Continuous Security by Design

by Rob Richardson

/rob_rich

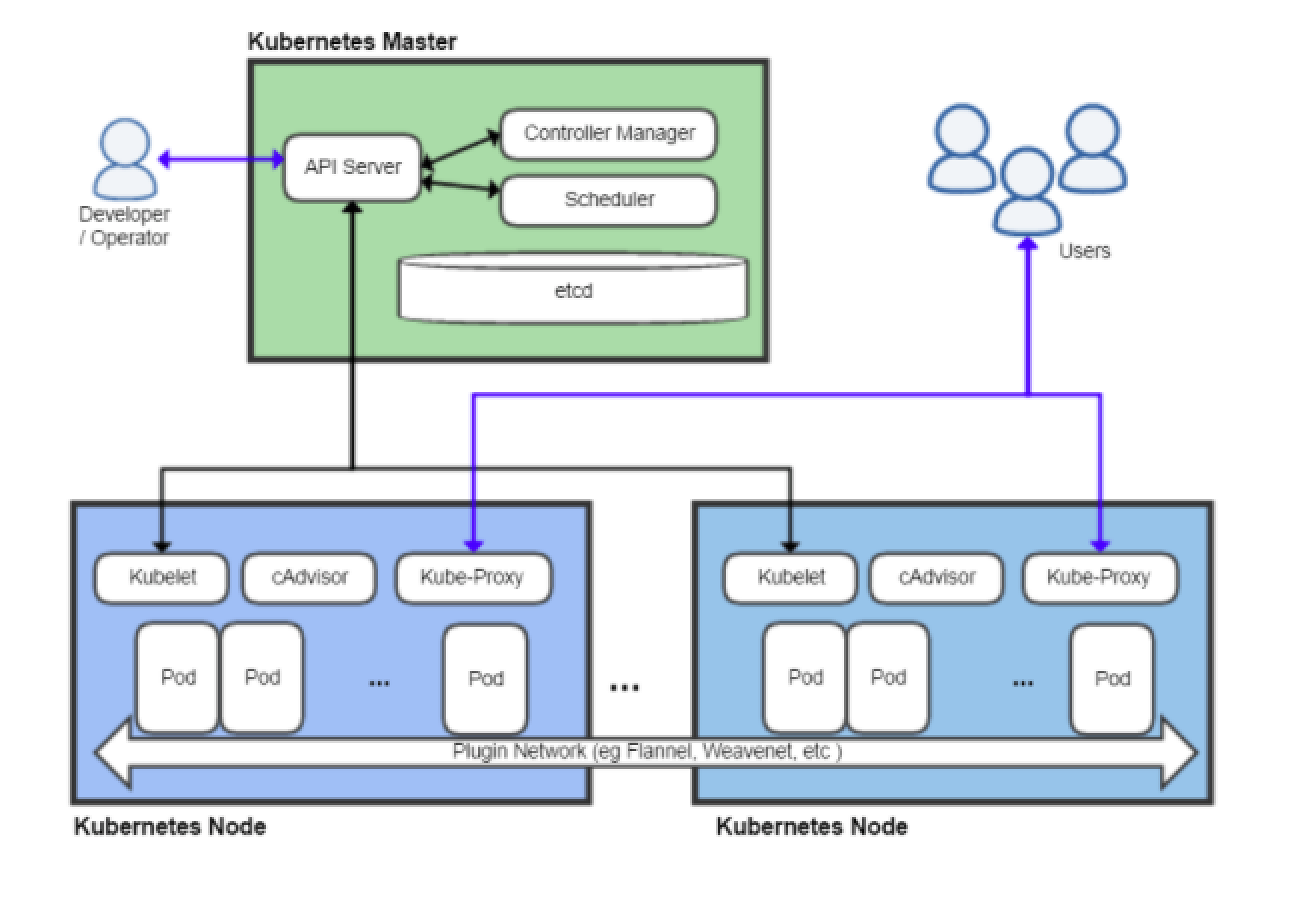
https://robrich.org/

About Me

Rob Richardson is a software craftsman building web properties in ASP.NET and Node, React and Vue. He's a frequent speaker at conferences, user groups, and community events, and a diligent teacher and student of high quality software development. You can find this and other talks on https://robrich.org/presentations and follow him on twitter at @rob_rich.

