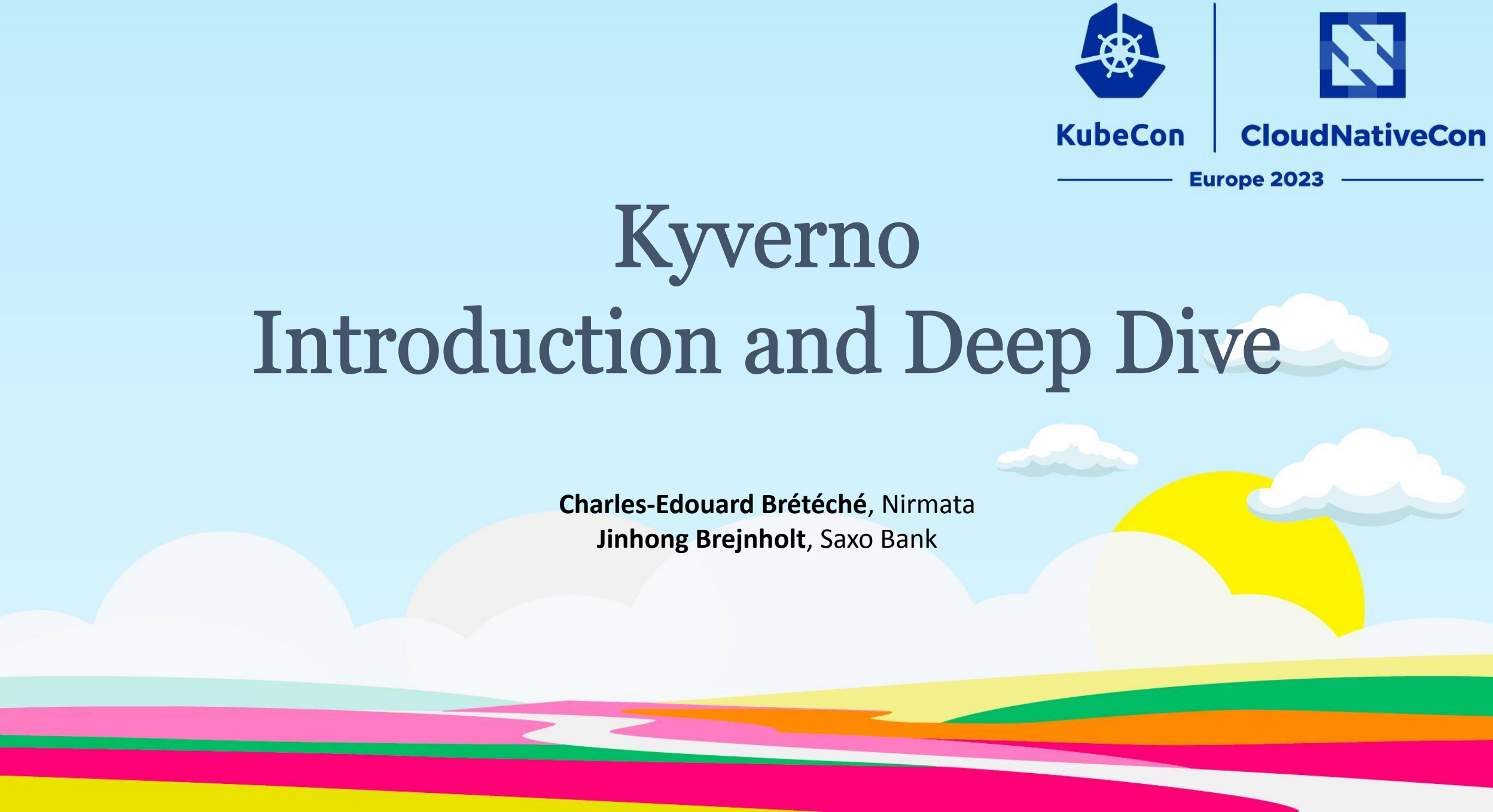


Kyverno
Introduction and Deep Dive

Charles-Edouard Brétéché, Nirmata
Jinhong Brejnholt, Saxo Bank

KubeCon | CloudNativeCon

Europe 2023

SAXO

BE INVESTED

Jinhong Brejnholt

Chief Cloud Architect &
Product Owner @ Saxo Bank

Cloud Native Copenhagen
Organizer

jbrejnholt

nirmata

Charles-Edouard Brétéché

Staff engineer @ Nirmata

Kyverno maintainer

eddycharly

Agenda

- **Intro to Kyverno**
  - Why do we need a policy engine ?
  - What is Kyverno and how does it work ?
  - More than an admission webhook
- **Saxo Bank & VELUX Use cases**
  - Why Kyverno
  - How
  - Demo
  - Learnings
- **Advanced features**
  - Notary V2
  - Internal service API calls
  - Validating Admission Policies

Intro to Kyverno

Why do we need a policy engine ?

Security

Multi tenancy

Collaboration

Cost control

Plenty of other things too…

What is Kyverno ?

