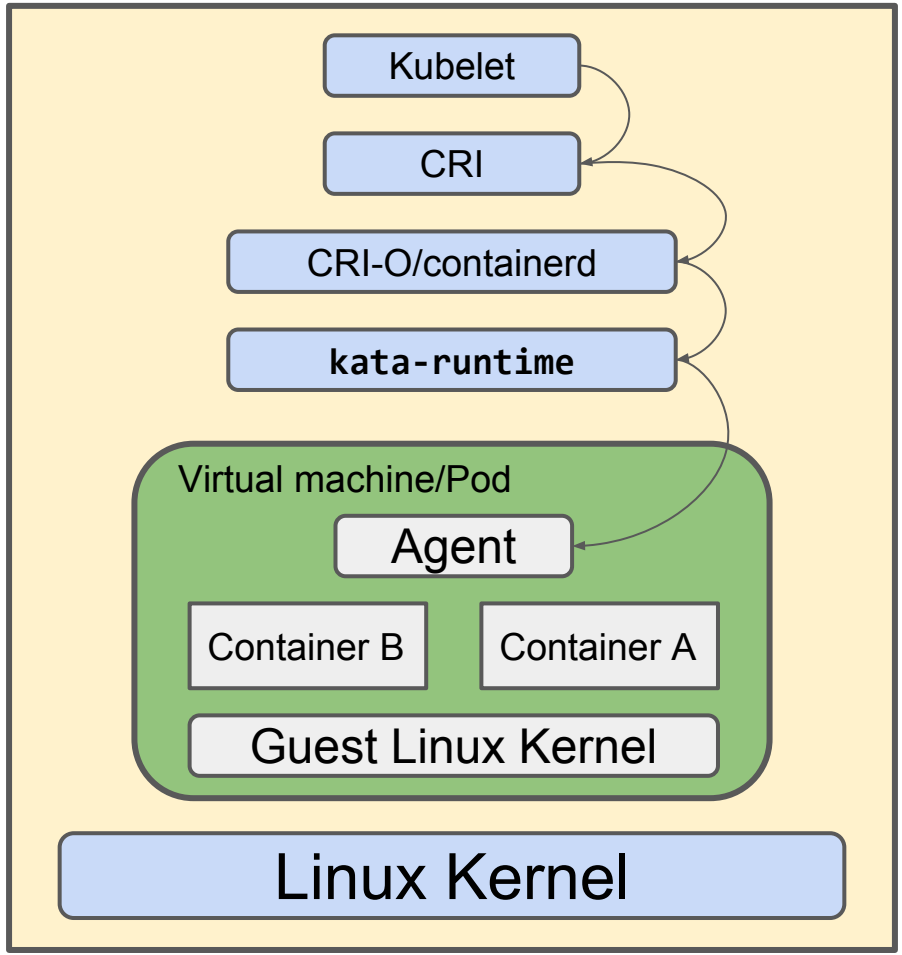
Native CRI APIs

Kata Containers

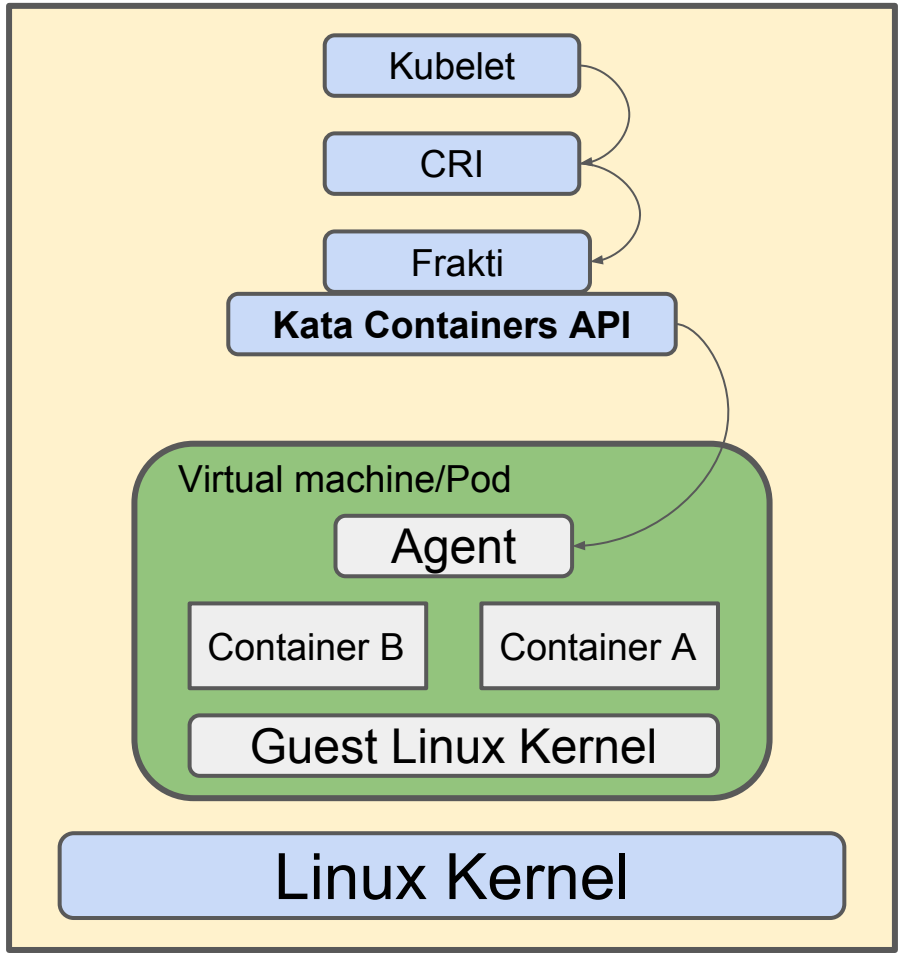
OCI compatible runtime

Native CRI APIs

One VM per pod



Kubernetes Node



Kubernetes Node

Kata Containers 1.0

Kata Containers 1.0

Kubernetes Secure Containers

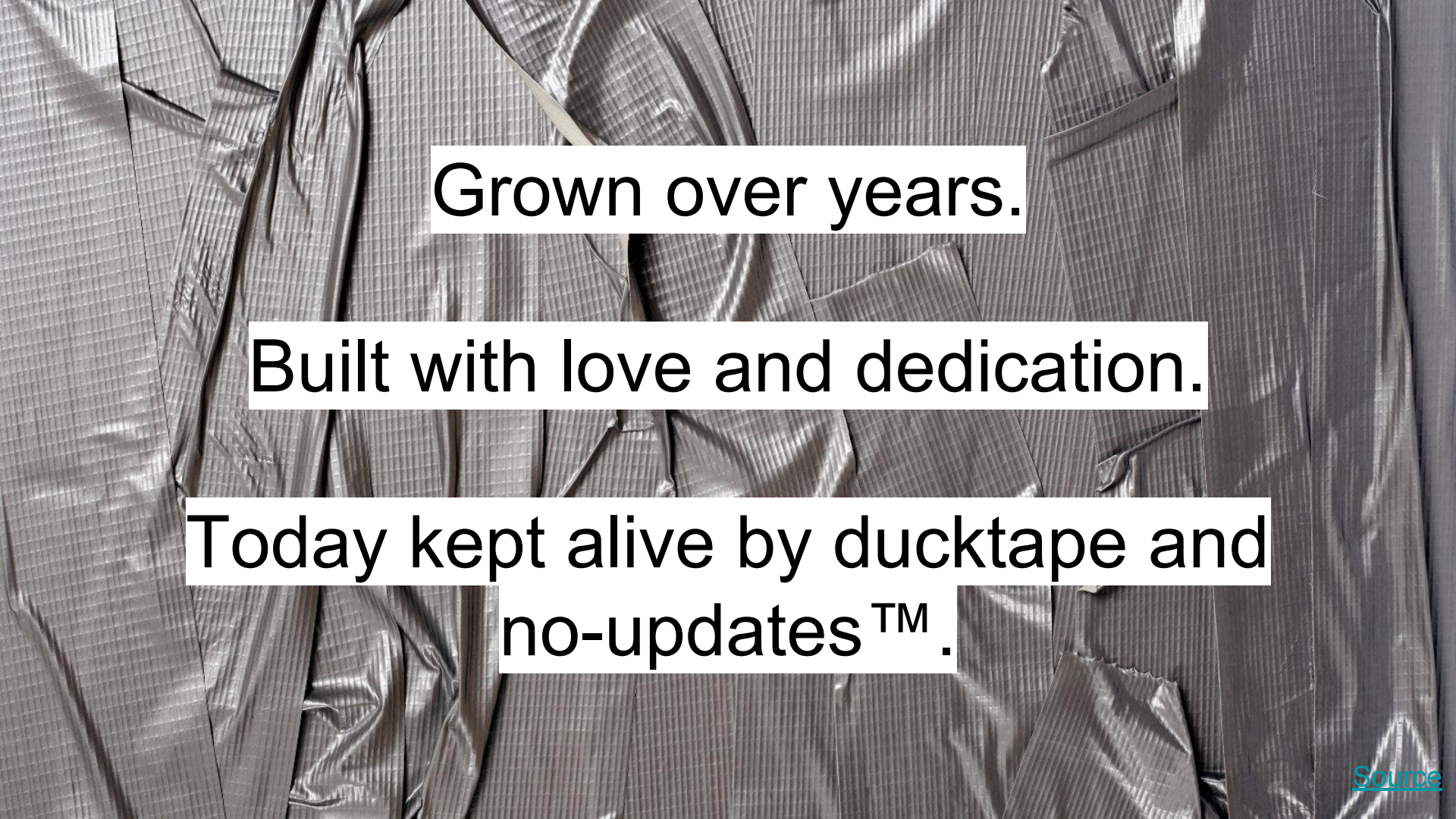
Kata Containers 1.0

Kubernetes Secure Containers

VMs as first class k8s citizen

Strong Isolation for Legacy Workloads

Legacy workloads?


Grown over years.
Built with love and dedication.
Today kept alive by ducktape and no-updates™.

Source

… and all the workflows around them

We can’t or don’t want to change them now

Because of technical and/or economical reasons to keep them as is.

How would you run legacy workloads in a cloud-native world then?

Virtual Machines

How would you run legacy workloads in a cloud-native world then?

