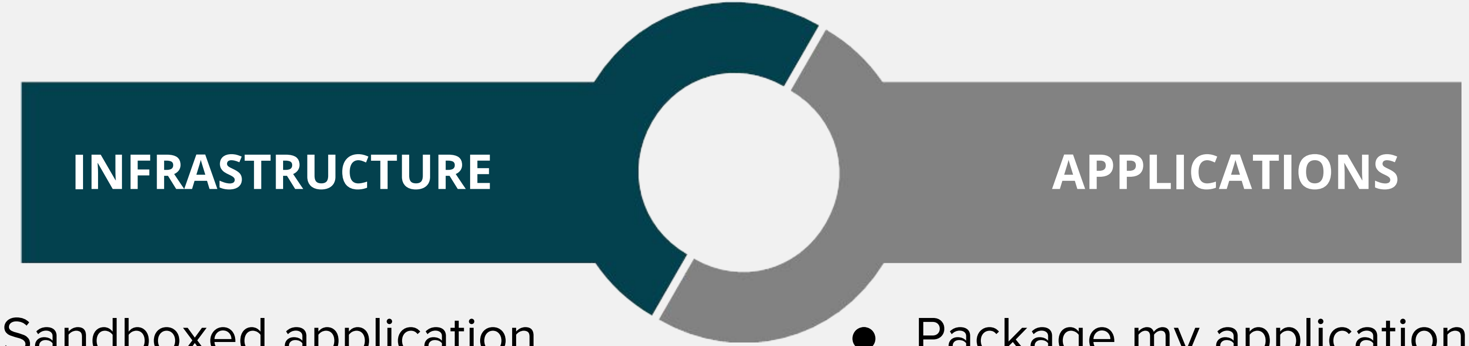
Securing Containers with Red Hat OpenShift

Kirsten Newcomer, Security Strategist
William Henry, DevOps and Cloud Strategist

CONTAINERS CHANGE HOW WE DEVELOP, DEPLOY AND MANAGE APPLICATIONS



INFRASTRUCTURE

APPLICATIONS

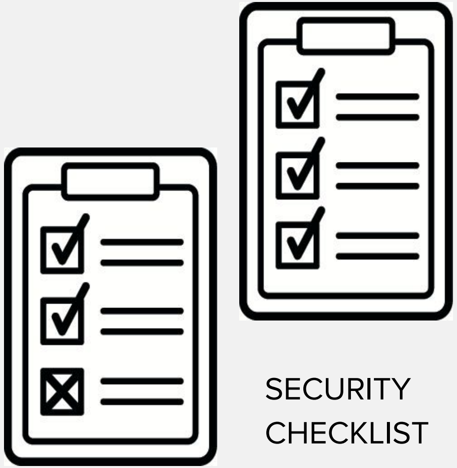
- Sandboxed application processes on a shared Linux OS kernel
- Simpler, lighter, and denser than virtual machines
- Portable across different environments

- Package my application and all of its dependencies
- Deploy to any environment in seconds and enable CI/CD
- Easily access and share containerized components

