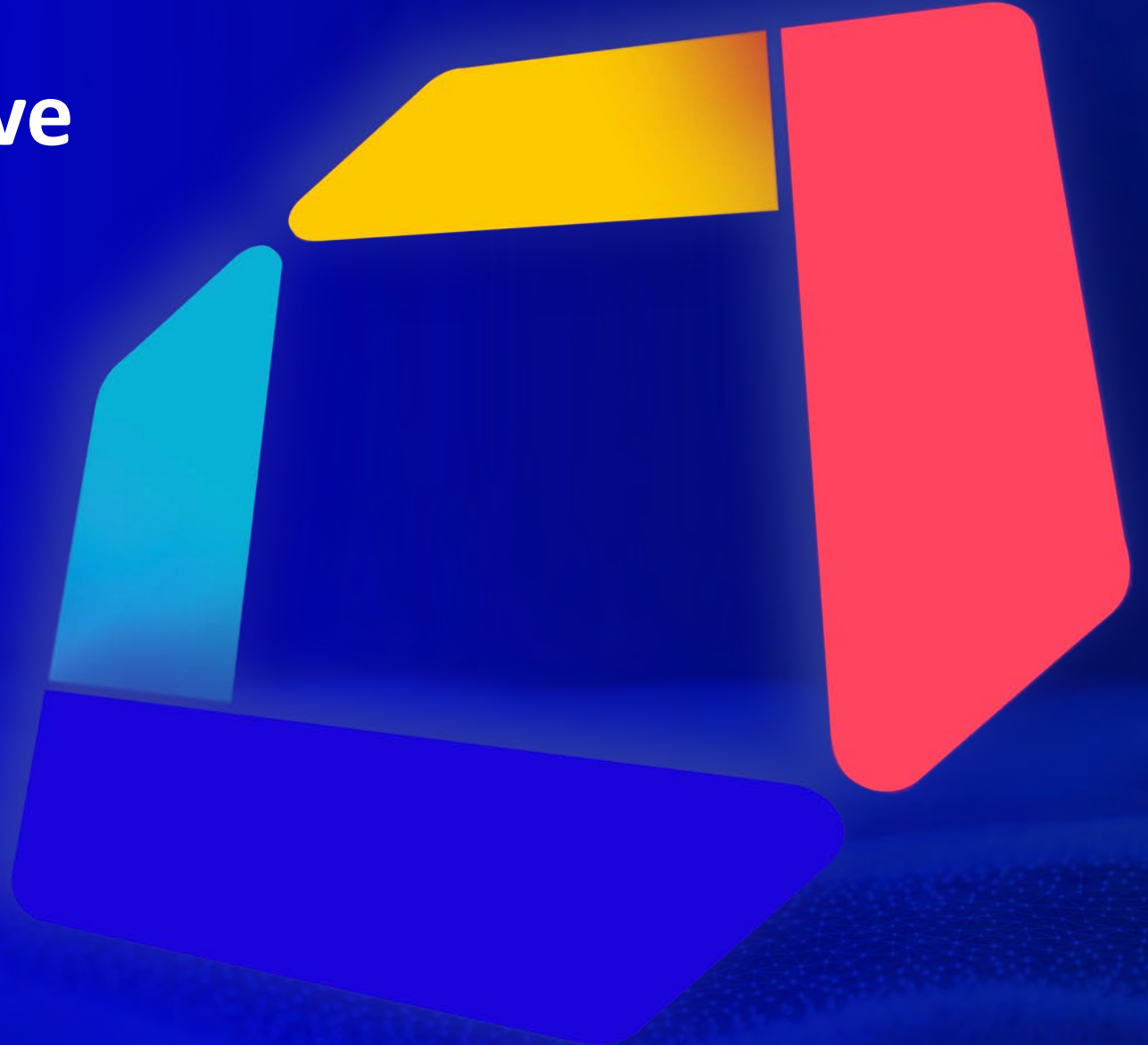




# Integrating Cloud Native Security into the SRE culture

Anais Urlichs



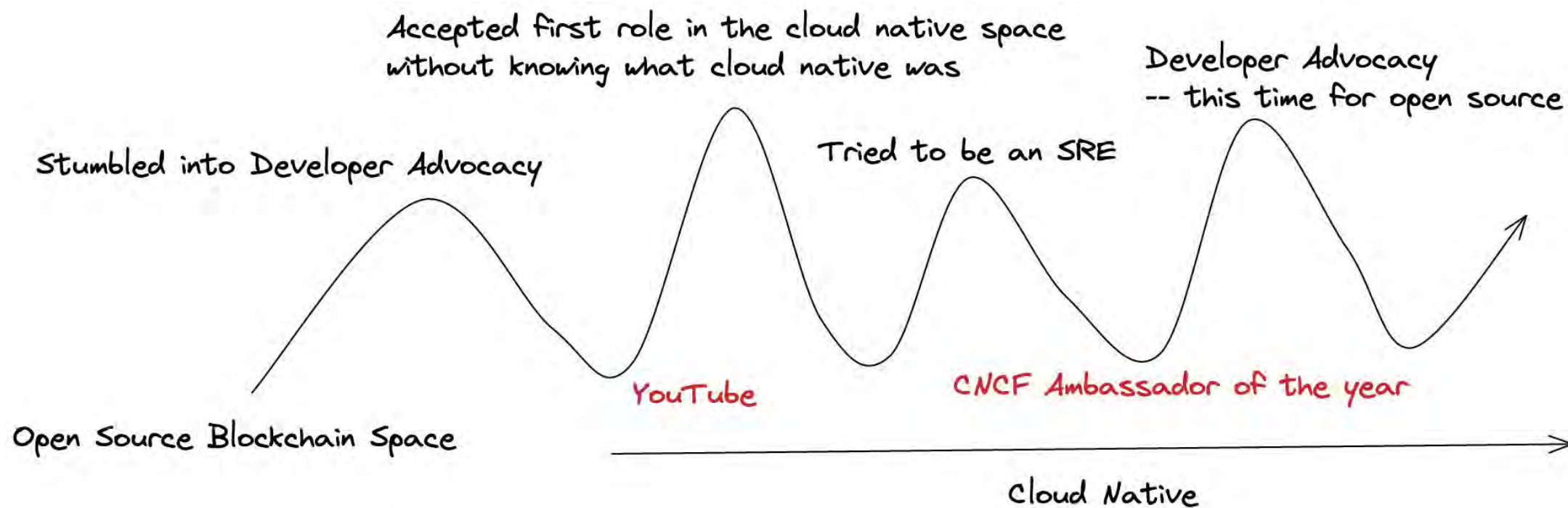
# Topic

Site Reliability Engineering



Cloud Native Security

# Who am I?



# YouTube

Search

23 853 50m 48h

## New Videos Weekly

### #100DaysOfKubernetes

urlichsanais

AnaisUrlich

anaisurl.com

**Anais Urlich**  
11.5K subscribers

CUSTOMISE CHANNEL MANAGE VIDEOS CSV EXPORT

HOME VIDEOS PLAYLISTS COMMUNITY CHANNELS ABOUT TRENDING STATS

#### Kyverno Overview -- Defining Kubernetes Cluster Policies

776 views · 4 weeks ago

This is another full tutorial on getting started with Kyverno.

Here is the blog post with all the details & using the Kyverno CLI  
<https://anaisurl.com/defining-kuberne...>

Resources  
\* Kyverno Website: [https://kyverno.io/...](https://kyverno.io/)

READ MORE

Uploads ▶ PLAY ALL

- Creating & using addons (15:32)
- The best learning platform with FREE courses (6:46)
- Kyverno Overview -- Defining Kubernetes Cluster Policies (15:00)
- How to become a Developer Advocate & My career path (12:16)
- Helm Chart Dependencies (10:10)
- lazytrivy: Scan all your container images with one... (5:59)

Writing and testing a microk8s addon on Mac (282 views · 6 days ago)

