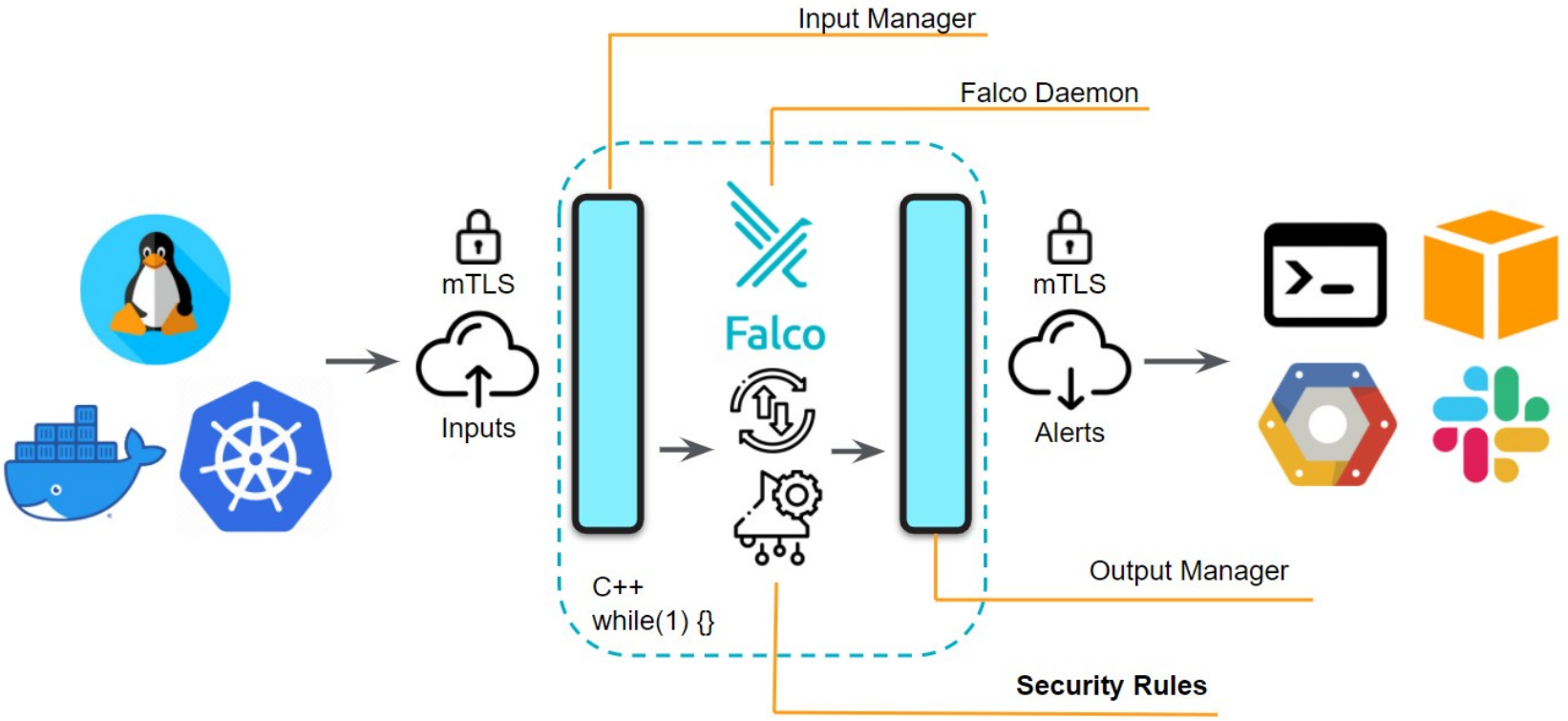
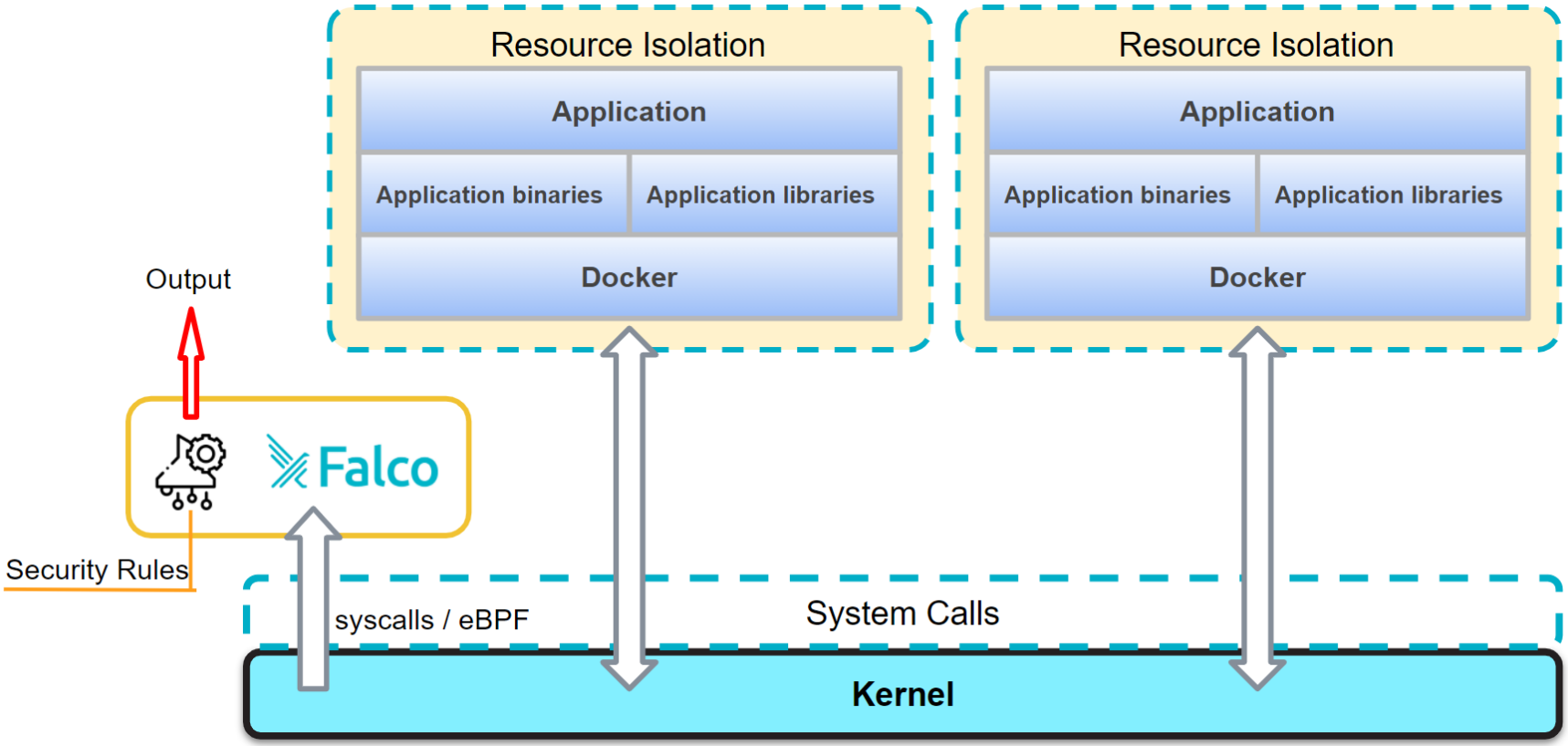
Falco para Kubernetes