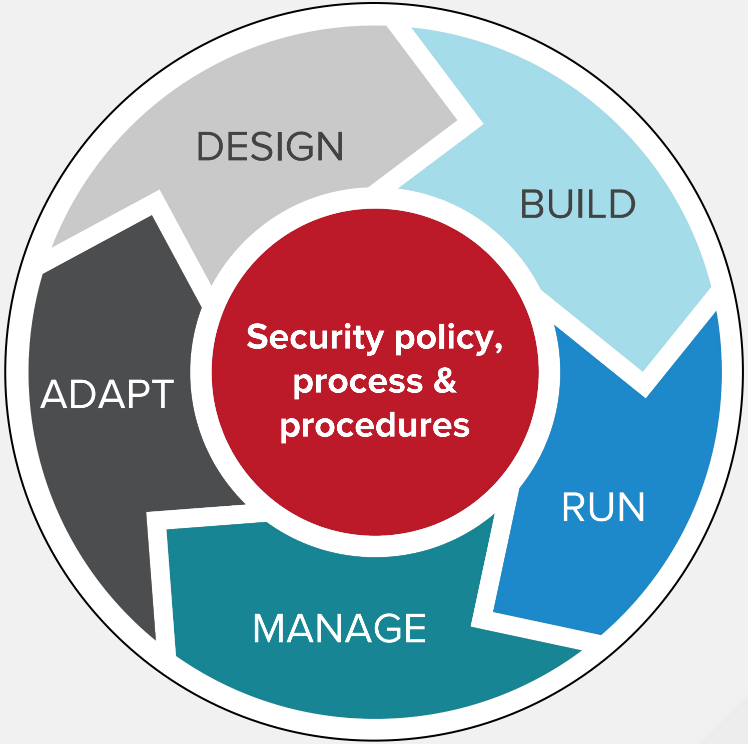
THEY ALSO CHANGE HOW WE SECURE OUR WORKLOADS

SECURITY MUST BE CONTINUOUS

And integrated throughout the IT lifecycle



SECURITY CHECKLIST



SECURING THE CONTAINER LIFECYCLE & THE CONTAINER STACK

CONTROL

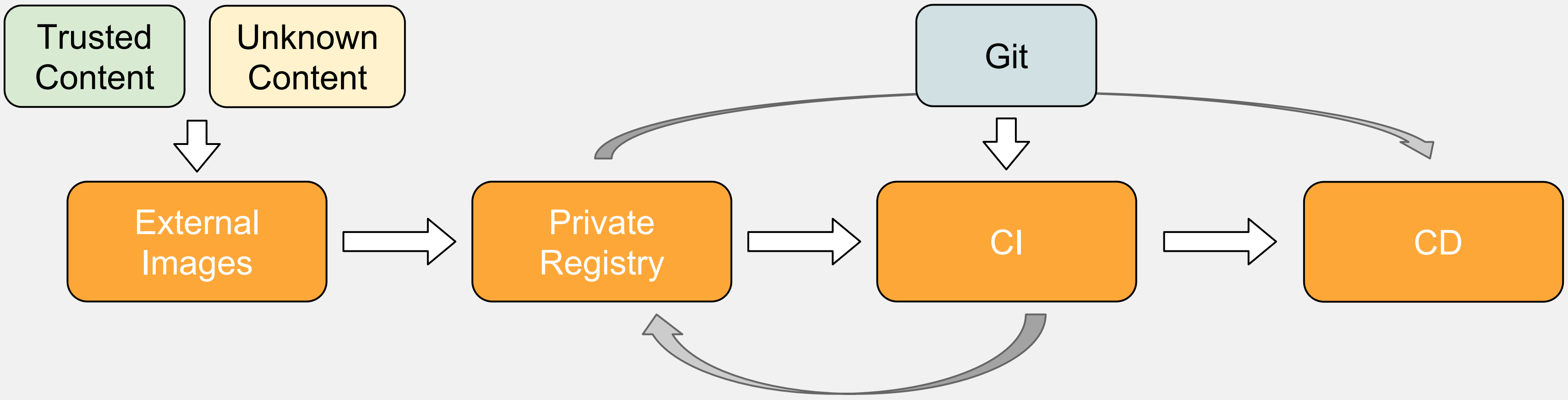
DEFEND

EXTEND

CONTROL

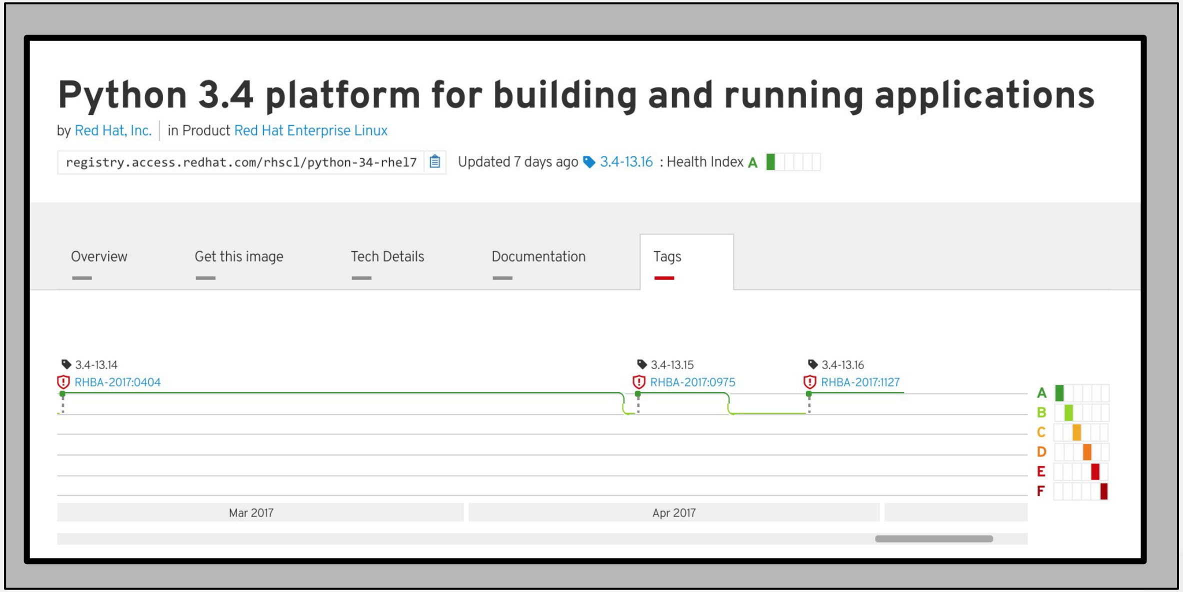
Secure the Pipeline & the Applications

THE CONTAINER CONTENT LIFECYCLE