The best learning platform with FREE courses (438 views · 9 days ago)

Kyverno Overview -- Defining Kubernetes Cluster Policies (776 views · 4 weeks ago)

How to become a Developer Advocate & My career path (919 views · 1 month ago)

A deep dive into Helm Dependencies (671 views · 1 month ago)

lazytrivy: Scan all your container images with one... (427 views · 1 month ago)

# Weekly DevOps Newsletter

## DevOps

DevOps Diary Weekly Newsletter and DevOps related blog posts


---

10 OCT 2022

PUBLIC

**#69 Seven-Day DevOps — Weekly DevOps Newsletter**

Weekly Newsletter with pupdates, tutorials, videos, fun memes and more!




---

5 OCT 2022

PUBLIC

**Writing a microk8s addon on Mac**

In this blog post, I first describe the steps I took to install microk8s and then to develop and test a microk8s addon for the Trivy...




---

22 SEP 2022

PUBLIC

**#68 Seven-Day DevOps — Weekly DevOps Newsletter**

Some exciting updates, great content incl. tutorials, talks and more events. Enjoy!




---

8 SEP 2022

PUBLIC

**#67 Seven-Day DevOps — Weekly DevOps Newsletter**

This week's newsletter is featuring lots of content on AWS, including tutorials, tweet-threads and podcasts – among other excitin...





