October 25, 2022

Building DevSecOps into AWS EKS using Red Hat ACS

Pathik Patel

Head of Cloud Security

Informatica

Pathik Patel

@pathikpa – Twitter

With Informatica for past 6 years

Before that Netflix, Yahoo!

#Kubernetes, #python, #cryptography

Informatica

Informatica @ a Glance

$1.44B IN REVENUE

OFFERS INTELLIGENT DATA MANAGEMENT CLOUD (IDMC)

32 TRILLION TRANSACTION PER MONTH

MULTI-CLOUD OFFERINGS

My DevSecOps Opinion

Dev (development), Sec(security), Ops(operations) coming together and moving at same speed

Security is a shared responsibility

DevSecOps is a shared security mindset

Build guardrails for early feedback

Shiftleft gates to build secure code

DevSecOps Lifecycle

Plan & Develop

Threat Modelling

Pre-commit hooks

Secure coding standards

Peer Review

Code Commit

SAST

Dependency management

Secure pipeline

Build & Test

DAST

Configurations scans

Vulnerability scans

Configuration validation

Ship & Deploy

Security Smoke Tests

Configuration Checks

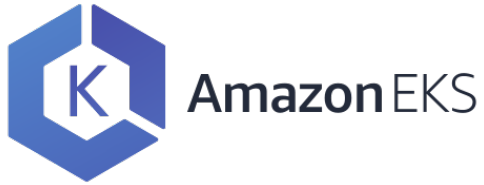
Pentest

Monitor

Continuous Monitoring

Vulnerability Scans

Tools of trade



Managed Kubernetes engine from AWS

Container registry offering hosted on AWS

Security configuration and monitoring tool

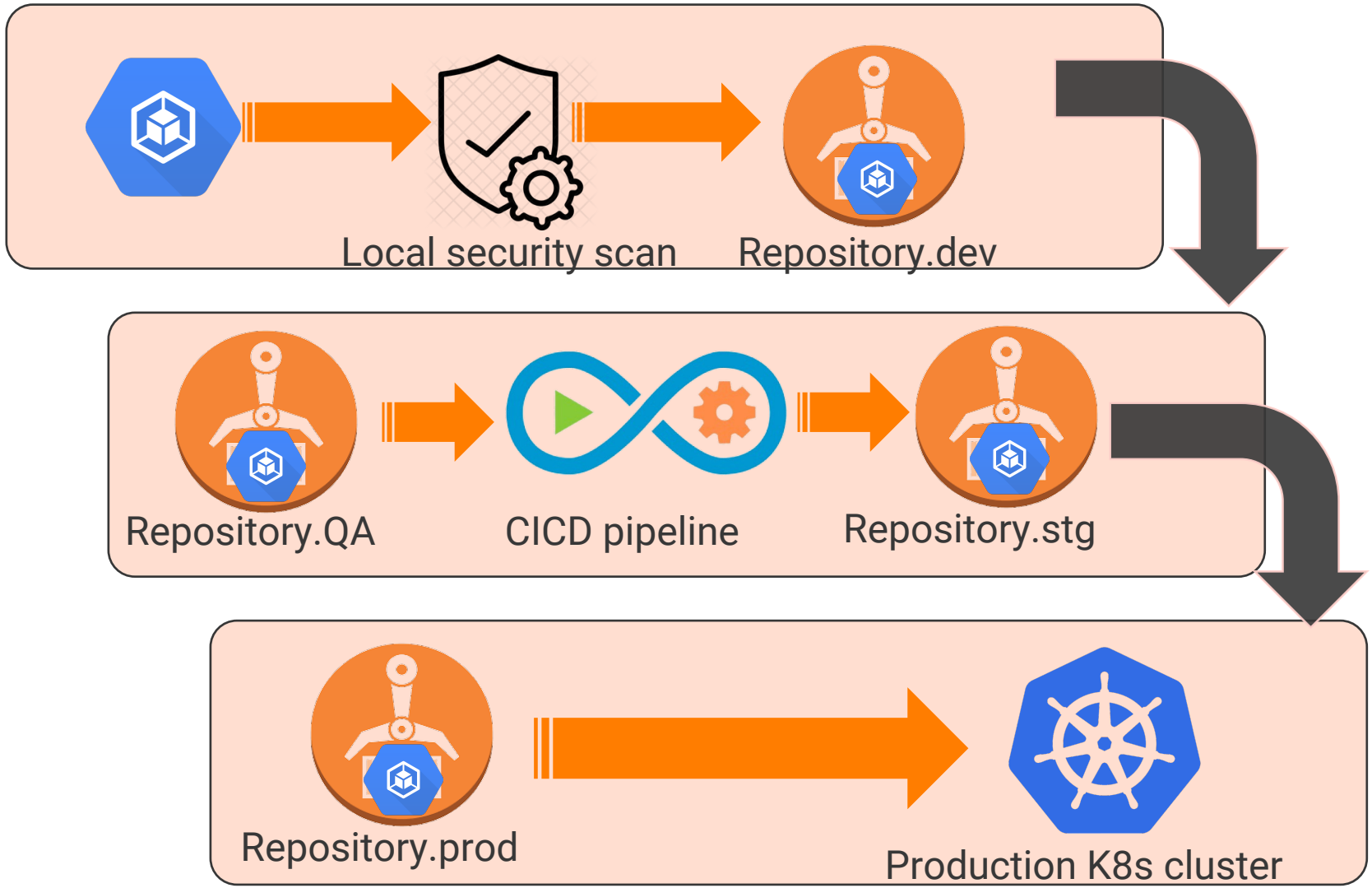
Ticketing and workflow management

Informatica

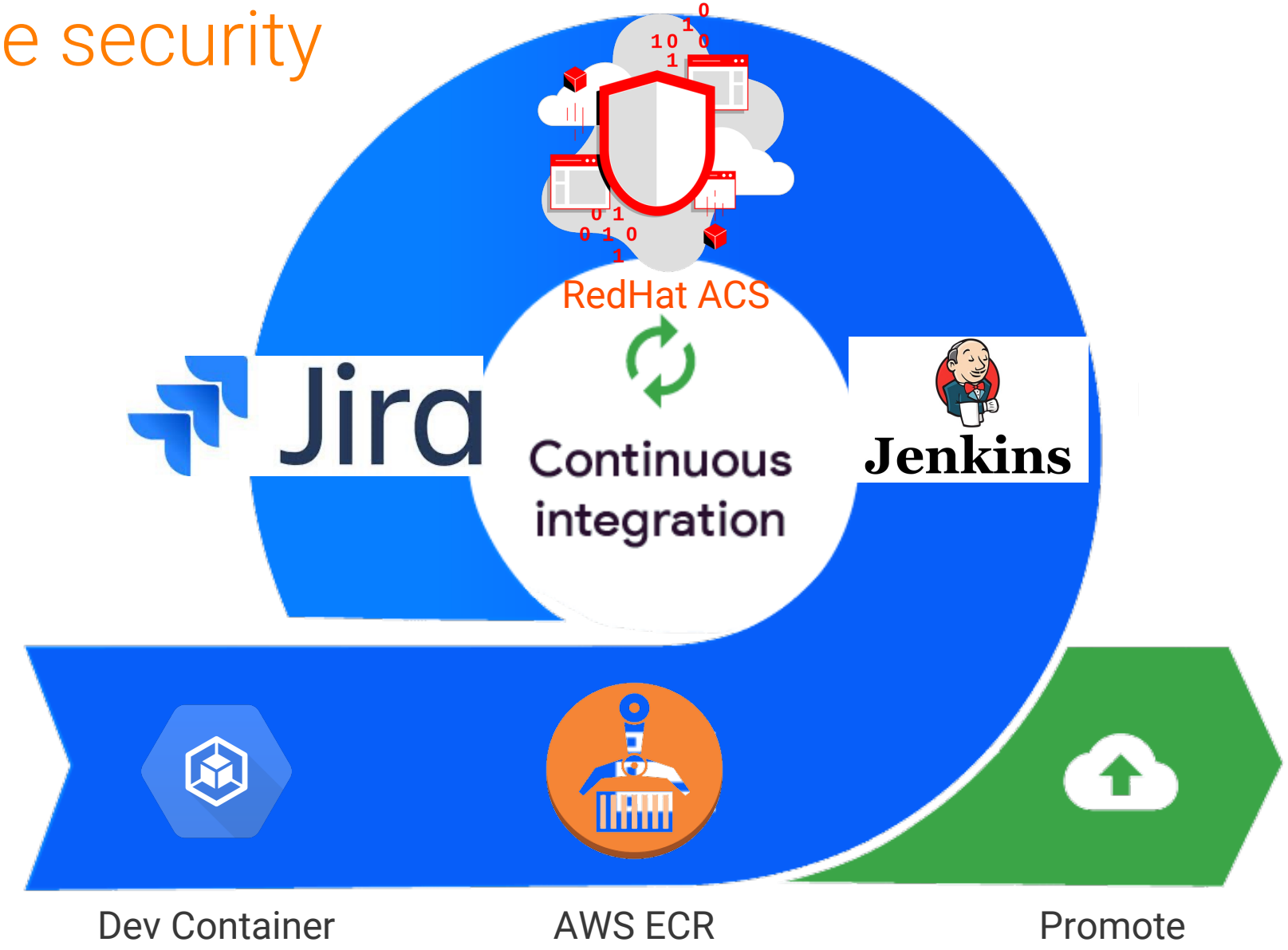
Best Practices

- Segregate your repositories
- Use distro-less container images
- Use admission controllers
- Enable Audit logs
- Consider build time and runtime security controls
- Adopt service mesh for optimal routing & encryption
- Implement CIS scans for container, worker node and clusters