CONTENT: USE TRUSTED SOURCES

- Are the container images signed?
- Are the runtime and OS layers up to date?
- How frequently will the container be updated and how will I know when it’s updated?



Python 3.4 platform for building and running applications

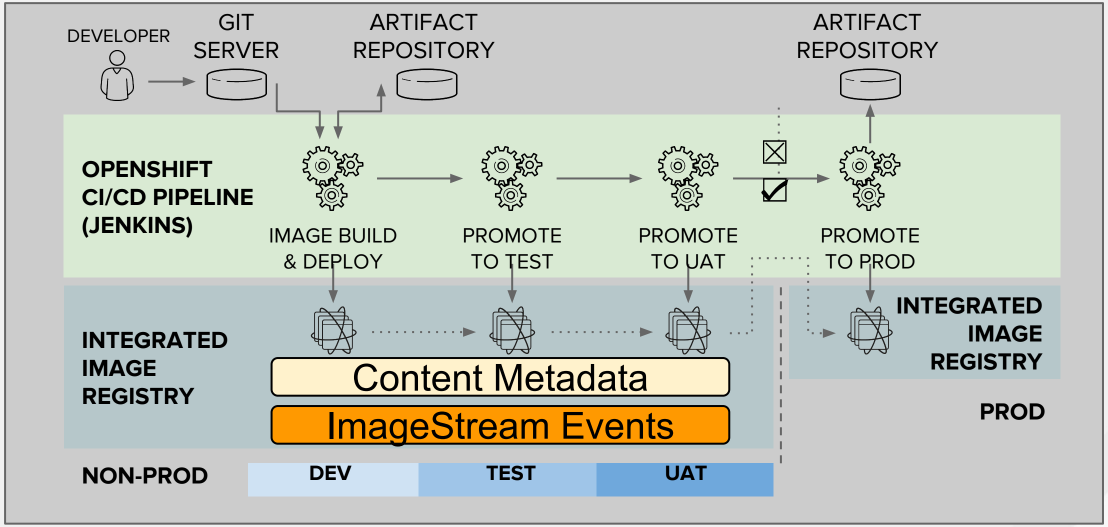
by Red Hat, Inc. | in Product Red Hat Enterprise Linux

`registry.access.redhat.com/rhscl/python-34-rhel7` Updated 7 days ago 3.4-13.16 : Health Index A