# Supercluster design: compute



KubeCon



CloudNativeCon

North America 2021

## Regional Hyper Converged Infrastructure



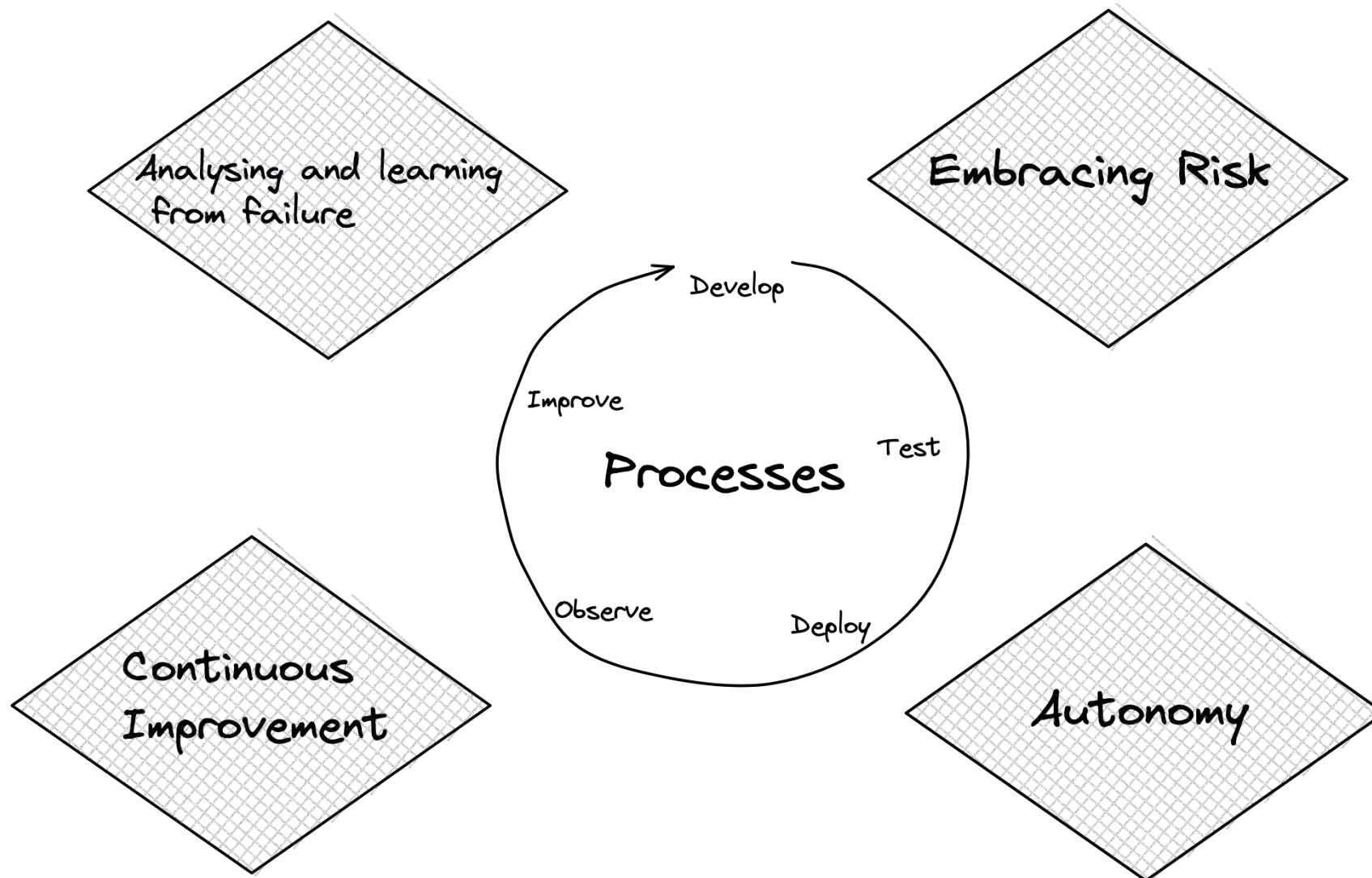
2xIntel Gold Xeon 20 core CPU  
384G,B 2666Mhz DRAM  
100 Gb Networking