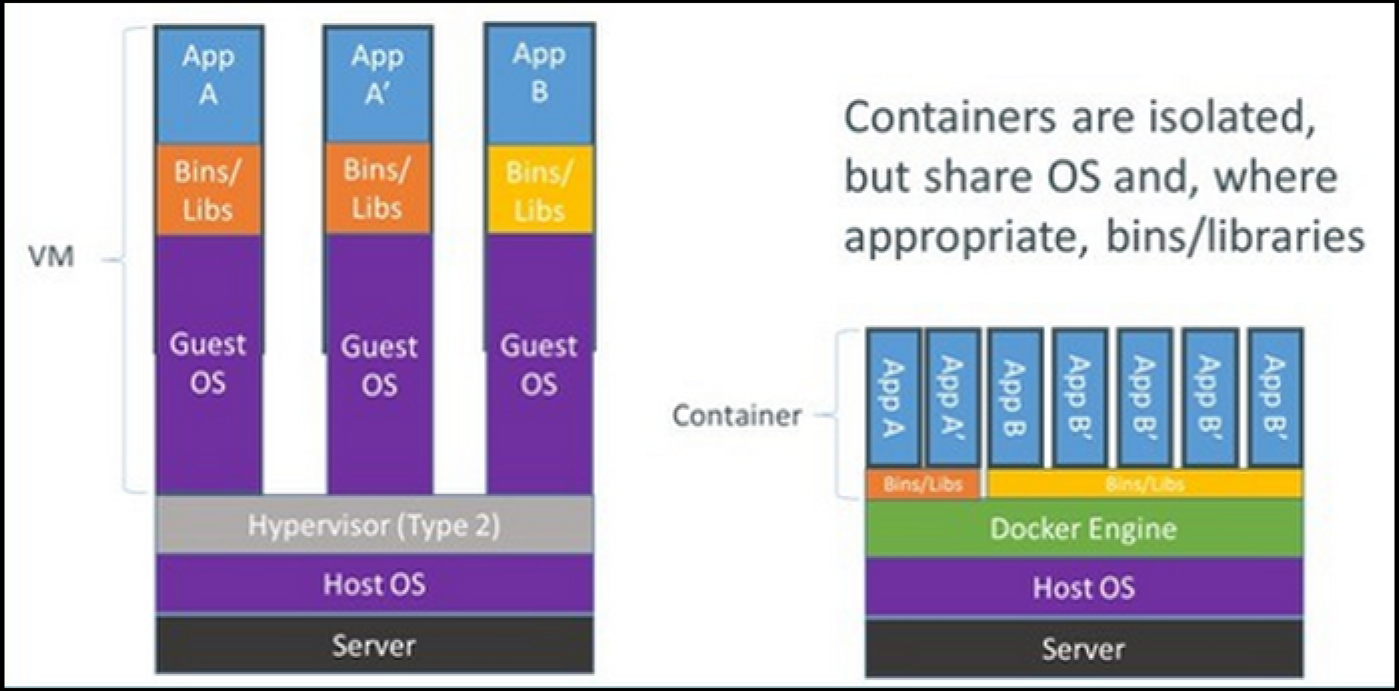
Doesn't Kubernetes just do this for me?

What is Kubernetes?



source: https://vitalflux.com/quick-glance-at-kubernetes-architectural-building-blocks/

Containers vs VMs



Containers are isolated, but share OS and, where appropriate, bins/libraries

Source: http://www.zdnet.com/article/what-is-docker-and-why-is-it-so-darn-popular/

What is Kubernetes?

- a cluster of machines
- a firewall in front of the container

Kubernetes can't secure the process in the containers

What is a Docker container?

- has file system
- has users
- has a process
- has ports

... a Linux machine

What is a Docker container?

a Linux machine except …

- ephemeral (short-lived)
- isomorphic (unchanging)
- deterministic (same every time)

Securing Containers

By default:

Every container can communicate with every other container

Securing Containers

By default:

Pop any container and you can pivot to attack all other containers from inside

Threat Vectors

- CVE in installed software
- Custom app has vulnerability
- Excessive permissions: running as root
- Exposed secrets

Attack Surface

- Exposed port(s) and content
- What can they pivot to?

Good News

Ephemeral, isomorphic hardware

If a container fails, throw it away and get a new one

Once we've scanned the image,
we know the contents;
it doesn't change

Types of Tests

1. unit and integration tests
2. static analysis of source code
3. machine inventory for known vulnerabilities
4. open-source license compliance
5. policy validation

Unit and Integration Tests

Just do it.

Static Code Analysis

- Sonarqube
- YASCA
- Veracode

License Analysis

- WhiteSource
- BlackDuck

Policy Validation

Review the results of the tests

Compare against corporate policy

go - no-go

Temperature Check:

All of this is pretty standard stuff.

We do this with machine-install software.