Open-source

Kubernetes native

Easy to use

Declarative

GitOps friendly

Kyverno policy types

Validation

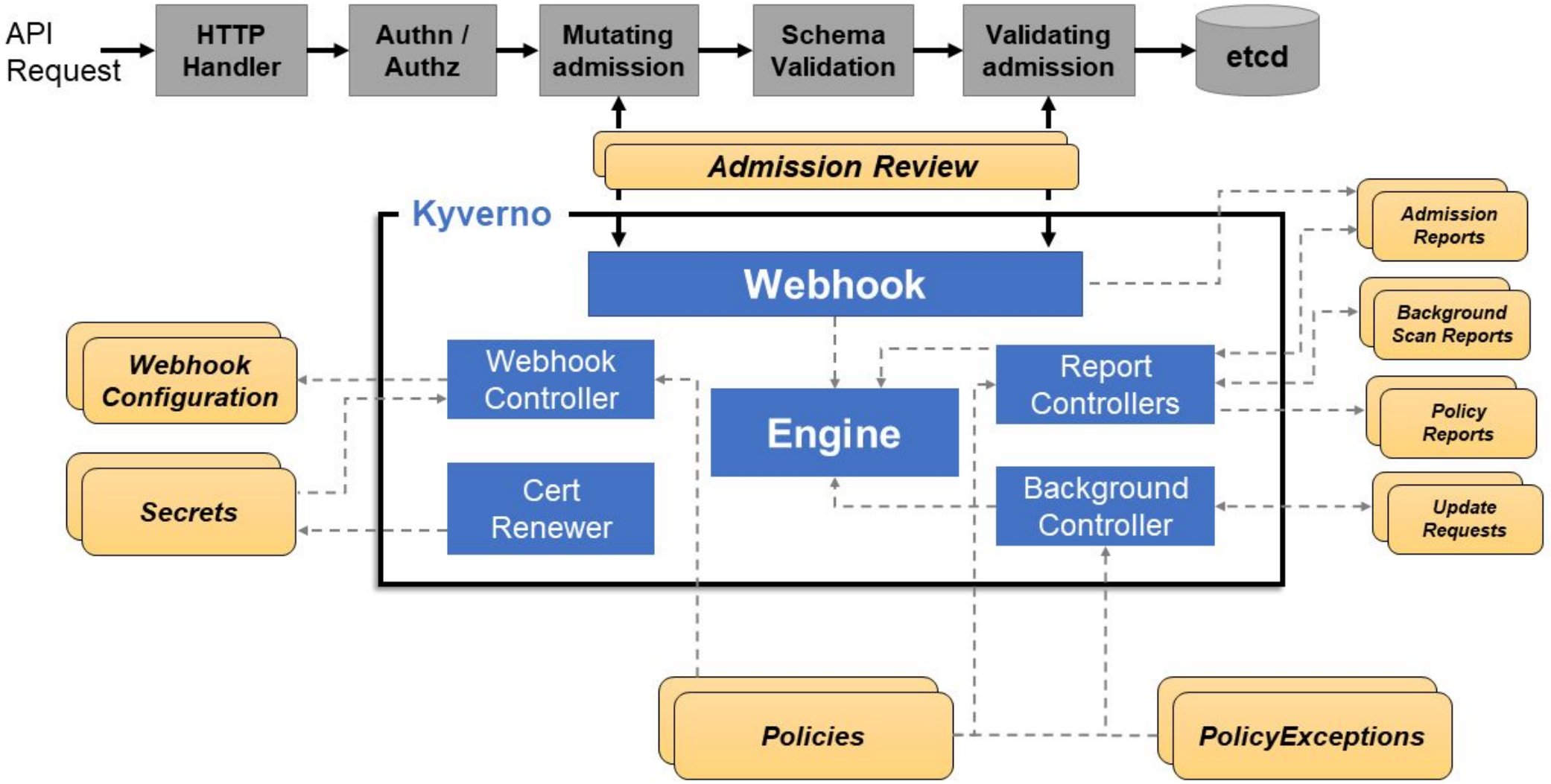
Generation

Image verification

Mutation

Cleanup

Kyverno architecture



Anatomy of a Policy


```yaml
apiVersion: kyverno.io/v1
kind: ClusterPolicy
metadata:
 name: require-labels
spec:
 validationFailureAction: Enforce
 background: true
 rules:
   - name: require-team
     match:
       any:
         - resources:
             kinds:
               - Pod
     validate:
       message: 'The label `team` is required.'
       pattern:
         metadata:
           labels:
             team: '?*'
```

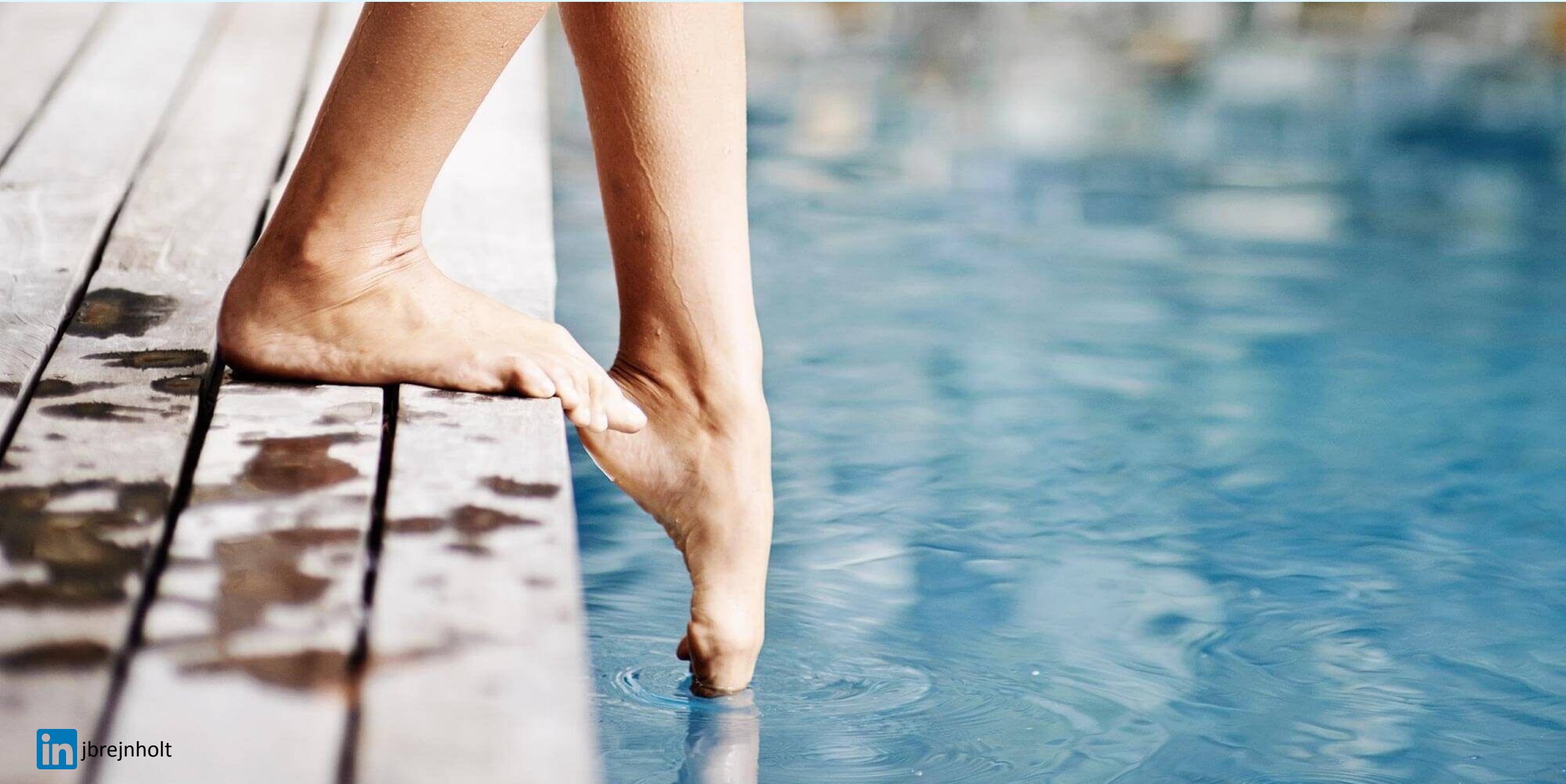
From policy definition to live enforcement

DEMO TIME

More than an admission webhook

Beside Kyverno being an admission webhook it can do a lot more !