**OPEN**  
Compute Project®

@urlichsanais

# SRE Culture – What is it?





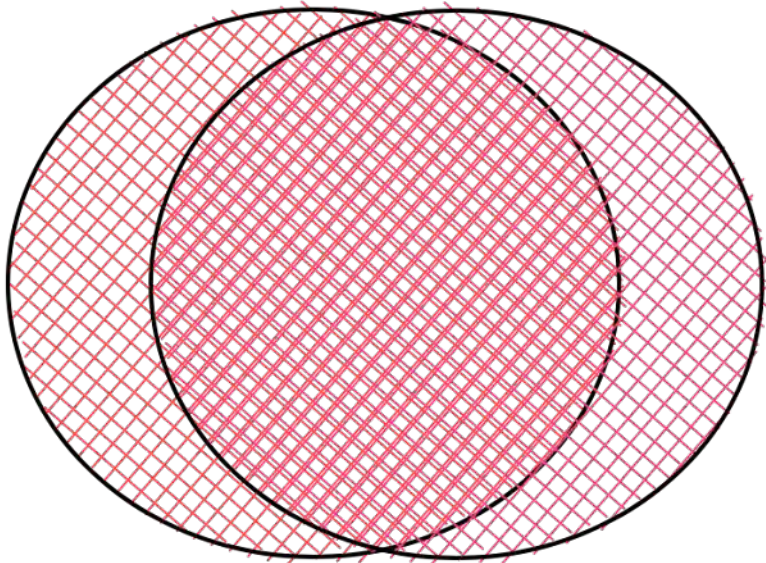
# DevSecOps/Cloud Native Security

## DevSecOps

Incorporating security into all other business functions  
by empowering people and creating accountability

# The premise

Security Practices  
&  
SRE Practices

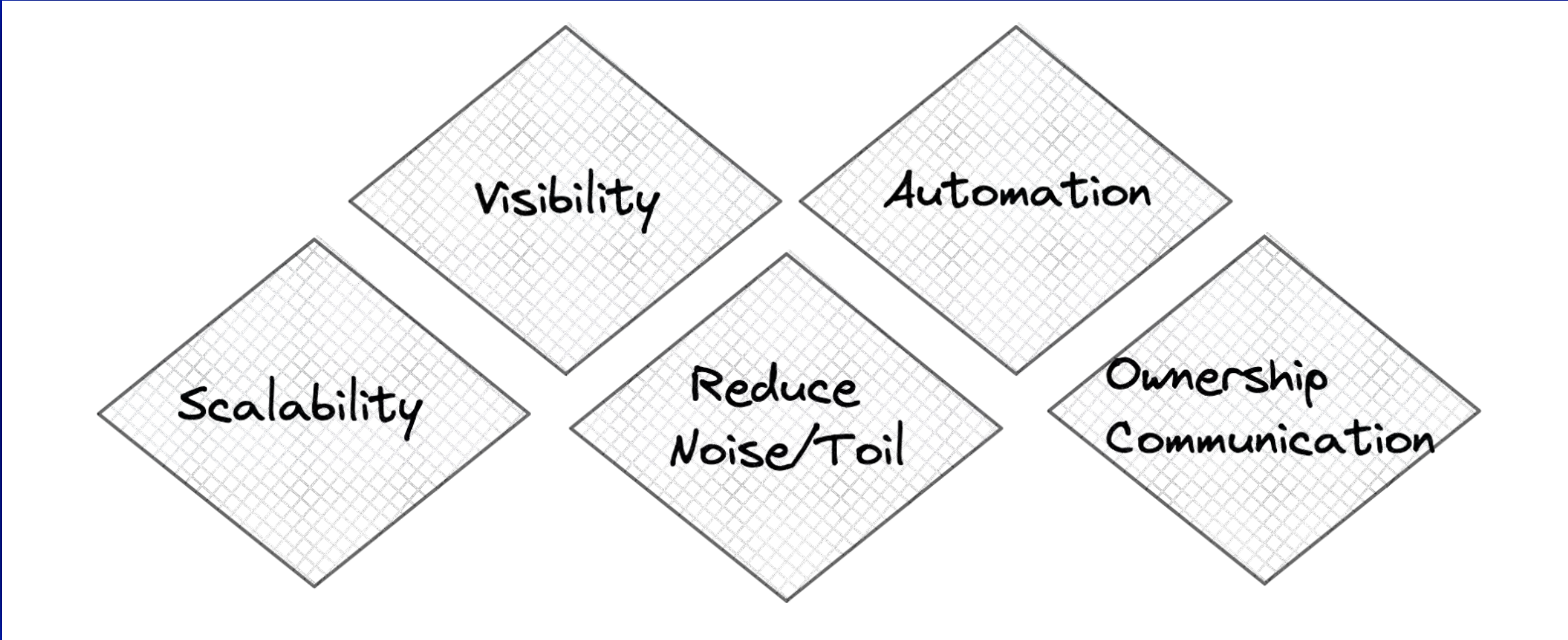


Healthy Services are Secure Services



Define what healthy services look like

# SRE Goals are Security Goals

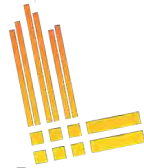


## Practices

- Investing in runbooks and documentation
- Developing a robust delivery pipeline
- Define ownership
- Set explicit expectations on people
- Active communication with engineers
- Define procedures/processes
- Define use and expectations on tools