So where's the serverless?

Container Scanning

Continer build created new isomorphic hardware.

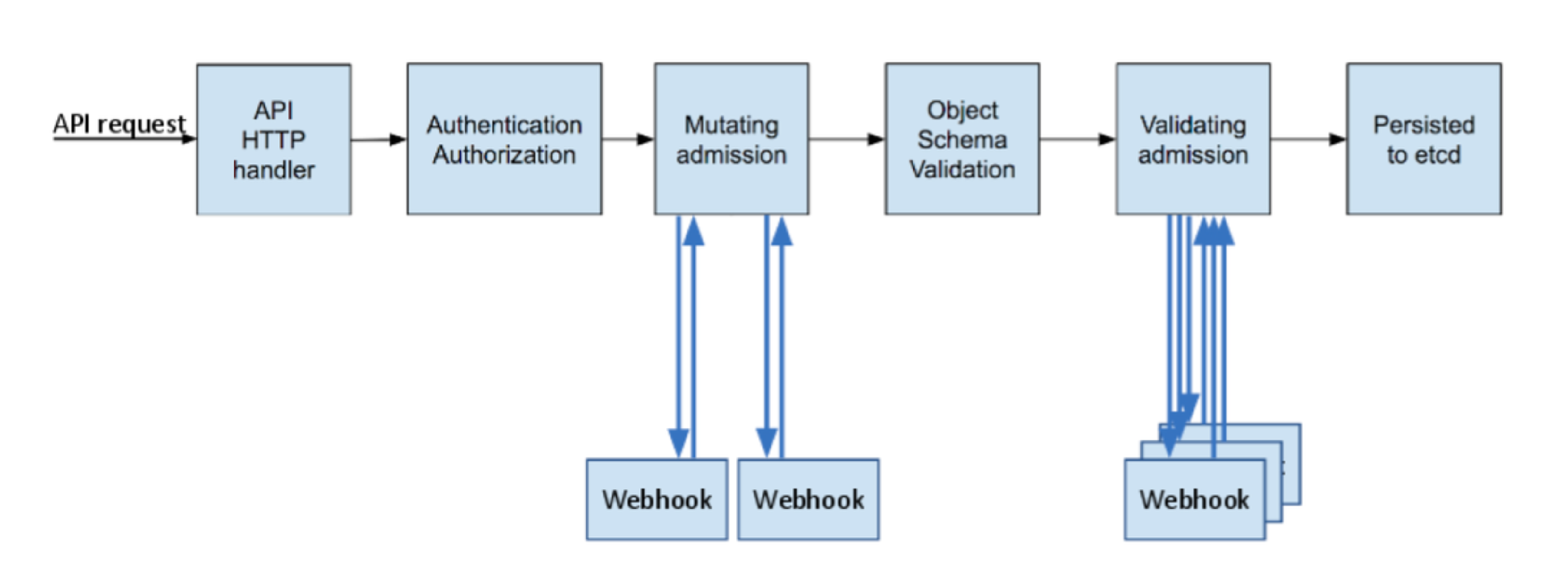
Let's catalog this hardware:

- Installed software
- Known vulnerabilities

When to Scan

- Incorporate into the build pipeline
  - if we assume nothing gets around it into production
- Scan the container registry
- Kubernetes webhooks during pod launch
- Scan the production cluster

Kubernetes Webhooks



source: https://kubernetes.io/blog/2019/03/21/a-guide-to-kubernetes-admission-controllers/

Choosing a Container Scanner

- Anchore
- Clair
- Dagda
- OpenSCAP
- Sysdig Falco
- AquaSec

Anchore

Anchore

- Free and open source
- Runs as microservice containers
- Software inventory
  - both OS packages and app packages
- Container scanning for CVEs
- It is not fast
- Docs are not great

Anchore docs

1. Download docker-compose.yaml from
   https://docs.anchore.com/current/docs/engine/quickstart/do
   compose.yaml

2. `docker-compose up`

3. `docker-compose exec api bash` or
   `pip install anchorecli`

4. Run commands

Anchore docs

```
anchore-cli system status
anchore-cli system feeds list
```

"it may take 10 minutes to populate all the scan data"

(It took me a week.)