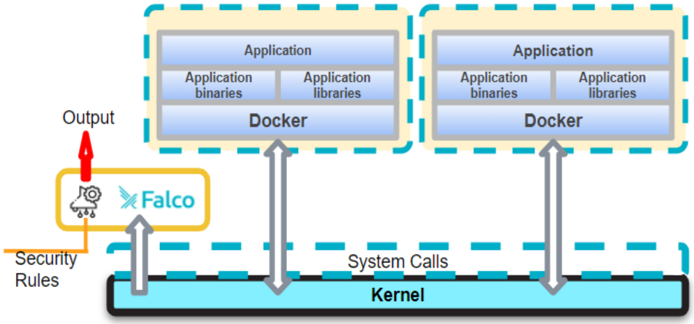
Seguridad runtime en clusters Kubernetes con Falco

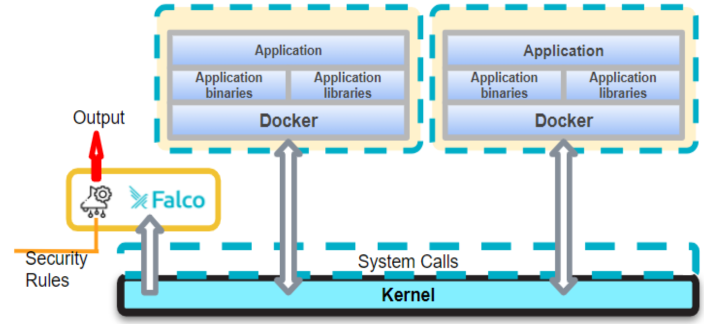
Arquitectura

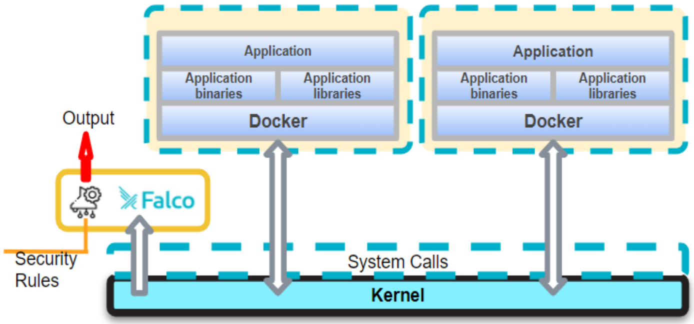


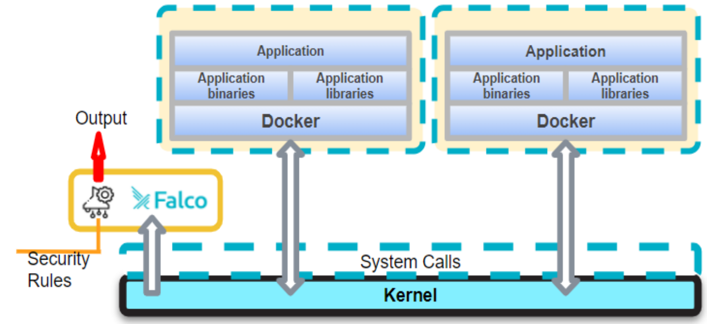

Resource Isolation
Application
Application binaries
Application libraries
Docker
Resource Isolation
Application
Application binaries
Application libraries
Docker
Output
Falco
Security Rules
syscalls / eBPF
System Calls
Kernel