# Tools

## Observability



Grafana loki



## Management



CI/CD

# Where are the security tools?

# 10 Steps to integrating cloud native security





aqua  
trivy

## Scan Targets

---

Git Repository & Filesystem Scanning

---

Container Image Scanning

---

Dockerfile Scanning

---

laC Scanning

---

Kubernetes Manifest Scanning

---

SBOM Generation & Scanning

---

AWS Account Scanning

---

In-cluster Scanning

---



# Step 1 Understanding your need

# Factors



Size of your team



Industry you are working in



Type of technologies you are working with



Company Goals and leadership



Budget and expertise



INSTALLATION



KUBERNETES  
RESOURCE TYPE



SCAN COVERAGE



INTEGRATION



FOCUS

# Needs – Based on Wise Engineering Blog post

- 1. Assign ownership of vulnerabilities**
- 2. Global view of the security state of services**
- 3. Develop Dashboards for different users and requirements**
- 4. Overcome difficult to use different UIs**

Note that these needs have been rewritten based on the following blog post <https://medium.com/wise-engineering/our-application-security-journey-part-1-fb7d449a7126>

# Step 2 Choosing a cloud native Security Scanner

## Open Source Security Scanning -- focus on cloud native

### Vulnerability Scans

Trivy

clair

Grype

KubeClarity

### IaC Misconfiguration Scans

Trivy

tfsec

terrascan

KICS

checkov

Conftest

Kubescape

regula

### Policy Checks & Schema validation\*

Trivy

cloudQuery

Cloud Custodian

\*a little vague since multiple tools that implement Rego etc. allow for policy checks

### Compliance Scans

kube-bench CIS

kube-beacon CIS

chain-bench CIS

Starboard NSA

Kubescape NSA

### SBOM

Trivy

Syft

cloudQuery

KubeClarity

### In-Cluster Scans

Trivy

Kubescape

### Kubernetes Pentesting

kube-hunter

kdigger

StackRox

### Secret Scanning

tfsec (terraform)

Trivy (K8s)