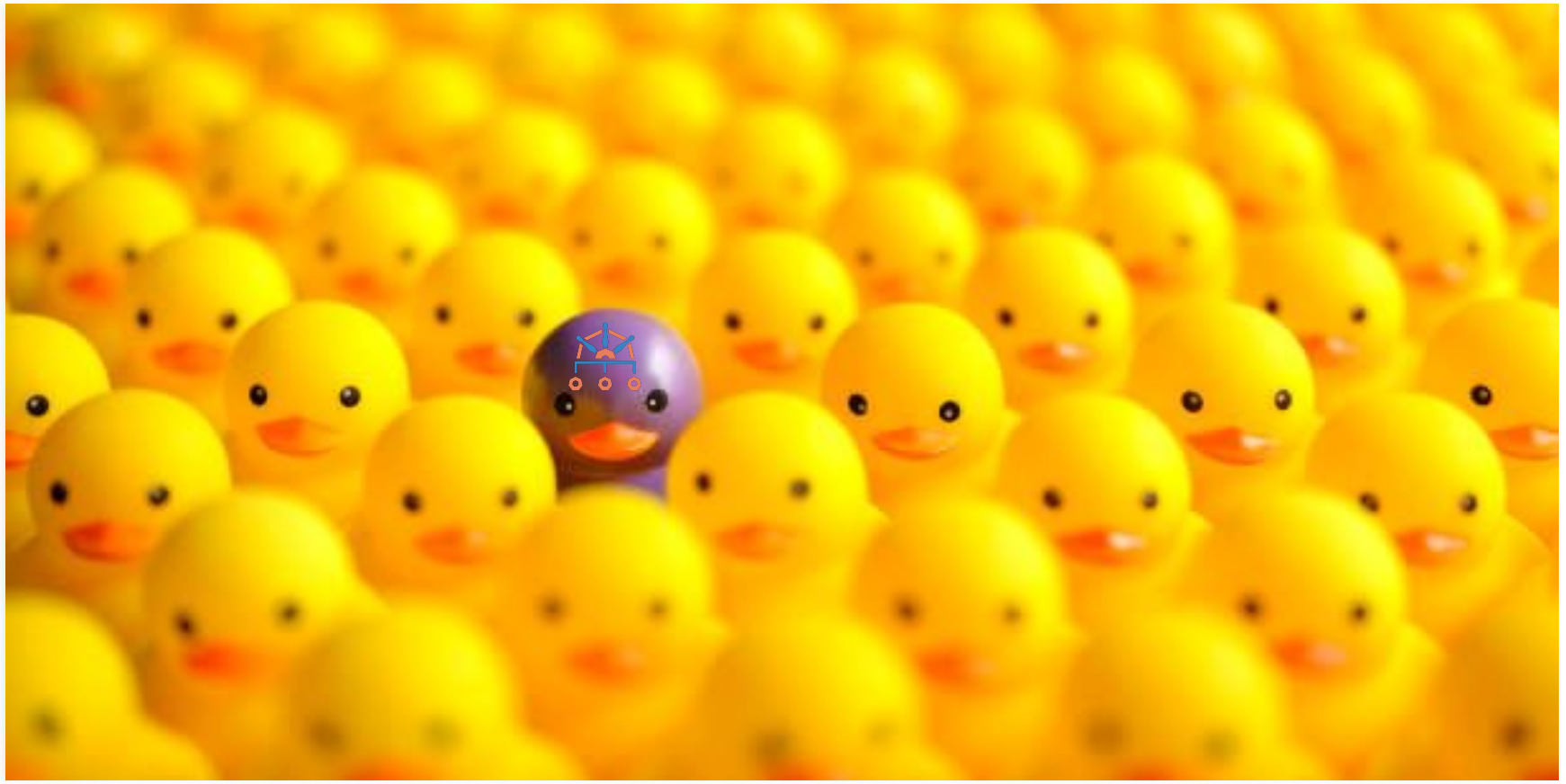
- Generate **reports** in the background
  - For all resources, not just the ones created after a policy was installed
- Create **events** corresponding to detected violations
  - When running in the background
  - At admission time, when a resource is blocked
- Run **offline** with Kyverno CLI
  - The Kyverno CLI can run against file manifest with or without a running cluster
  - Perfect to evaluate manifests in your CI pipelines
- **Visualise** violations in real time with policy reporter
  - By installing the policy reporter optional component
- Large policy catalog available
  - Accelerate Kyverno adoption



At Saxo Bank, we do!