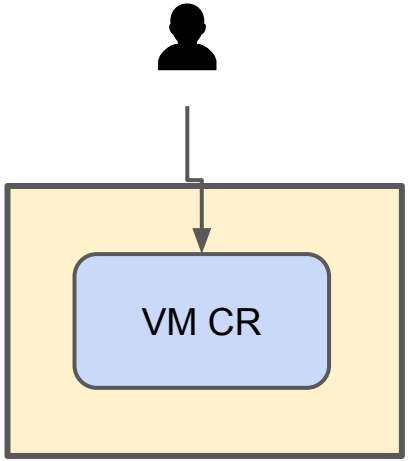
KubeVirt

Virtualization Operator

API and runtime

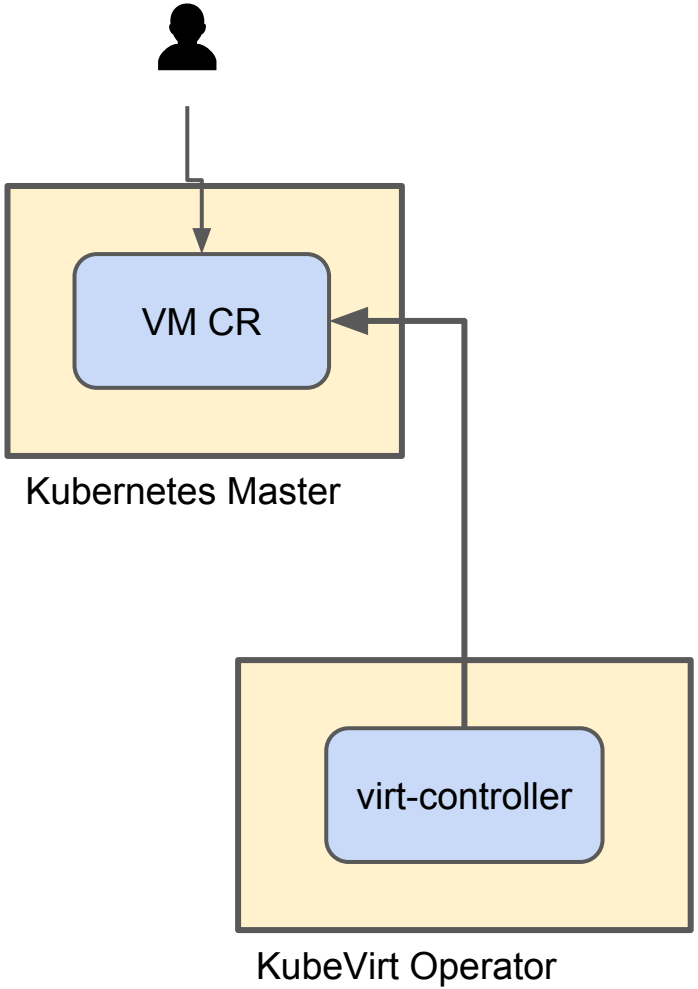
VMs as you known them

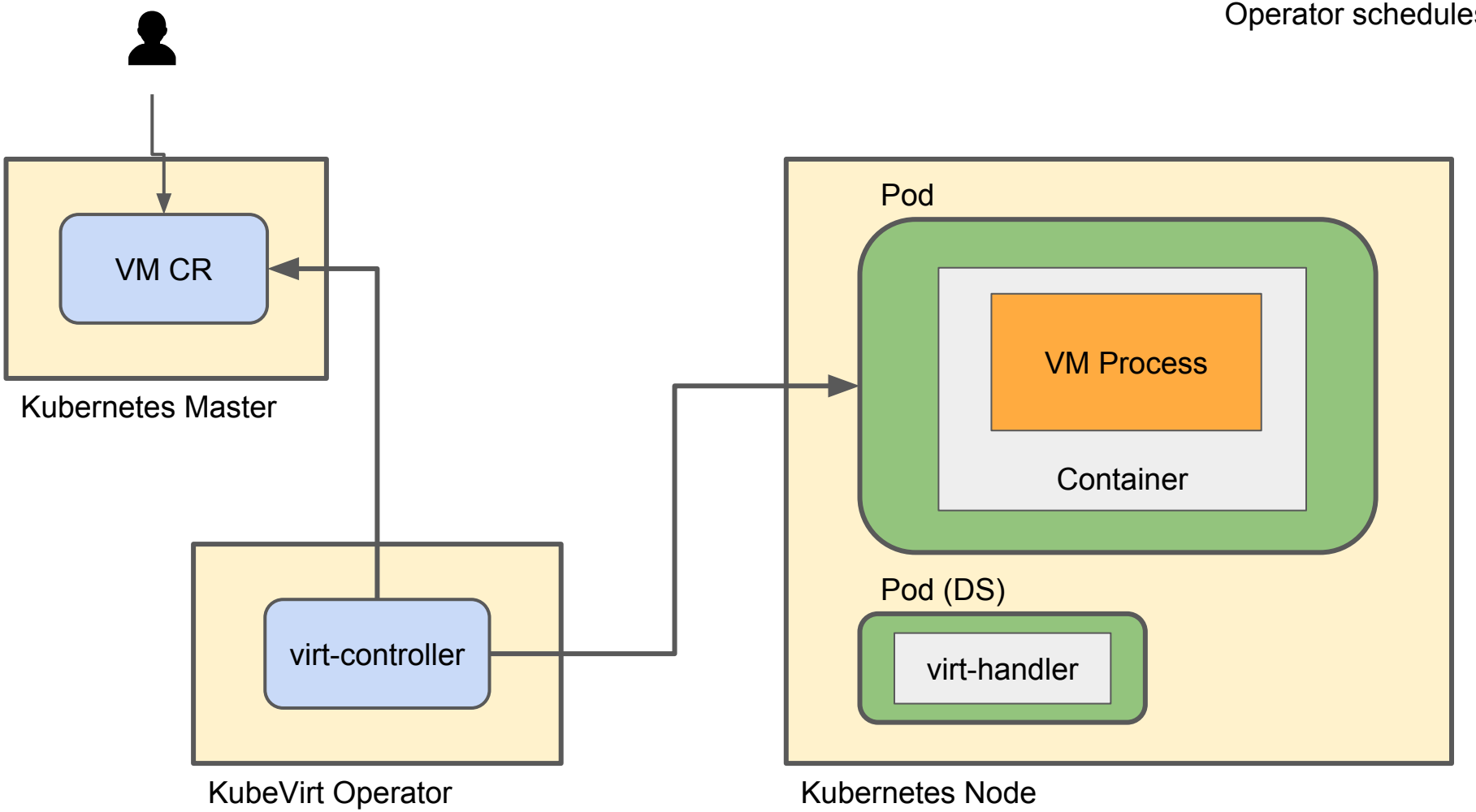
User request



Kubernetes Master

Operator notices




Operator schedules
VM CR
Kubernetes Master
Pod
VM Process
Container
Pod (DS)
virt-handler
virt-controller
KubeVirt Operator
Kubernetes Node

```
$ kubectl apply -f kubevirt-operator.yml
$ kubectl get vms
$ virtctl vnc |
```

Try on minikube: https://github.com/kubevirt/demo

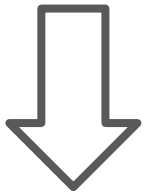
Status v0.4.1

- Native Kubernetes storage integration (PV)
- Native Kubernetes network integration (pod networking)
- `virtctl` client utility
- Distros: Vanilla Kubernetes and OpenShift
- `VirtualMachinePresets` and `OfflineVirtualMachines` (User Guide, API)
- Future
  - Cockpit UI
  - Import from OVA / VMware / libvirt

Recap

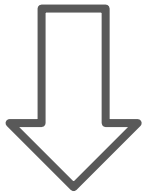
Two projects addressing **two** use-cases in the ecosystem.