Promoting your images



Build time security

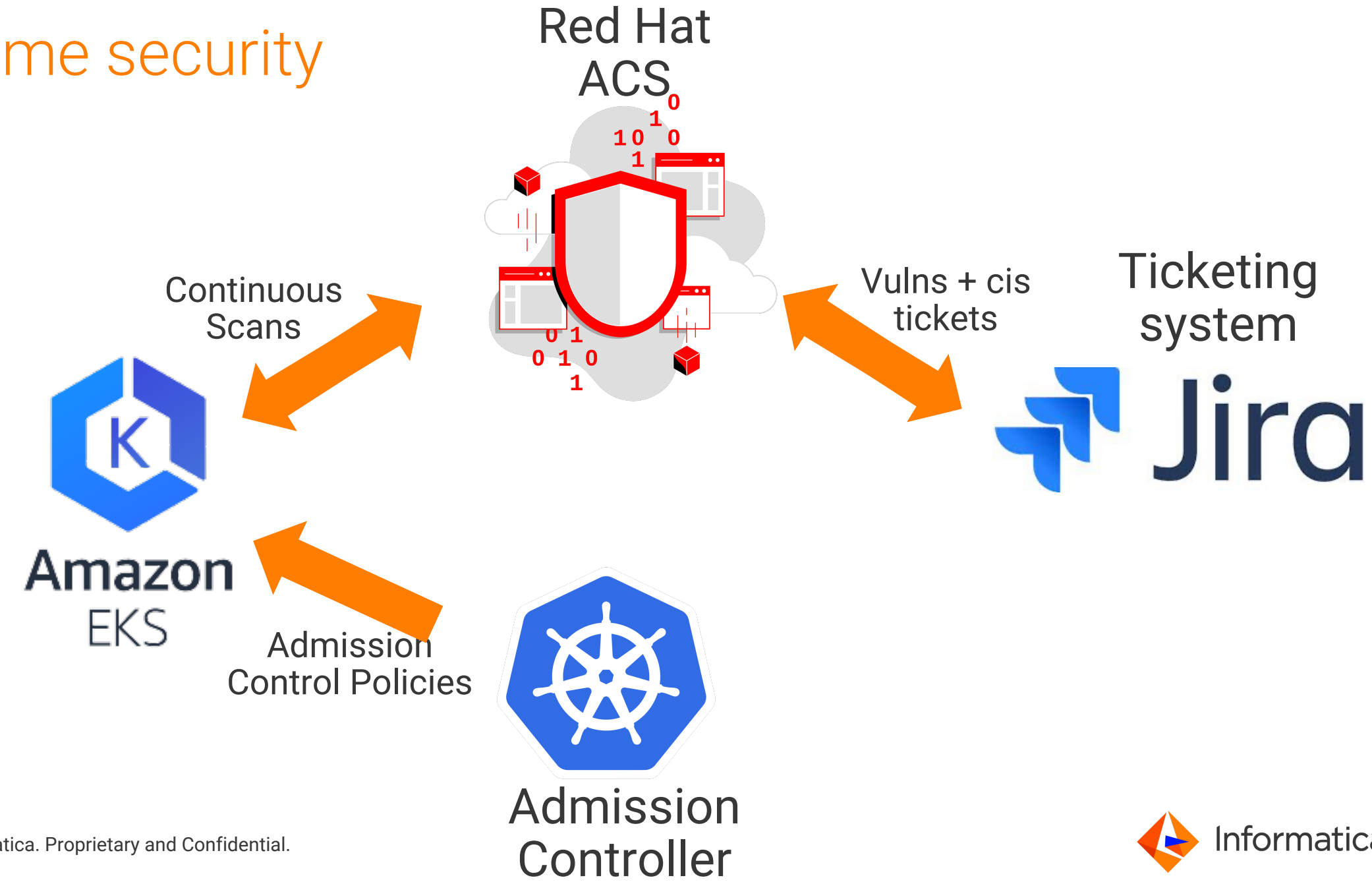


Informatica

Build secure images

- Distro-less images will save you lot of pain
- Remove package managers, unused utilities (e.g. network, filesystem)
- Encourage developers to scan while building image
- Integrate ACS with Jenkins CICD for continuous feedback during build time
- Scan at dev time and promote to QA
- Scan in CI pipeline and promote to staging
- Block build system from using non-compliant repositories
- Allow only from approved labeled repositories