VELUX®

jbrejnholt



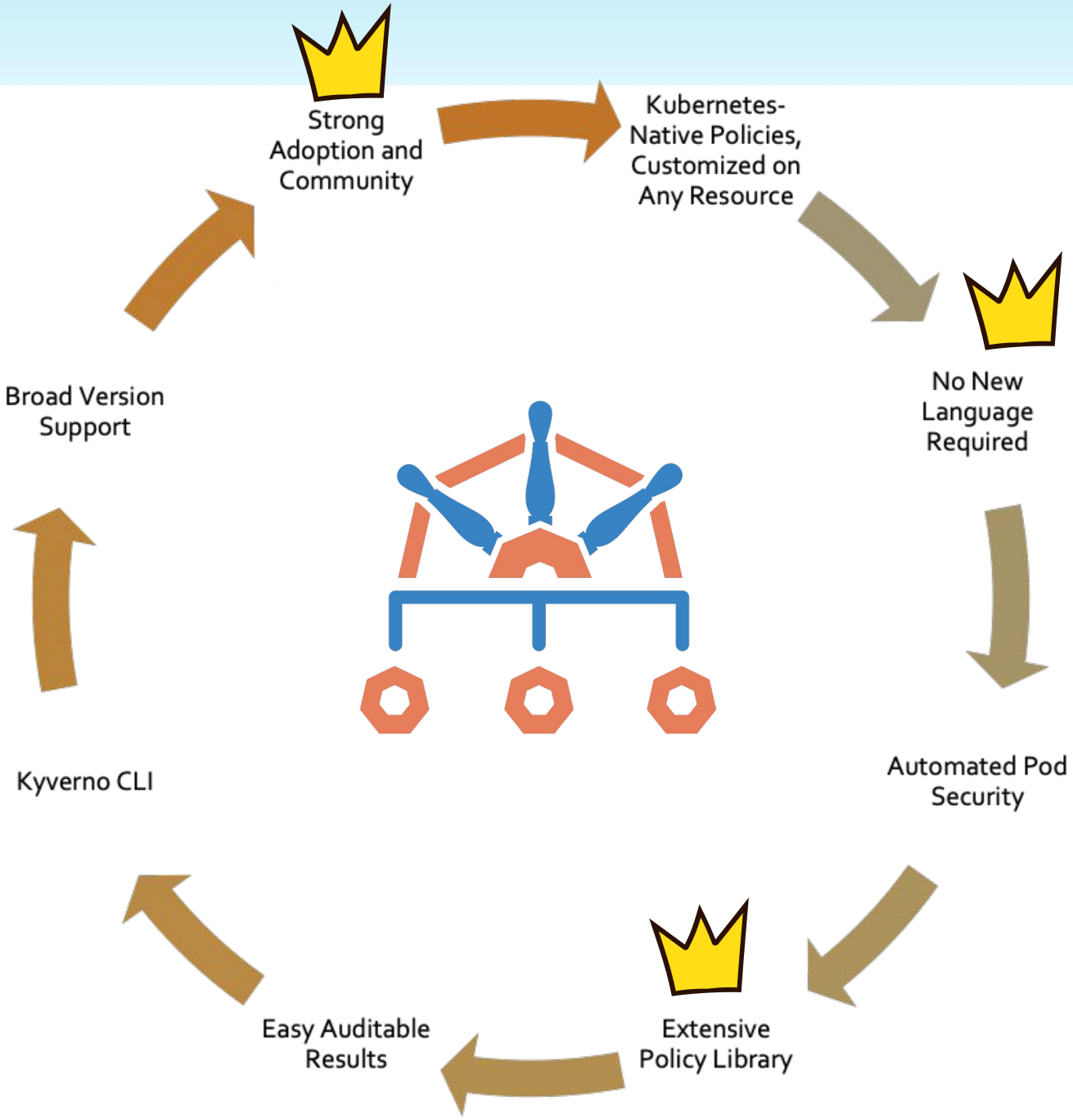
OPA

VS

Kyverno

VS

VAP/PSP


Strong Adoption and Community
Kubernetes-Native Policies, Customized on Any Resource
No New Language Required
Automated Pod Security
Extensive Policy Library
Easy Auditable Results
Kyverno CLI
Broad Version Support