Anchore docs

```
# 1. Add the container for analysis:
anchore-cli image add mycontainer:latest
# 2. Wait for scan:
anchore-cli image wait myimage:latest
# 3. Get summary:
anchore-cli image get myimage:latest
# 4. Get scan results: os, npm, gem, etc
anchore-cli image vuln myimage:latest all
# 5. List installed packages:
anchore-cli image content myimage:latest all
```

Anchore with GitHub Actions

```
- name: Anchore scan
  uses: anchore/scan-action@1.0.6
  with:
    image-reference: myimage:latest
    dockerfile-path: Dockerfile
    include-app-packages: true
    fail-build: true

- name: Show Anchore results
  run: for j in `ls ./anchore-reports/*.json`; do echo "---- ${j} -
  if: ${{ always() }}
```

DEMO

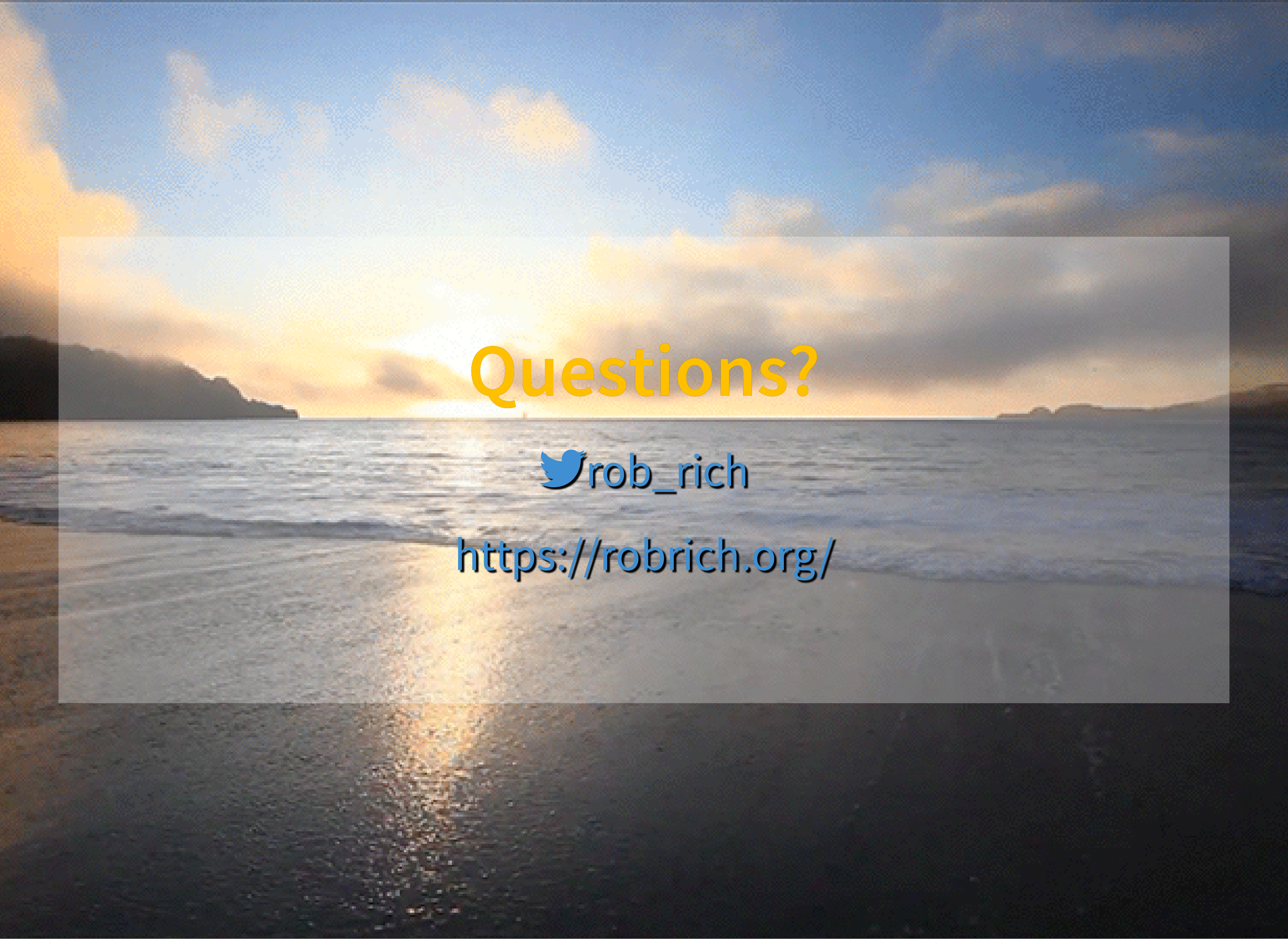
Anchore container scanning

GitHub test pass demo | GitHub test fail demo

Kubernetes only protects itself.

We secure the containers.

Use container scanning.



Questions?

rob_rich

https://robrich.org/