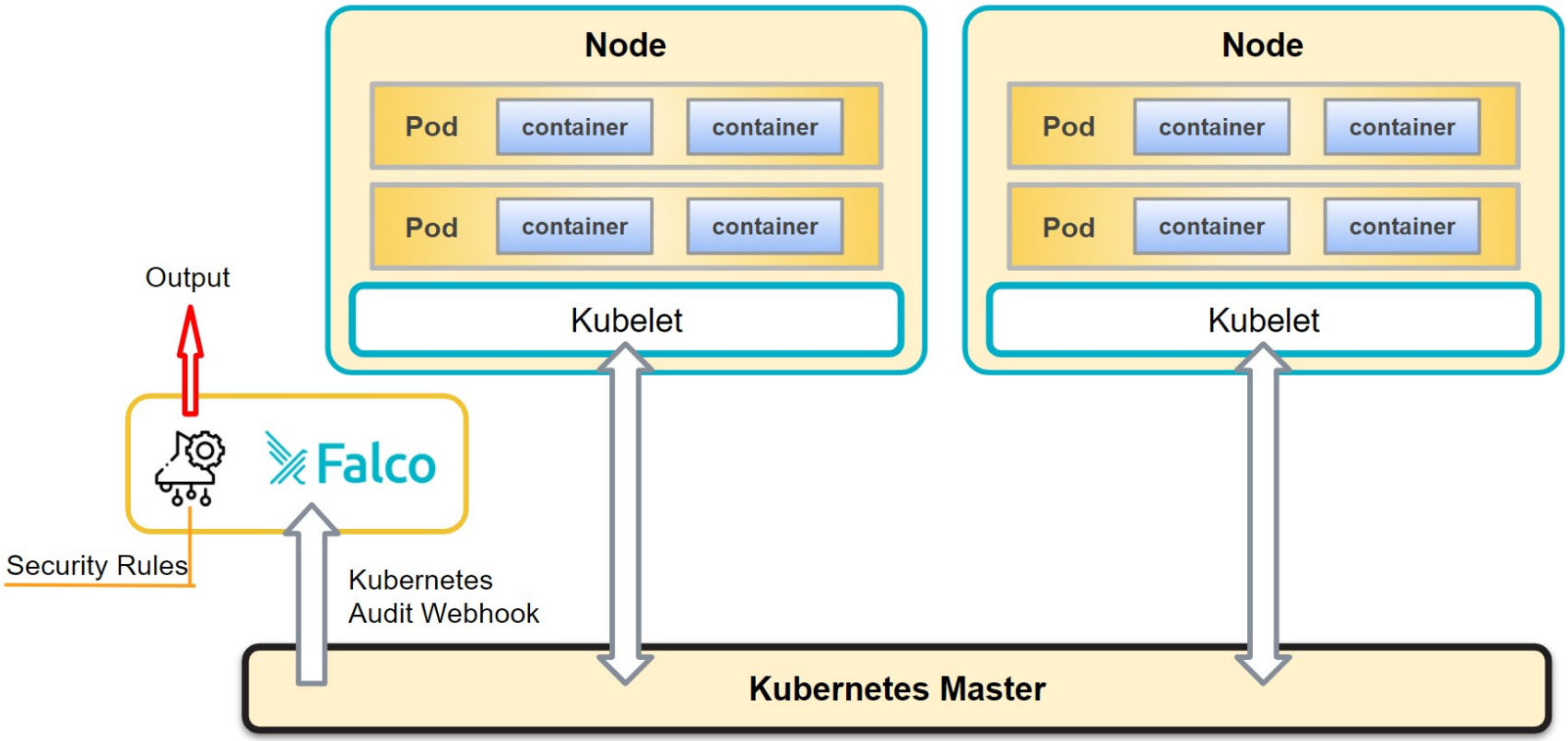
QUANTIKA 14


Application
Application binaries
Application libraries
Docker
Application
Application binaries
Application libraries
Docker
Output
Falco
Security Rules
System Calls
Kernel


Application
Application binaries
Application libraries
Docker
Application
Application binaries
Application libraries
Docker
Output
Falco
Security Rules
System Calls
Kernel


Application
Application binaries
Application libraries
Docker
Application
Application binaries
Application libraries
Docker
Output
Falco
Security Rules
System Calls
Kernel


Application
Application binaries
Application libraries
Docker
Application
Application binaries
Application libraries
Docker
Output
Falco
Security Rules
System Calls
Kernel


Node
Pod
container
container
Pod
container
container
Kubelet
Node
Pod
container
container
Pod
container
container
Kubelet
Output
Falco
Security Rules
Kubernetes
Audit Webhook
Kubernetes Master

falco.org

Sitio principal:

falco.org

Documentación:

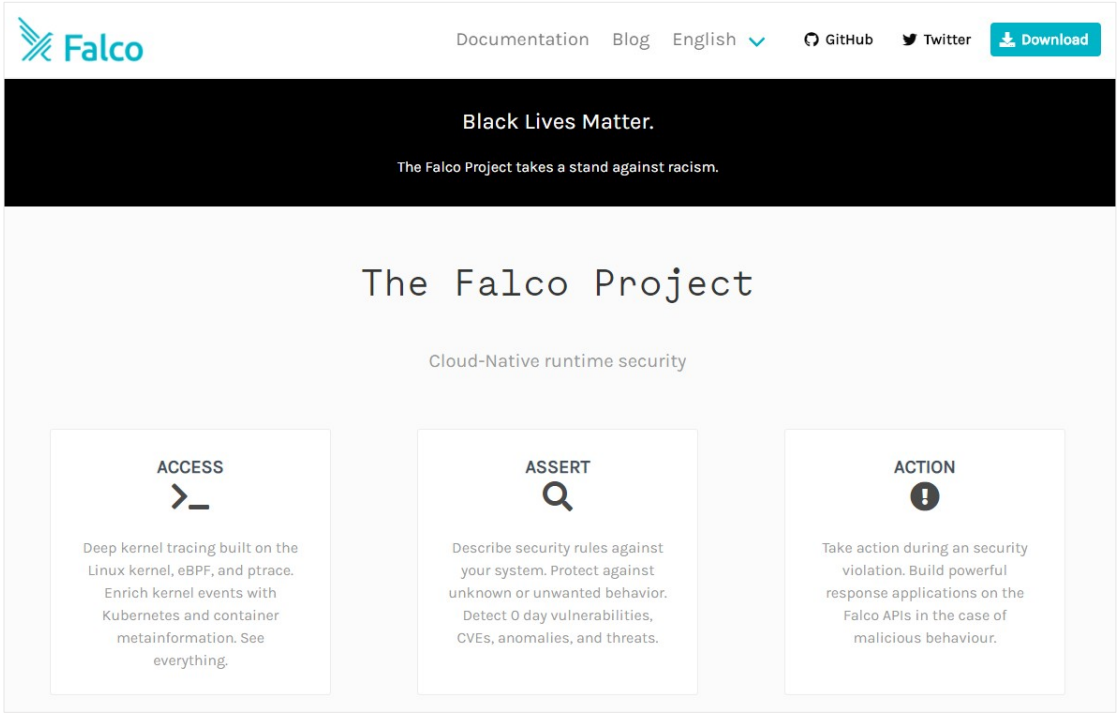
falco.org/docs/

Repositorio git:

github.com/falcosecurity/falco

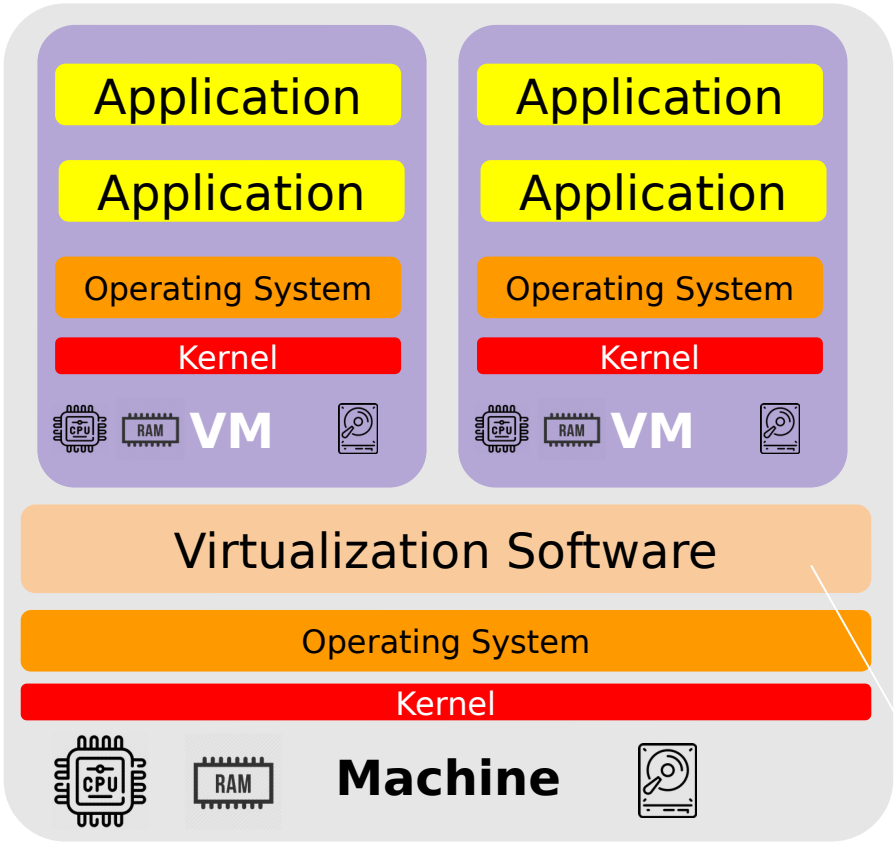
Blog:

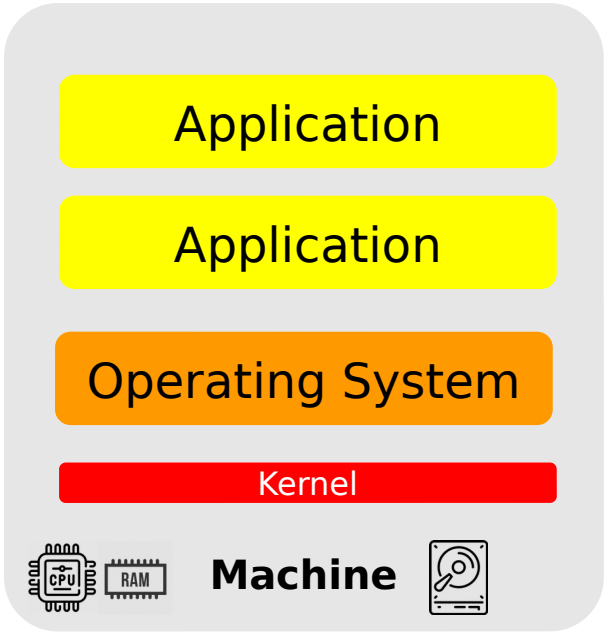
falco.org/blog/



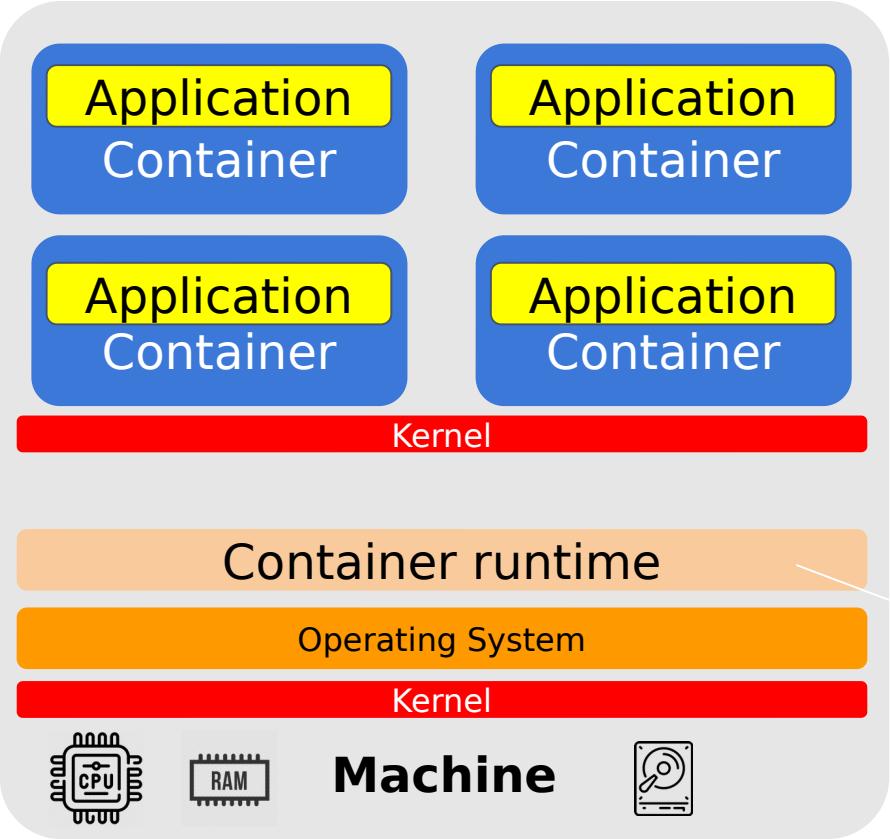
QUANTIKA 14

VM vs Machine





Containers

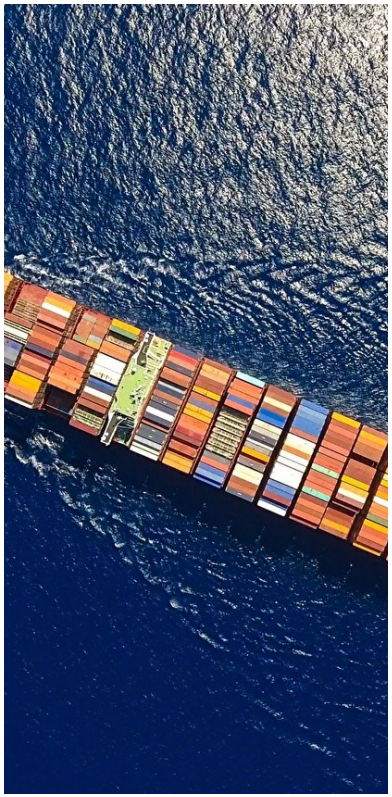


Dockerfile