jbrejnholt

Step by step

Additional security enforcement

Process automation

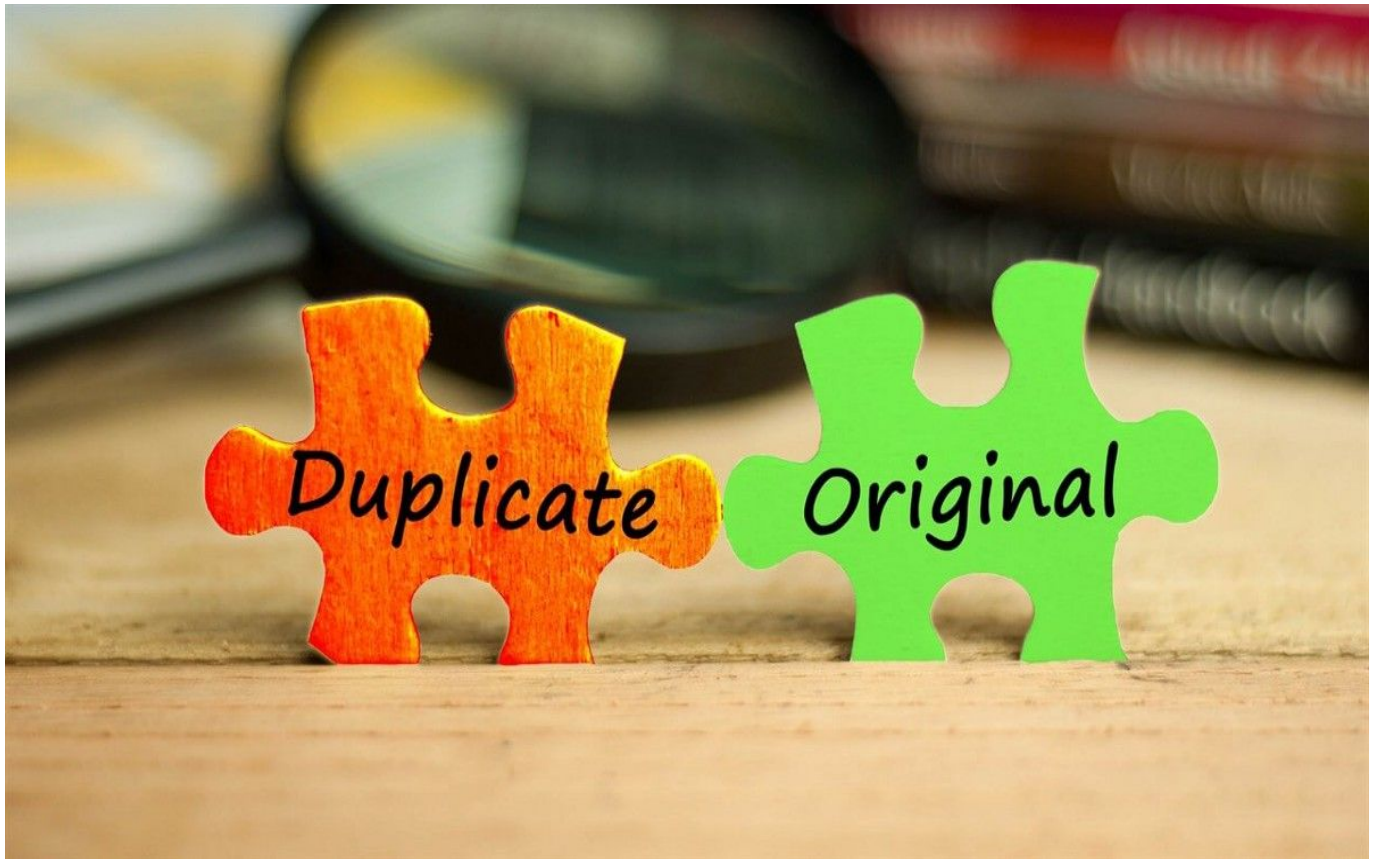
Out of the box policies

SHARING

jbrejnholt

1 Multi-tenancy

2 Resource Management

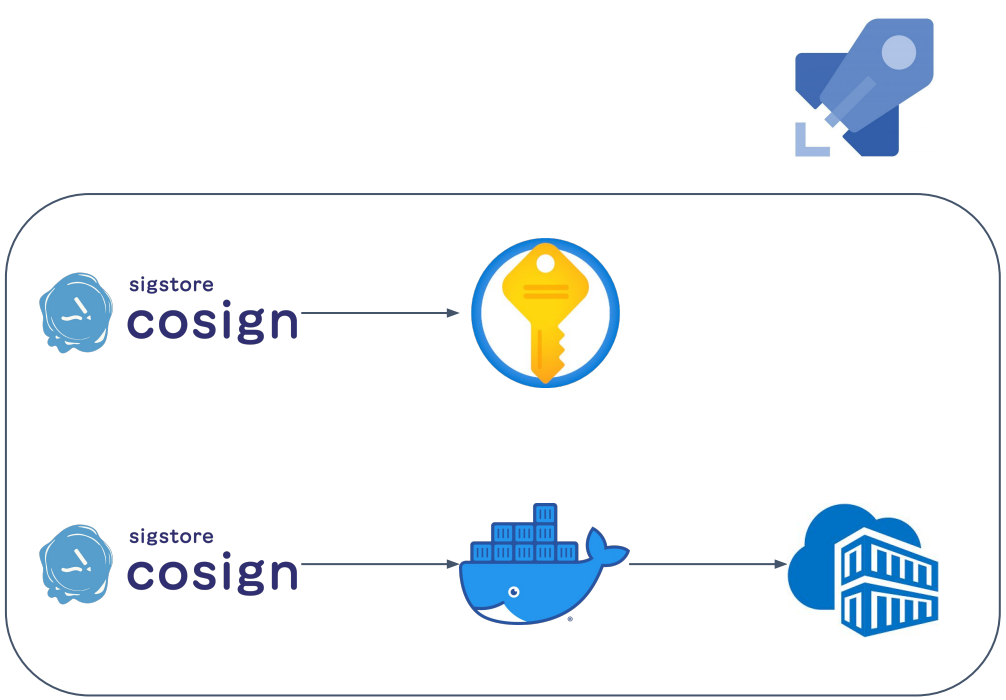


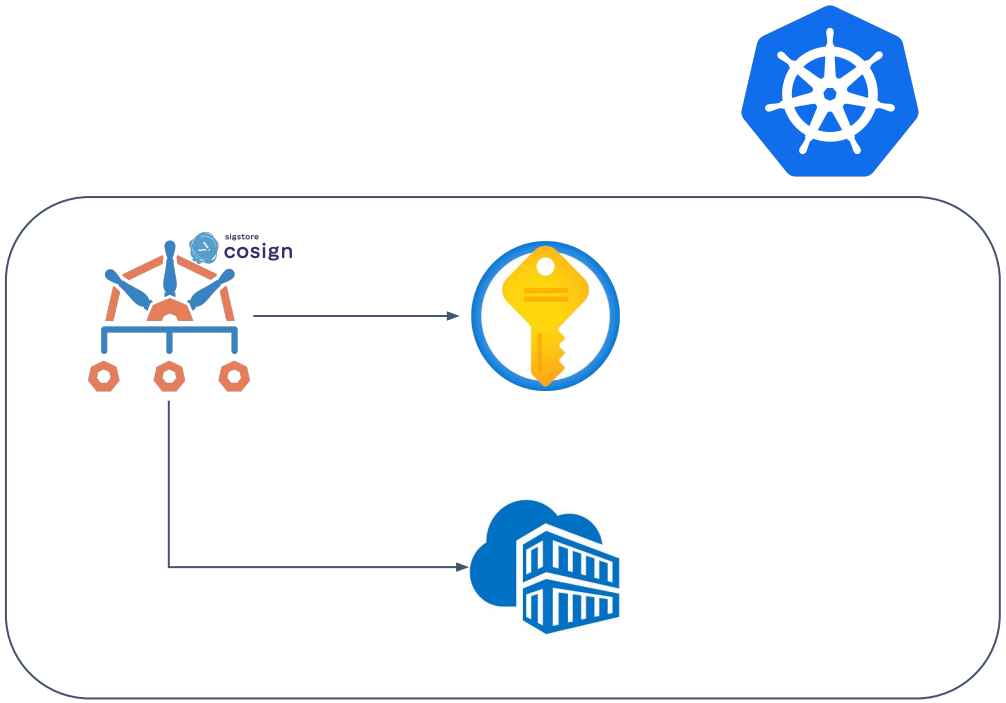
3 Resource Validation