Run time security

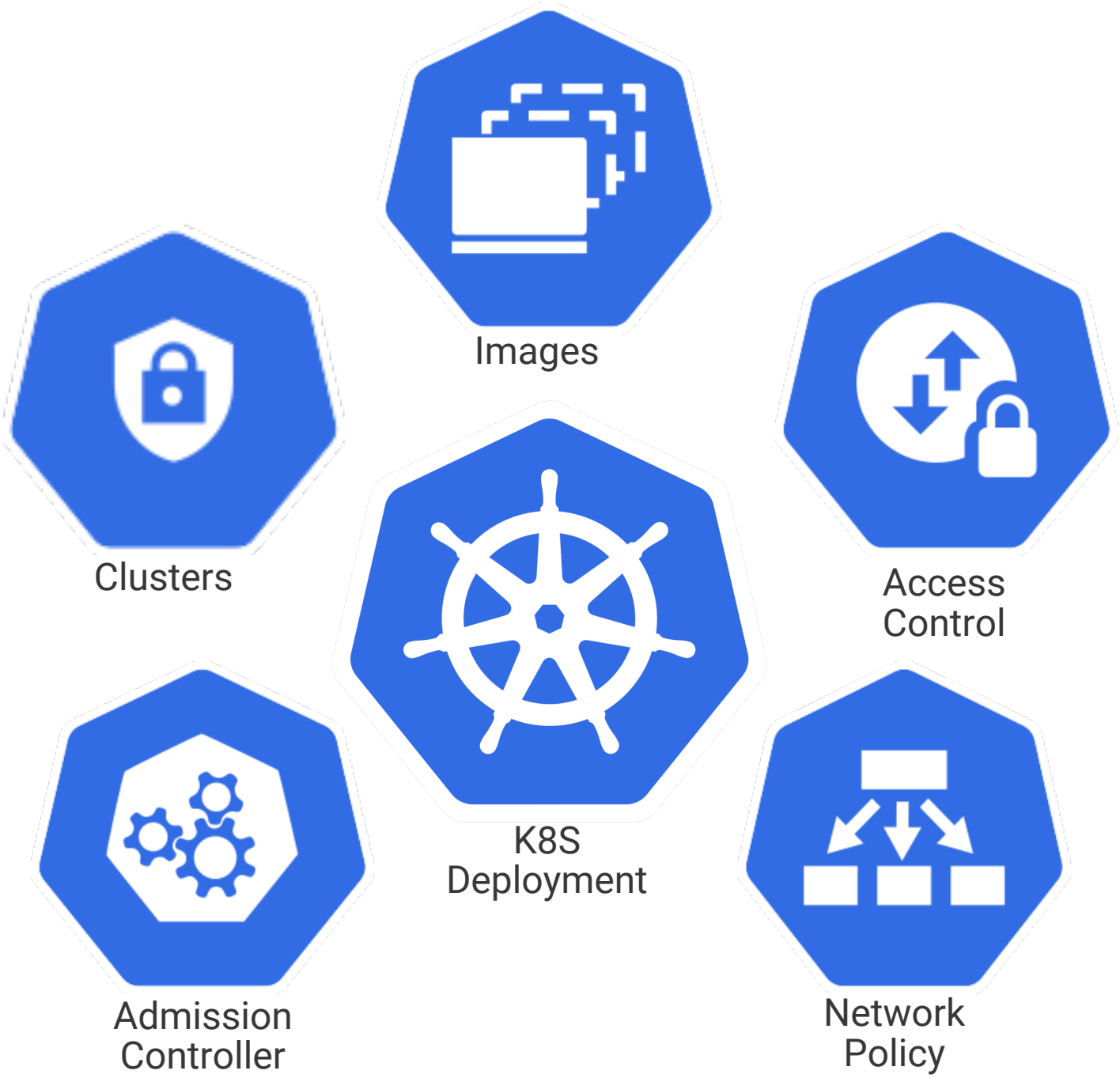


Informatica

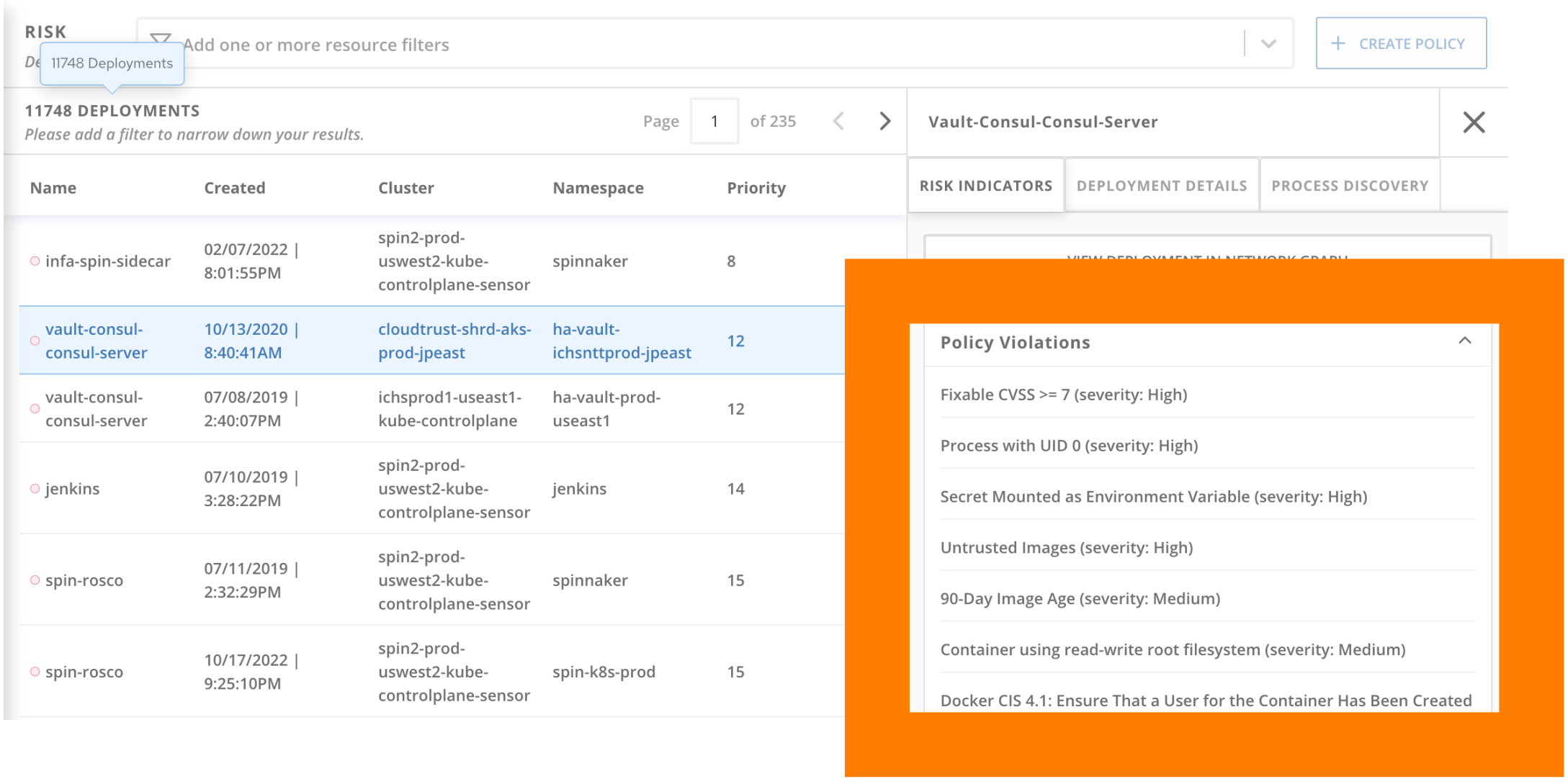
Secure your runtime

- Enforce NameSpace usage to
  - Build out segregation between products, business units
  - Enforce RBAC to allow access within namespace only
  - Create network policies and segmentation
- Use Admission Controller to
  - Enforce company wide guardrails
  - Implement PodSecurityPolicies(PSP)
  - Protect your API server
  - Enforce repository usage
- Monitor configurations to
  - Detect CIS compliance failures
  - Rogue container processes
  - Enforce RBAC

Risk Lens


Images
Clusters
Access Control
K8S Deployment
Admission Controller
Network Policy

Risk based evaluation



Take Aways

- Define guard rails and document
- Codify guard rails using Red Hat ACS
- Provide early feedback to developers
- K8s amplifies your problems (1000s of containers)
- Build automation to report

Questions??

Thank you

ppatel@informatica.com