Layers
Image building
Registries
Health check
Volumes
Networking

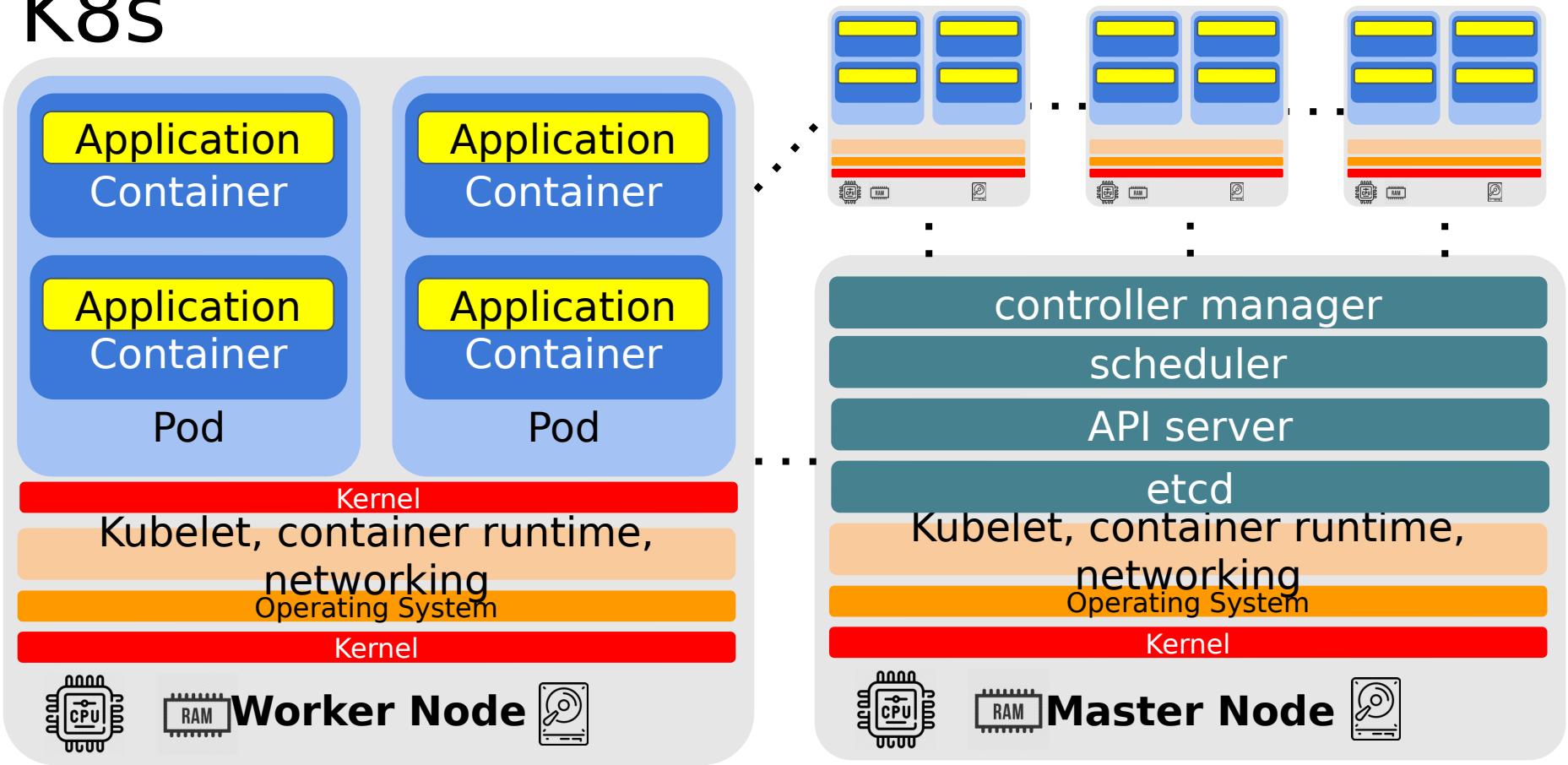
OCI specification

Docker
Containerd
CRI-O
rkt
podman

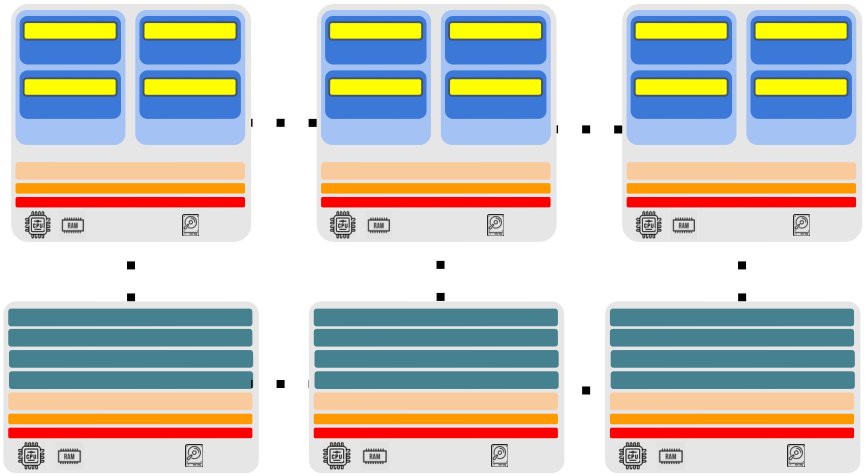