Red Hat rebuilds container images when security fixes are released

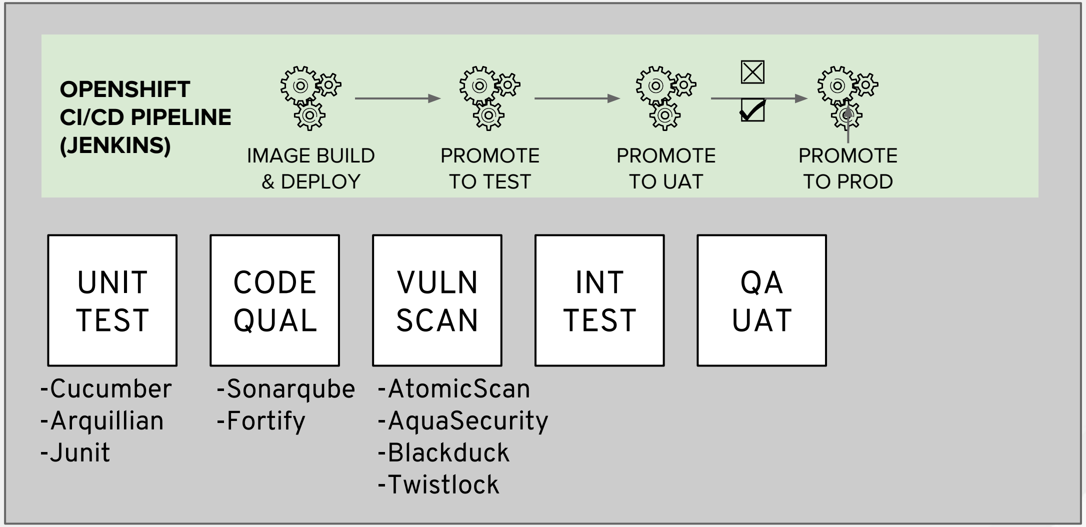
PRIVATE REGISTRIES: SECURE ACCESS TO IMAGES

- Manage access to and promotion of images
- Metadata to automate policies for approved use (e.g. dev, test, UAT, production)
- Monitor changes to external sources
- Manage image signatures for your custom containers



CONTINUOUS INTEGRATION MUST INCLUDE SECURITY GATES

- Integrate security testing into your build / CI process
- Use automated policies to flag builds with issues
- Trigger automated rebuilds
- Sign your custom container images



OPENSHIFT CI/CD PIPELINE (JENKINS)

IMAGE BUILD & DEPLOY → PROMOTE TO TEST → PROMOTE TO UAT ☒ ☑ → PROMOTE TO PROD

UNIT TEST
-Cucumber
-Arquillian
-Junit

CODE QUAL
-Sonarqube
-Fortify

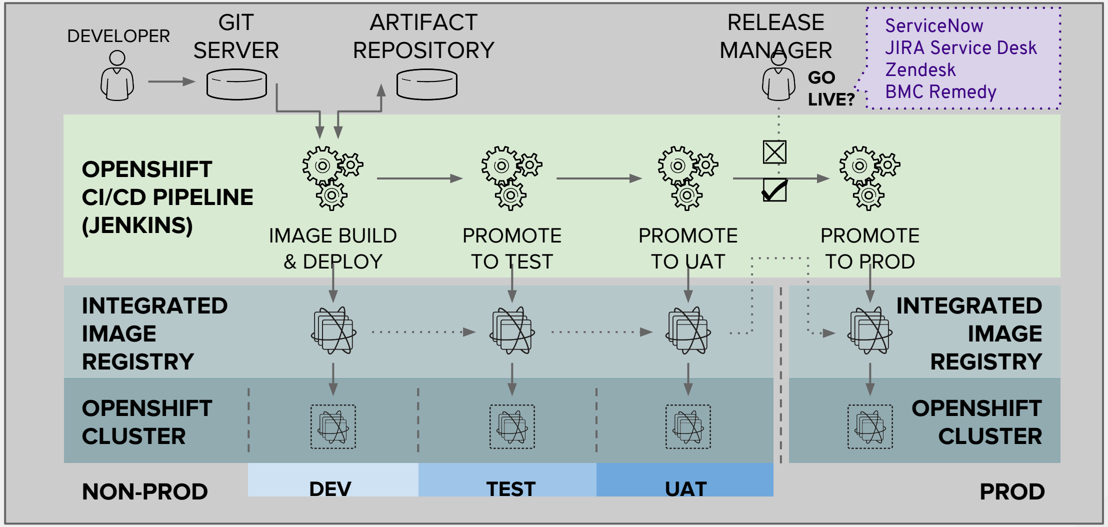
VULN SCAN
-AtomicScan
-AquaSecurity
-Blackduck
-Twistlock