4 Image Signature Verification

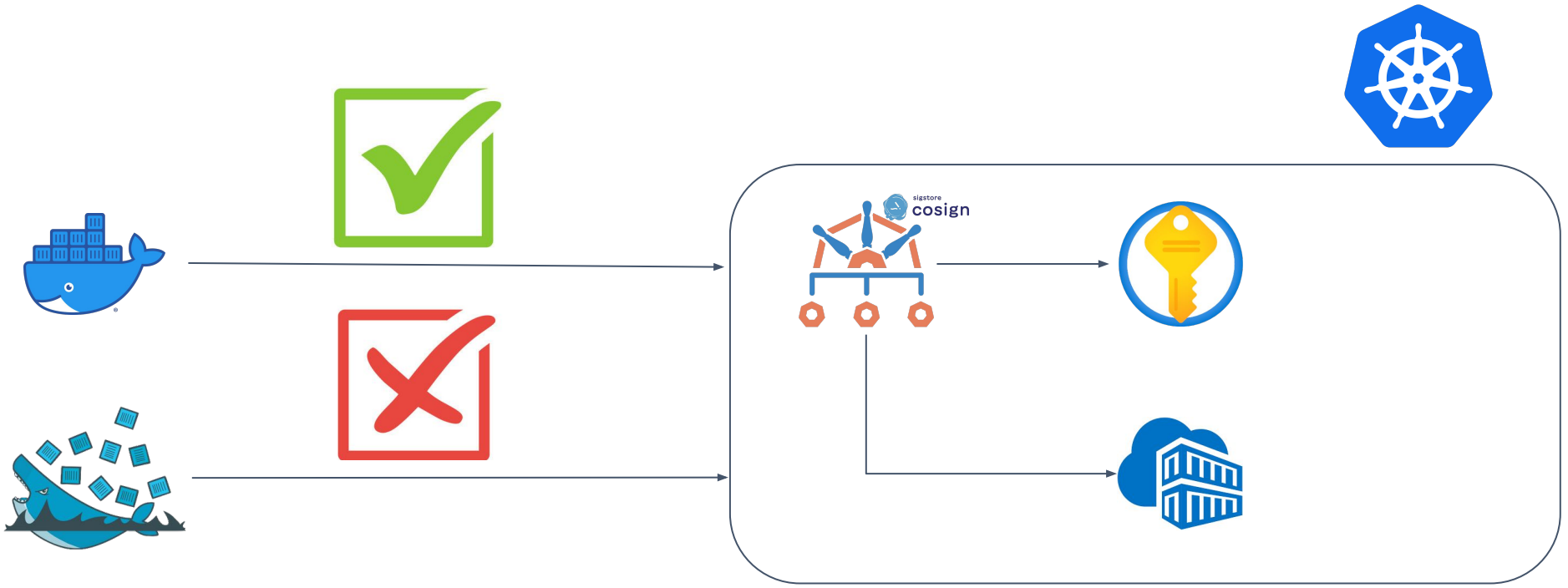
DEMO

The Setup

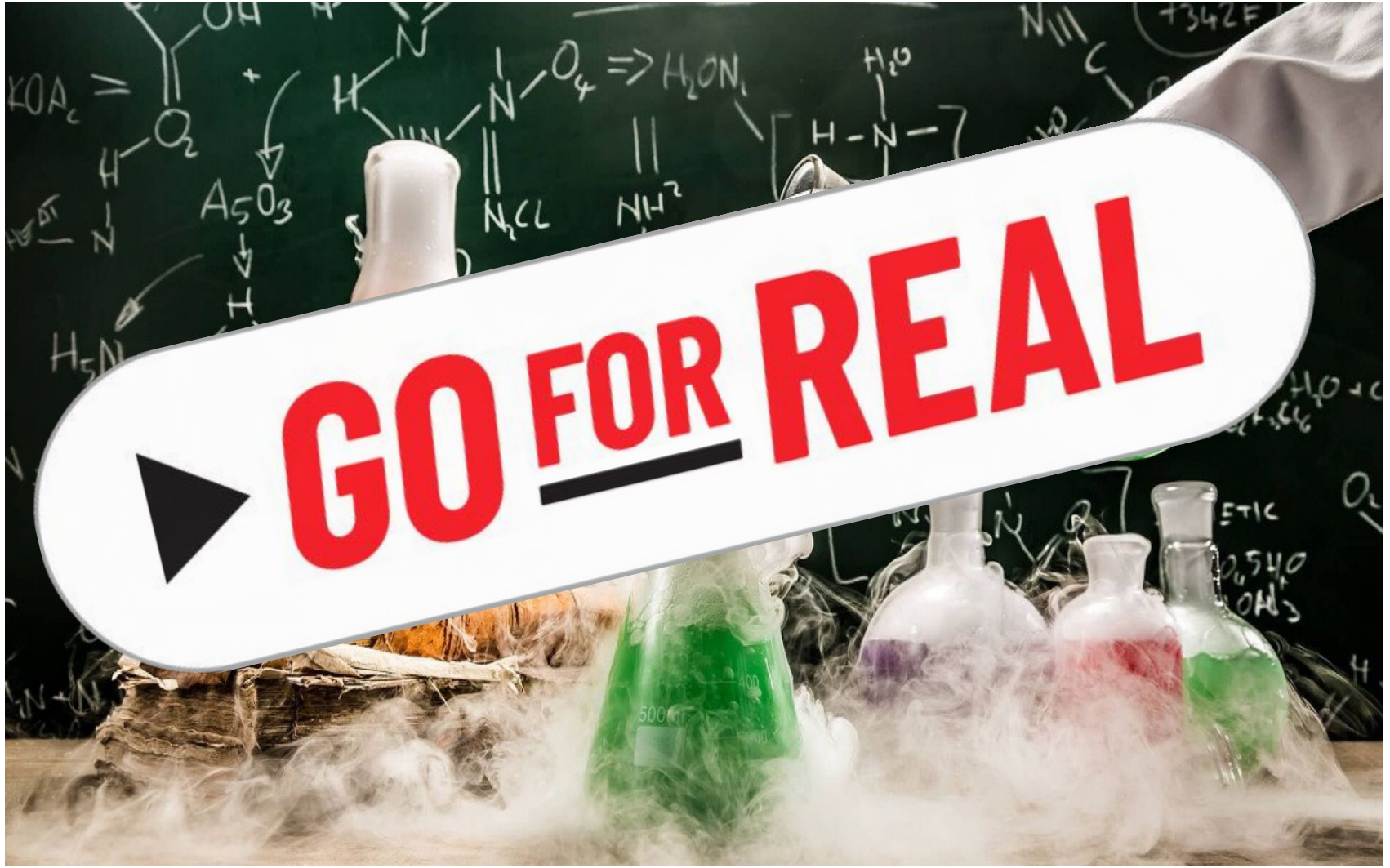




The Target



DEMO



KubeCon | CloudNativeCon

Europe 2023

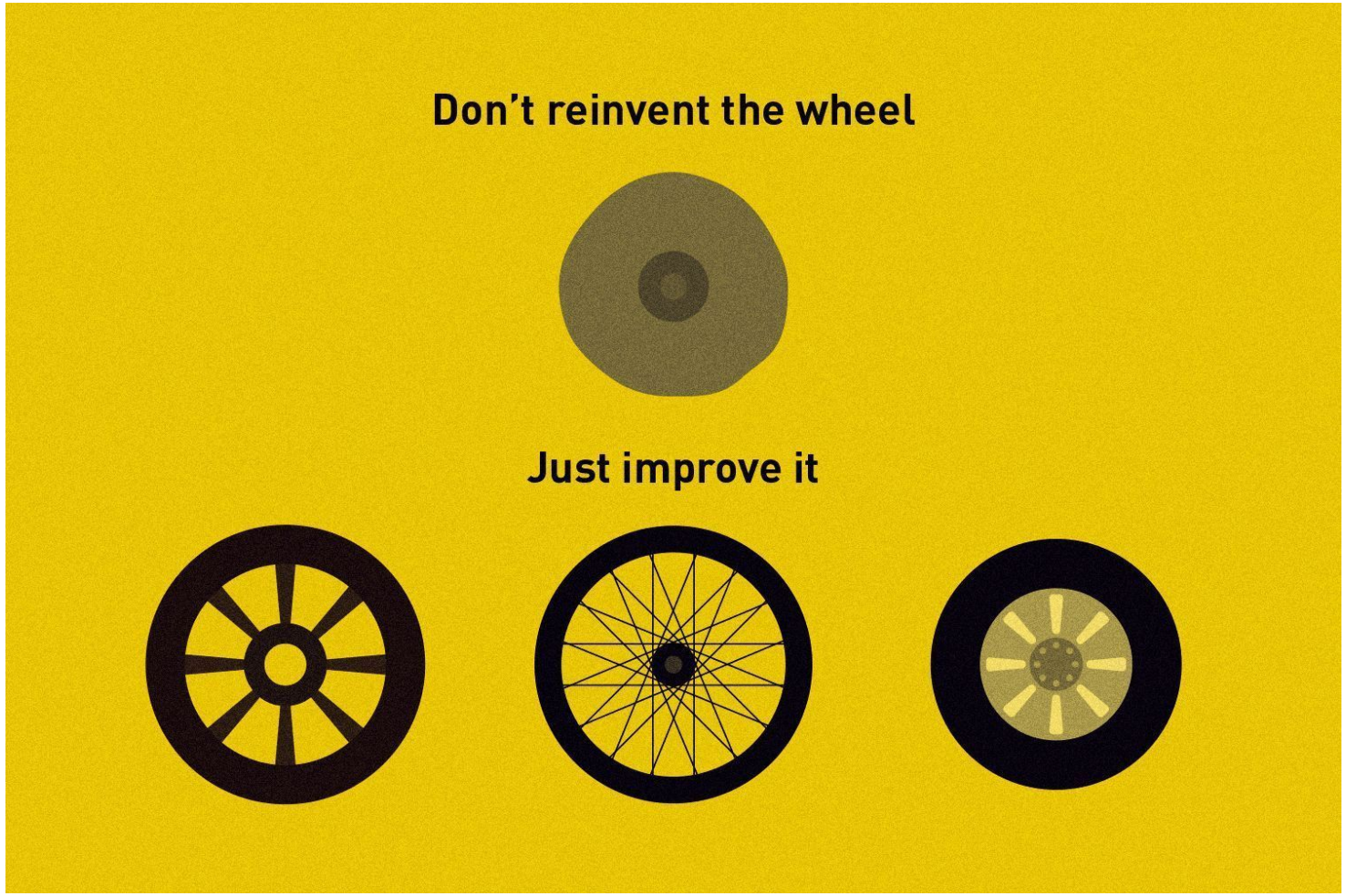
Learnings

jbrejnholt

Learning 1

–

Take advantage of the recommended policies.