Isolating cloud-native



https://katacontainers.io

Migrating legacy



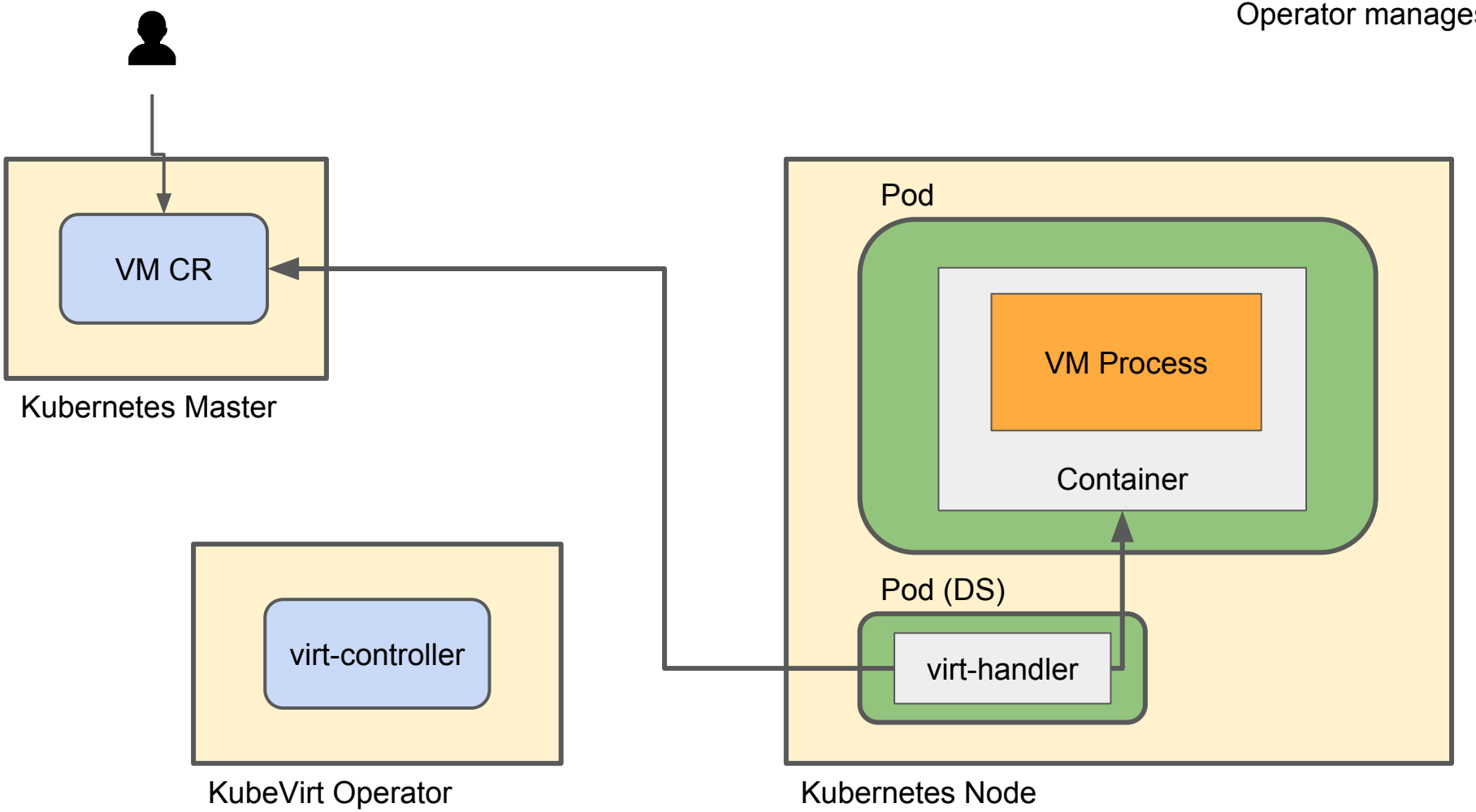
https://kubevirt.io

Questions?

Thank you.

https://katacontainers.io

https://kubevirt.io


Operator manages
VM CR
Kubernetes Master
Pod
VM Process
Container
Pod (DS)
virt-handler
virt-controller
KubeVirt Operator
Kubernetes Node