QUANTIKA 14

K8s



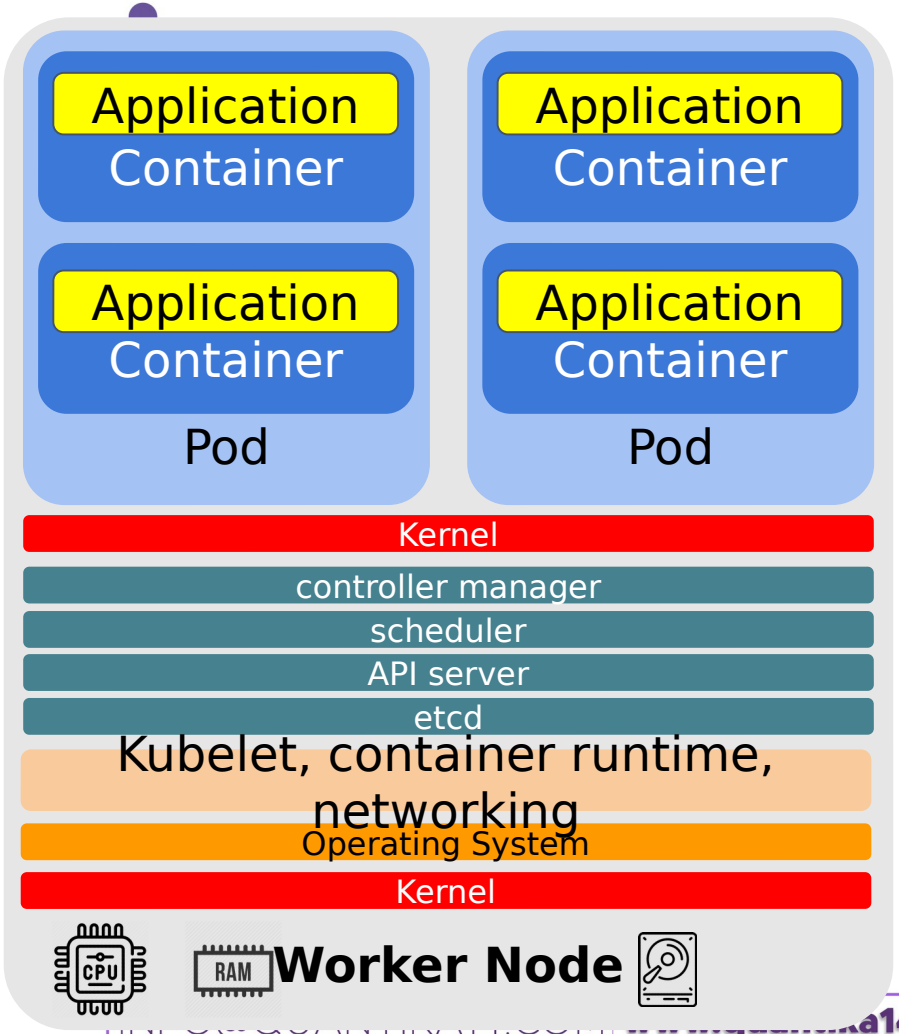
Cluster típico



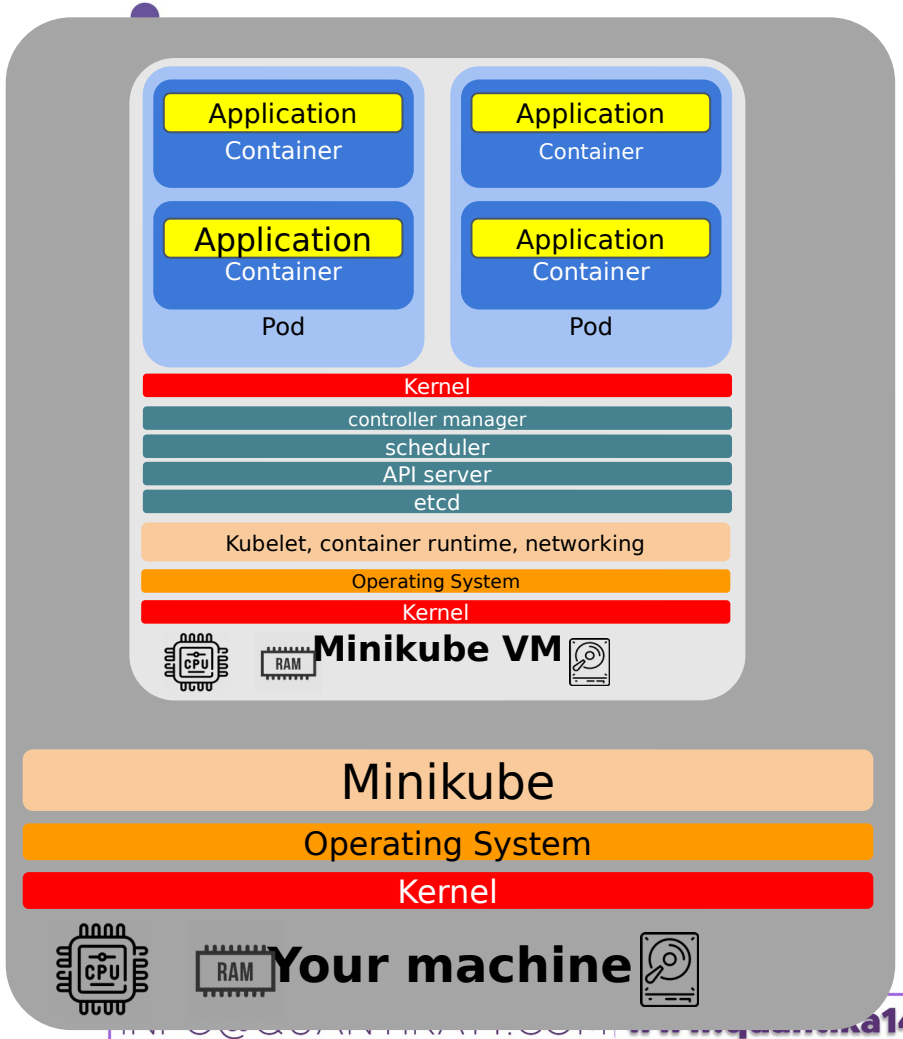
Nodos
“Worker”

CONTROL
PLANE

Número impar
de nodos
“master”