Learning 2

–

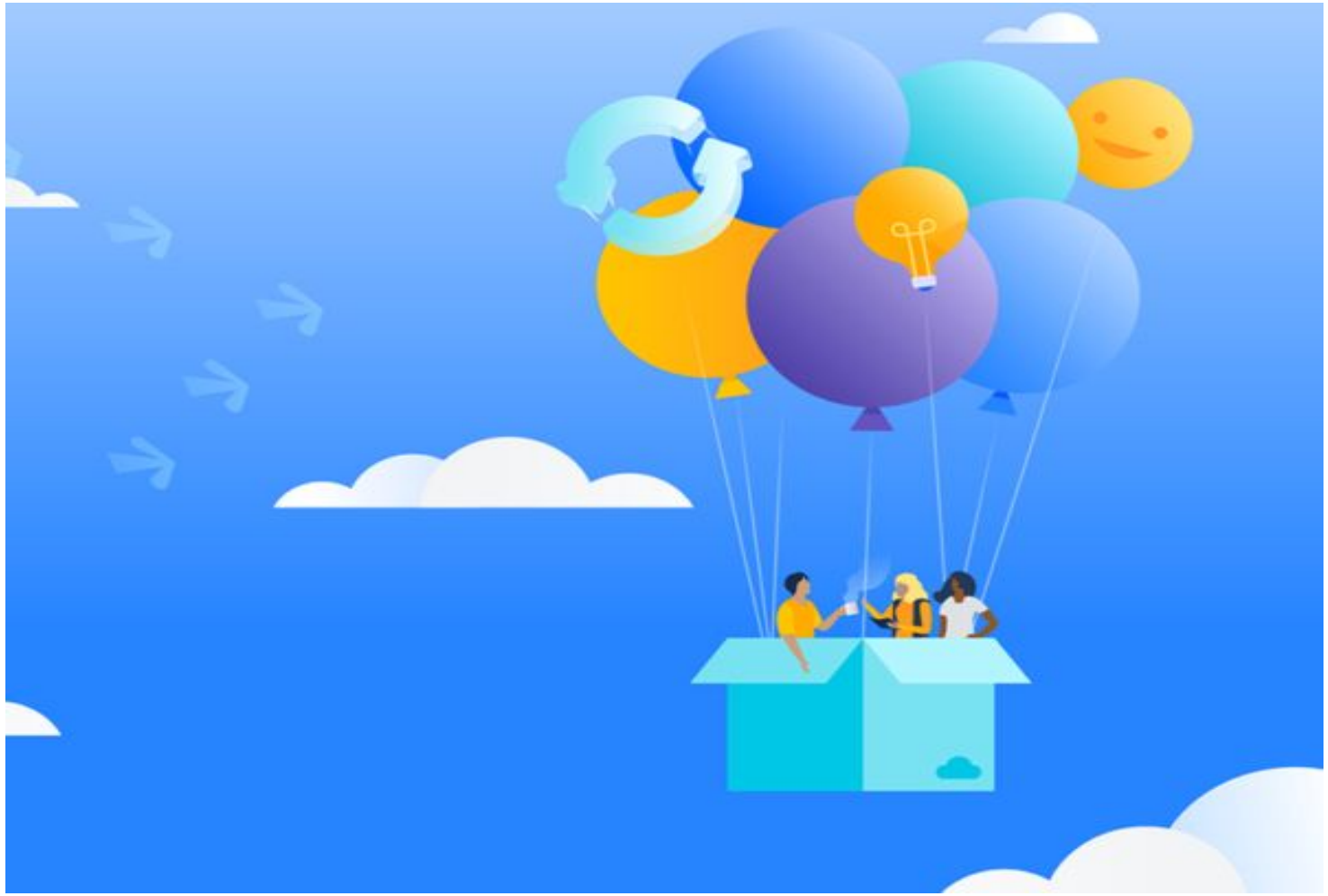
Start with:

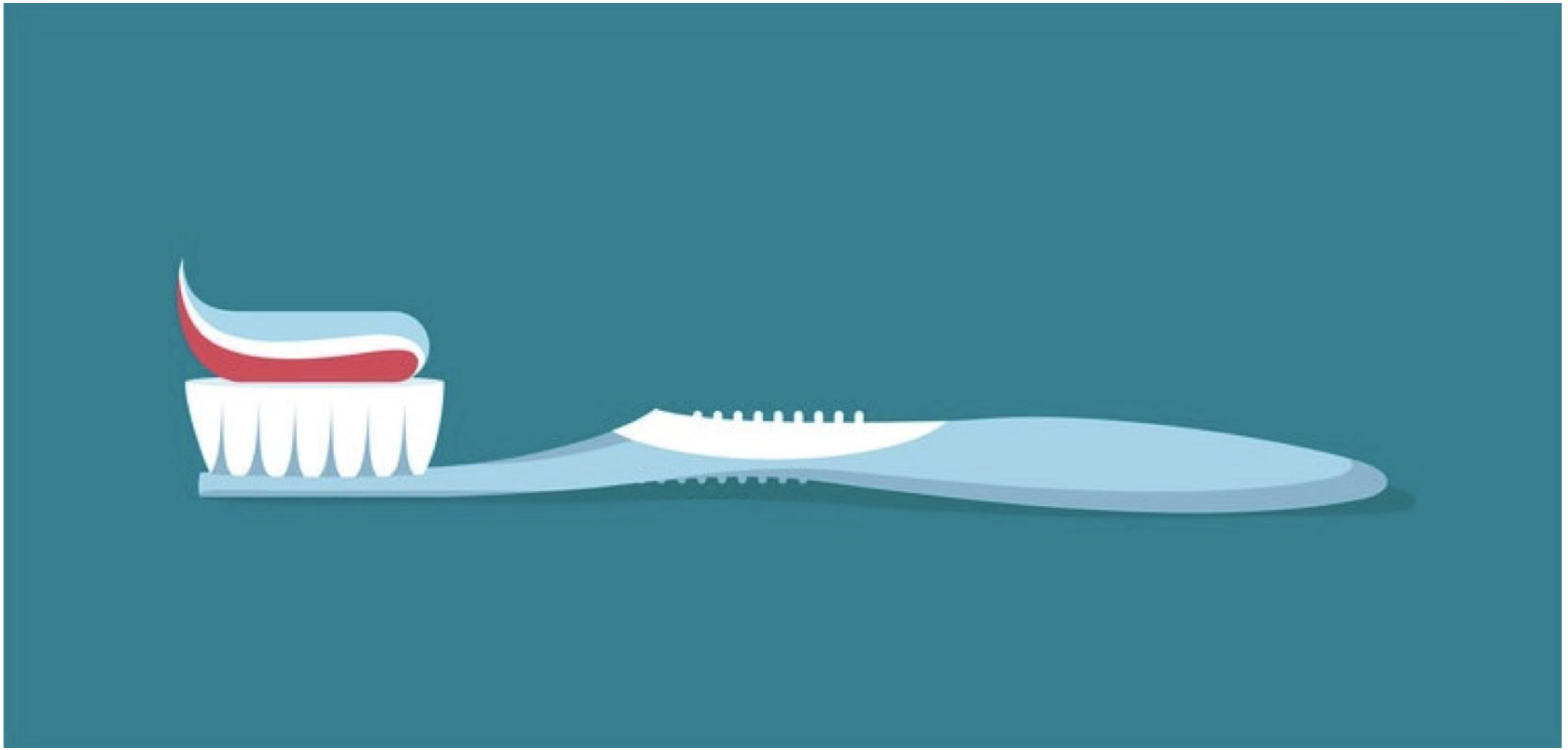
validationFailureAction: **Audit**

Learning 3

–

Migration from OPA doesn’t mean exact 1 to 1.





Advanced use cases / Upcoming features

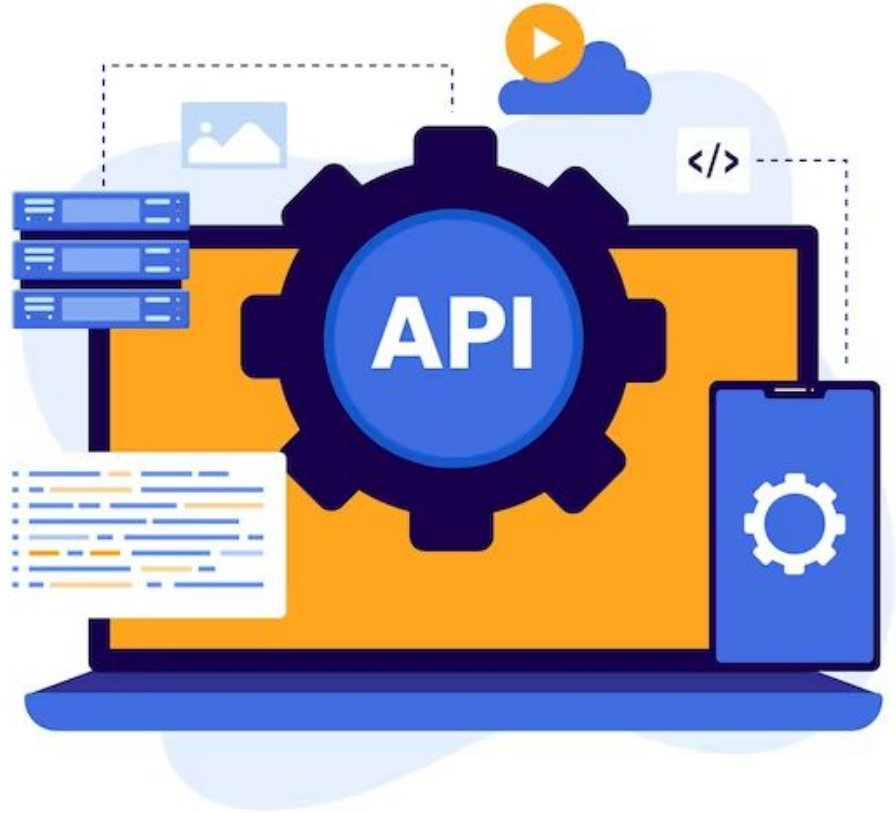
Initial support coming in 1.10

Advanced use cases / Upcoming features

- Notary uses OCI artifacts and referrers API (now part of the OCI standard)
- Not supported everywhere yet (ecr, acr, dockerhub)
- No keyless signature in Notary

```
oras discover -o tree jimnotarytest.azurecr.io/jim/net-monitor:v1
jimnotarytest.azurecr.io/jim/net-monitor@sha256:ba7000206594c2d72c3ab550453004c0dc50961157
e5ebd2fb8ea1890099d02d
├── vuln-scan
│   └── sha256:10c0b24faa551466b708d1677694eb65bbe4679ba10c3a5290ecec2e4f0af6c8
│       └── application/vnd.cncf.notary.signature
│           └── sha256:22aed3423fda62d88a09c47cbf1e4ca0441376fbc0270cafb193d4b9146eedc4
└── application/vnd.cncf.notary.signature
    └── sha256:e9277c9696d262e699583b7f4304eb9a3e7899a0d6c6b3c6b88327926347aaec
```

Advanced use cases / Upcoming features



Advanced use cases / Upcoming features


```yaml
rules:
- name: call-extension
  match:
    # ....
  context:
  - name: result
    apiCall:
      service:
        requestType: POST
        urlPath: http://sample.kyverno-extension/check-namespace
        data:
        - key: namespace
          value: "{{request.namespace}}"
  validate:
    message: "namespace {{request.namespace}} is not allowed"
    deny:
      conditions:
        all:
        - key: "{{ result.allowed }}"
          operator: EQUALS
          value: false
```

- Supports GET and POST methods
- Supports HTTP and HTTPS protocols
- Can POST and receive any JSON data
- Request result is available in the rule context

Advanced use cases / Upcoming features

Advanced use cases / Upcoming features

- Initial support should come in 1.11
- Currently experimenting with VAP in the CLI
- Can be challenging in certain areas like event generation
- Not all kyverno policies can be translated to VAP
  - Rule context equivalent is not possible yet
- But we are convinced it will improve overtime
- CEL itself is a great addition and users might want to bring CEL expressions in Kyverno policies


KubeCon
CloudNativeCon
Europe 2023
Thank you
jbrejnholt

Links:

Kyverno GitHub: https://github.com/kyverno/kyverno

Image Signature verification: https://kyverno.io/docs/writing-policies/verify-images/

PDB verification: https://main.kyverno.io/policies/other/pdb-minavailable/pdb-minavailable/

Require labels: https://main.kyverno.io/policies/best-practices/require_labels/require_labels/