many-more non cloud native



aqua  
trivy

## Scan Targets

---

Git Repository & Filesystem Scanning

---

Container Image Scanning

---

Dockerfile Scanning

---

laC Scanning

---

Kubernetes Manifest Scanning

---

SBOM Generation & Scanning

---

AWS Account Scanning

---

In-cluster Scanning

# Step 3 Setting it up & Making sure everything is running properly

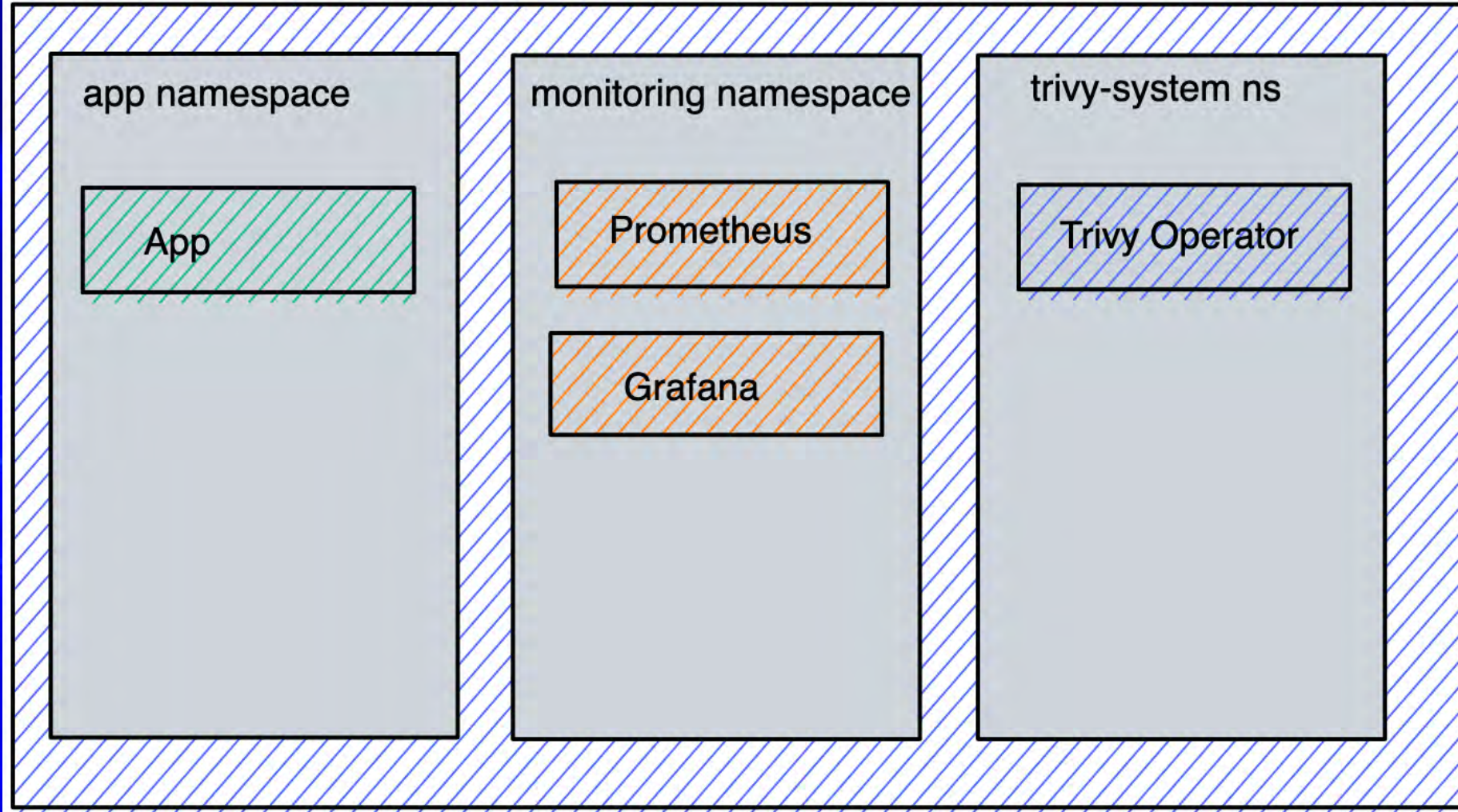


1. Identifying the best installation option
2. Deciding upon the configuration
3. Testing custom configuration
4. Ensuring everything is working together

```
▼ MONITOR-SECURITY
  ▼ app-manifests
    ! deployment.yaml
    ! service.yaml
  > assets
  > dashboards
  ▼ observability-conf
    ! prom-values.yaml
    ! promtail-values.yaml
    ! tracee.yaml
    ! trivy-service-monit...
  > README.md
```



## Kubernetes Cluster



If everything is a Kubernetes resource, you can use the same processes across your stack



```
> kubectl get all -n trivy-system
```

NAME	READY	STATUS	RESTARTS	AGE
pod/trivy-operator-8556cdf857-snhdj	1/1	Running	0	4m56s

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
service/trivy-operator	ClusterIP	None	<none>	80/TCP	4m56s

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
deployment.apps/trivy-operator	1/1	1	1	4m57s

NAME	DESIRED	CURRENT	READY	AGE
replicaset.apps/trivy-operator-8556cdf857	1	1	1	4m57s

```
return go(f, seed, [])  
}
```












# Step 4 Setting up a dashboard



The dashboard can be found in the following repo: <https://grafana.com/grafana/dashboards/16652-trivy-operator-reports/>

# Step 5 Avoiding Vulnerability Hell

 **Alex**   
@AlexJonesax

I'll just give up and die now then

**554**  
known vulnerabilities

3	C	132	H	117	M	302	L
---	---	-----	---	-----	---	-----	---

11:38 AM · Sep 14, 2022 · Twitter Web App

Tweet <https://twitter.com/AlexJonesax/status/1569998923955142657>

## Some possible strategies

- 1. Ignore all but Critical Vulnerabilities**
- 2. Don't scan everything at once**
- 3. Filter by Vulnerabilities with known-fixtures**
- 4. Filter Vulnerabilities by team & by application**
- 5. Make the Vulnerabilities context-specific**

# Step 6 What are metrics without alerts

State	Name	Health	Summary
Normal	Critical Vulnerability	ok	

[Silence](#) [Show state history](#) [View](#) [Edit](#) [Delete](#)

Evaluate: Every 1m  
For: 5m  
Data source: Prometheus

Matching instances: Search by label  State: Normal 1, Alerting, Pending, NoData, Error

State	Labels	Created
Normal	alertname=Critical Vulnerability grafana_folder=test	-