Cluster de un solo nodo



Minikube

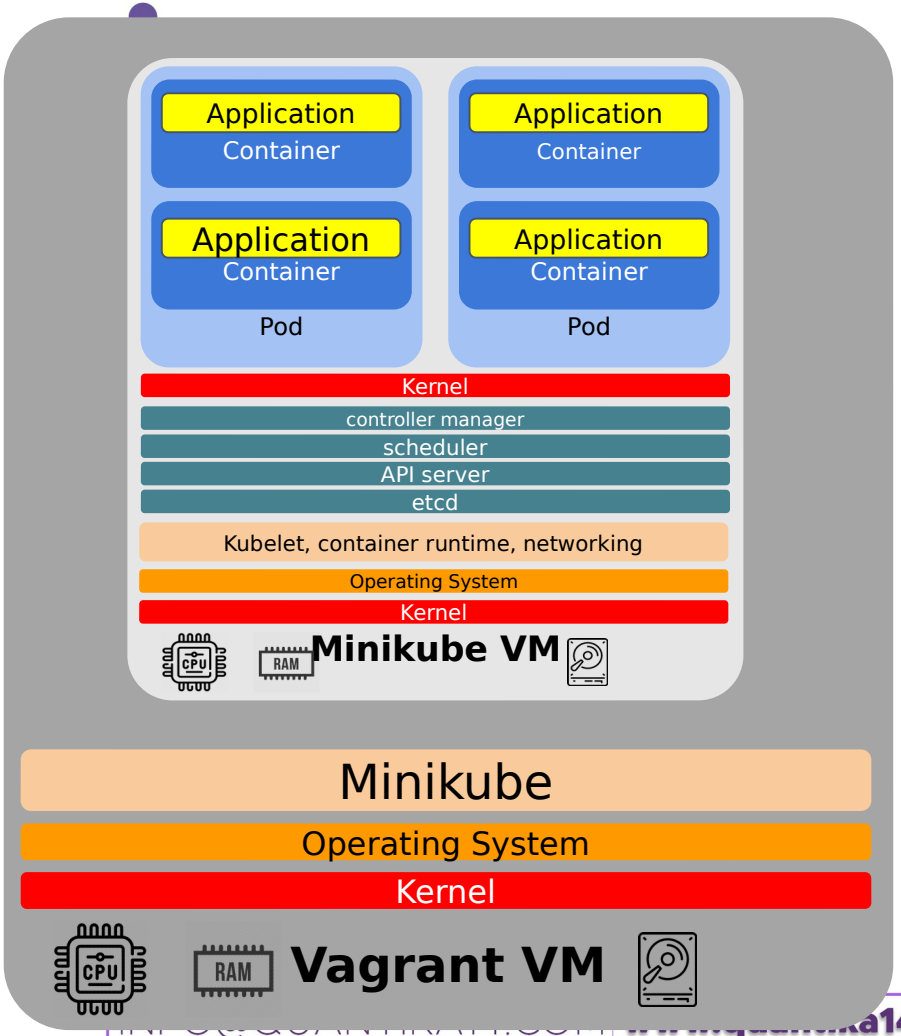
```
minikube --vm-driver=virtualbox
minikube --vm-driver=hyperkit

Accessing minikube VM:

minikube ssh
$HOME/.minikube/files
$HOME/.minikube/machines
$HOME/.minikube/cache

OS: Alpine Linux

Alternative: --vm-driver=none
(insecure)
```



Vagrant + Minikube + Docker

```
vagrant up
vagrant ssh
kubectl get nodes
```

Instalación de Falco en Kernel de cada nodo

- Instalamos Falco en cada uno de los nodos
- Utilizamos un “Daemonset” para asegurar que todos los nodos tienen Falco
- Coordinamos una única configuración y conjunto de reglas

Kubernetes Falco Events

- Monitorizamos todos los eventos de Kubernetes desde una instancia de Falco
- Configuramos Websink o auditlog
- Utilizamos un “servicio” de Kubernetes para recibir los eventos

QUANTIKA 14

Instalación de Falco en Kernel de cada nodo

- Instalamos Falco en cada uno de los nodos
- Utilizamos un “Daemonset” para asegurar que todos los nodos tienen Falco
- Coordinamos una única configuración y conjunto de reglas

Instalación usando Helm

```
$ helm repo add falcosecurity https://falcosecurity.github.io/charts
$ helm repo update
$ helm install falcosecurity/falco --namespace falco

# Add custom rules file and restart engine:
$ helm upgrade falco falcosecurity/falco -f rule_update_config.yaml
```

Kubernetes Audit Log

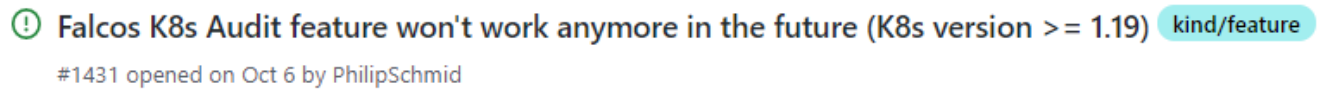
Formas alternativas de instalación

- Auditsink: Ha dejado de estar soportado en la última version de Kubernetes
- Webhook a servicio: Webserver de Falco no soporta HTTPS
- NodePort: No recomendado por seguridad
- Ingress: No hay instalación automática

Situación actual:

- Puede que deje de estar directamente soportado
- Puede que se cree un instalador con Ingress
- Puede que se incorpore como parte de Cloud Connector

Ver: https://github.com/falcosecurity/falco/issues/1431



Algunas reglas de Falco para K8s

https://github.com/falcosecurity/falco/blob/master/rules/k8s_audit_rules.yaml

- Disallowed K8s User
- Create Disallowed Pod
- Create Privileged Pod
- Create Sensitive Mount Pod
- Create HostNetwork Pod
- Create NodePort Service
- Create/Modify Configmap With Private Credentials
- Attach/Exec Pod
- EphemeralContainers Created
- Create Disallowed Namespace
- Pod Created in Kube Namespace
- Service Account Created in Kube Namespace
- Attach to cluster-admin Role
- ClusterRole With Wildcard Created
- ClusterRole With Write Privileges Created
- ClusterRole With Pod Exec Created
- Full K8s Administrative Access
- Ingress Object without TLS Certificate Created
- Untrusted Node Successfully Joined the Cluster
- Untrusted Node Unsuccessfully Tried to Join the Cluster