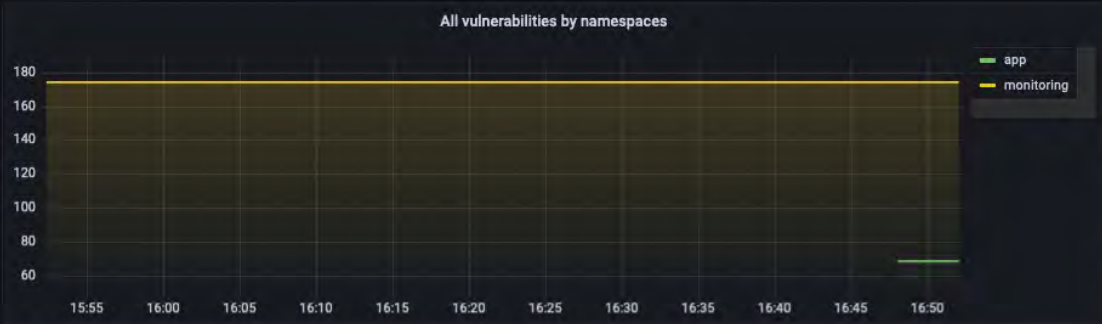
It's easy to ignore vulnerabilities –  
give them a "voice"



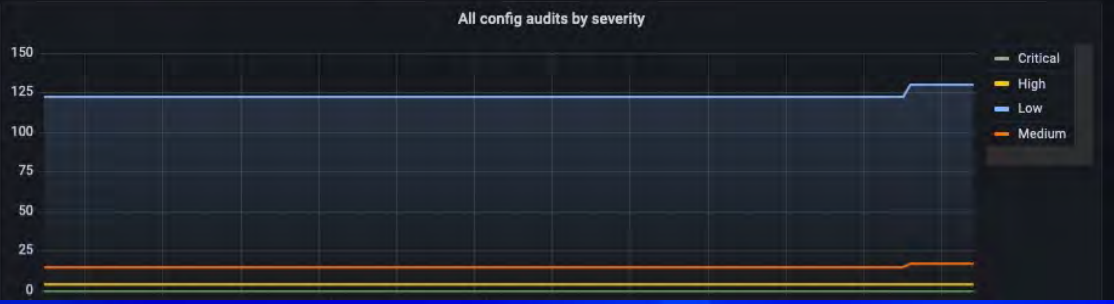
# Step 7 Correlating Metrics

datasource Prometheus Cluster All namespace All

Summary of vulnerabilities



Summary Resource Config Audit



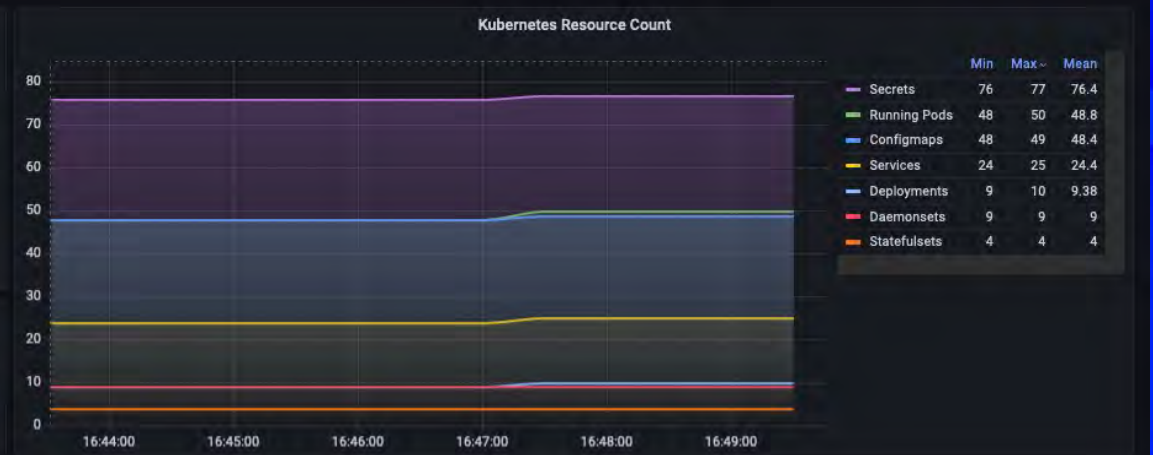
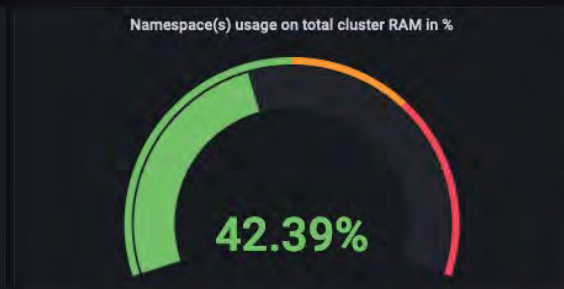
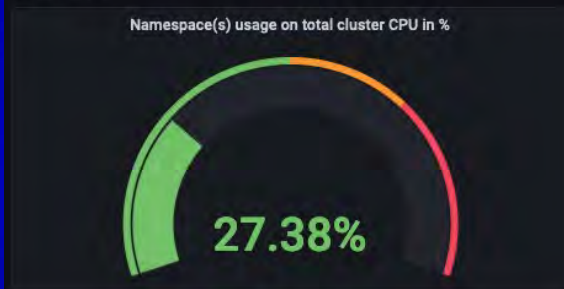


General / Kubernetes / Views / Namespaces ☆ 🔊

2022-10-11 16:43:31 to 2022-10-11 16:49:46

datasource Prometheus namespace All resolution 30s

Overview



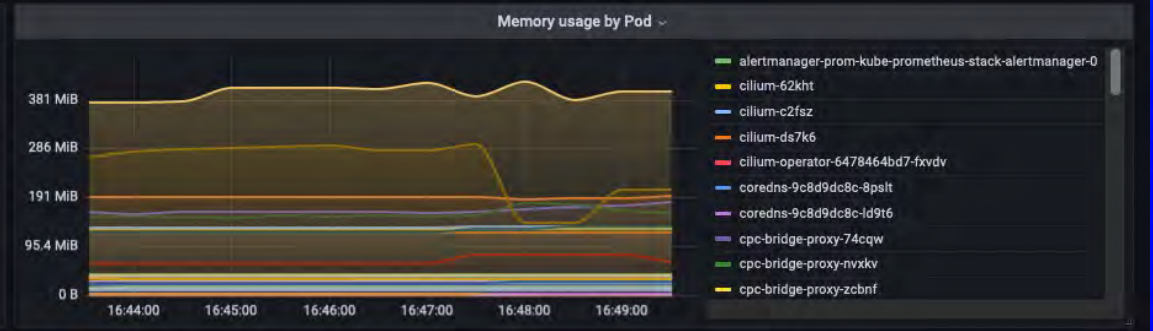
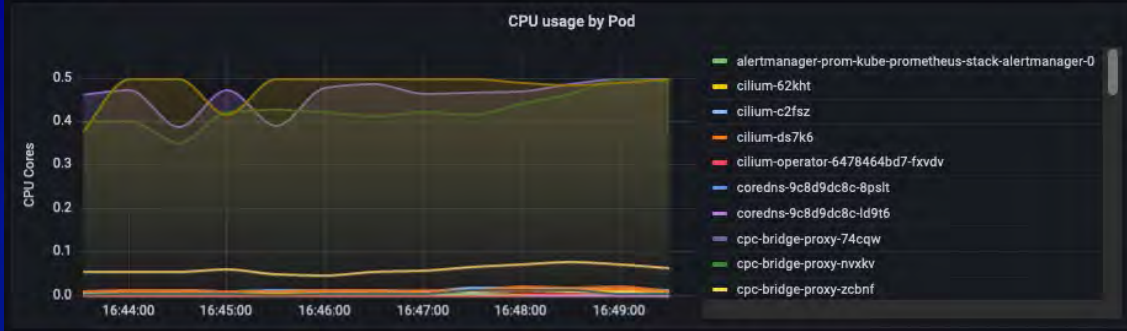
#### Namespace(s) CPU Usage in cores

Real	Requests	Limits	Cluster Total
<b>3.03</b>	<b>3.68</b>	<b>2.93</b>	<b>6</b>

#### Namespace(s) RAM Usage in bytes

Real	Requests	Limits	Cluster Total
<b>4.95 GiB</b>	<b>2.98 GiB</b>	<b>2.91 GiB</b>	<b>11.5 GiB</b>

Resources



# Step 8 Some additional tips

## Just a few more things to think about...

- **Assign Ownership**
- **Don't introduce "too many" new tools at once**
- **Utilise existing workflows, platforms and processes**

# Step 9 Optimise based on what works for your team

**The initial setup might be the same but everything else will be different**

# Step 10 Don't stop at security scanning





<https://github.com/aquasecurity/tracee>



# Additional Resources

- [Our Application Security Journey \(Part 1\) by Wise Engineering](#)
- [The Aqua Open Source YouTube Channel](#)
- [The Trivy GitHub Repository](#) and the [Trivy Operator Repository](#)
- [The demo project on GitHub](#)

& You can [find us on Slack](#): [slack.aquasec.com](https://slack.aquasec.com)

# Questions



*@urlichsanais*

# Thanks



*@urlichsanais*