INT TEST

QA UAT

MANAGING CONTAINER DEPLOYMENT

- Monitor image registry to automatically replace affected images
- Enforce signatures at node level via signing trust policy
- Use policies to gate what can be deployed: e.g. if a container requires root access, prevent deployment
- Trust is temporal; rebuild & redeploy as needed



DEFEND

Secure the Infrastructure

CONTAINER HOST & MULTI-TENANCY
THE OS MATTERS

RED HAT ENTERPRISE LINUX

RED HAT ENTERPRISE LINUX ATOMIC HOST

THE FOUNDATION FOR SECURE, SCALABLE CONTAINERS

A stable, reliable host environment with built-in security features that allow you to isolate containers from other containers and from the kernel.

Minimized host environment tuned for running Linux containers while maintaining the built-in security features of Red Hat Enterprise Linux..

SELinux

Kernel & User namespaces

Capabilities

Cgroups

Seccomp

SECURING THE CONTAINER PLATFORM

Use a container orchestration platform with integrated security features including

- Role-based Access Controls with LDAP and OAuth integration
- Platform multitenant security
- Integrated & extensible secrets management
- Logging, Monitoring, Metrics
- Enable integration with the security ecosystem

Hawkular

Prometheus

fluentd

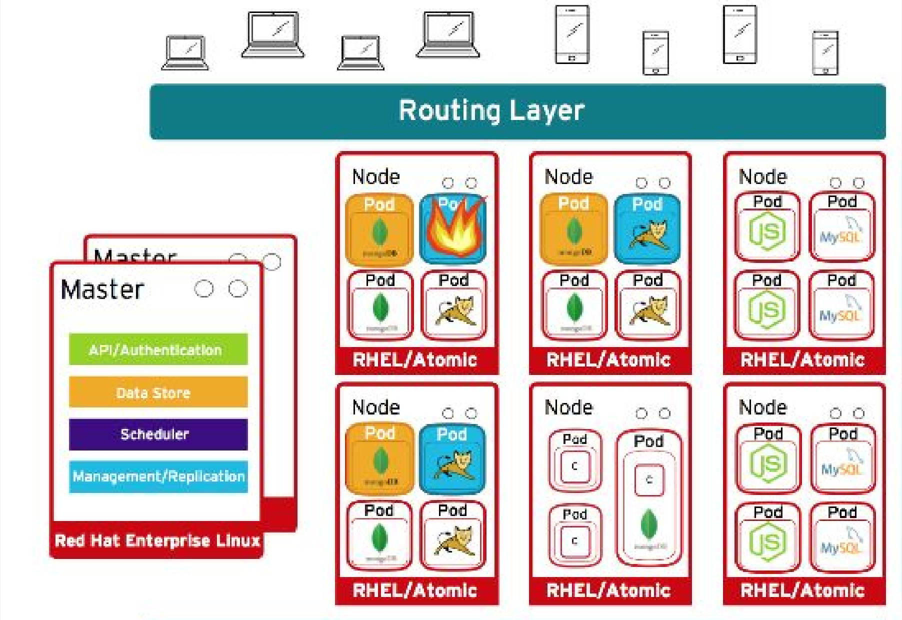
elastic

kibana

NETWORK DEFENSE

Use network namespaces to

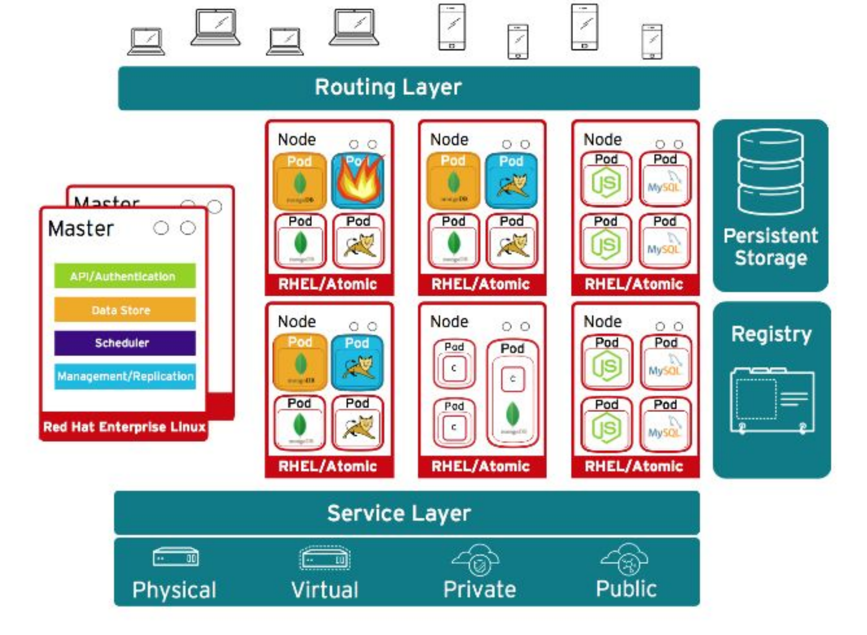
- Isolate applications from other applications within a cluster
- Isolate environments (Dev / Test / Prod) from other environments within a cluster



ATTACHED STORAGE

Secure storage by using

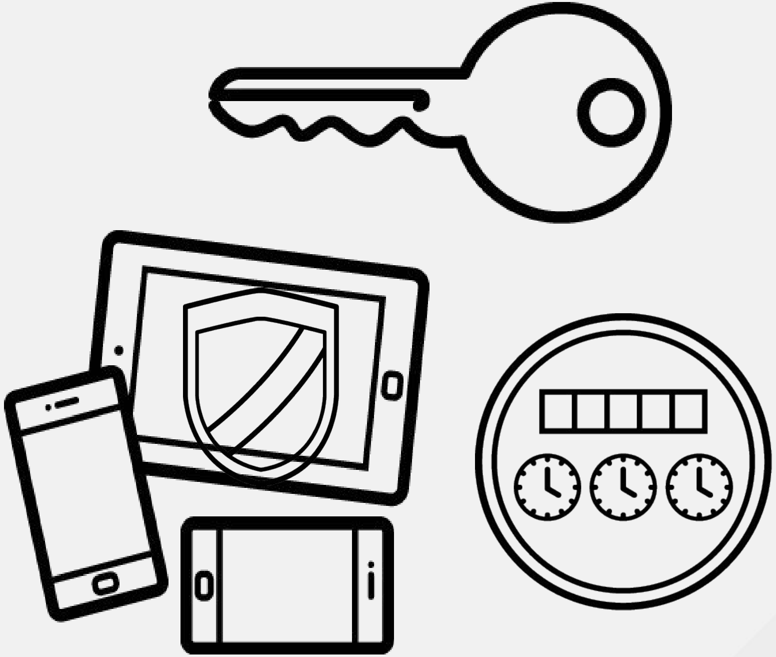
- SELinux access controls
- Secure mounts
- Supplemental group IDs for shared storage



API MANAGEMENT

Container platform & application APIs

- Authentication and authorization
- LDAP integration
- End-point access controls
- Rate limiting



EXTEND

Leverage the Ecosystem

THE SECURITY ECOSYSTEM

For enhanced security, or to meet existing policies, integrate with enterprise security tools, such as

- Identity and Access management / Privileged Access Management
- External Certificate Authorities
- External Vaults / Key Management solutions
- Container content scanners & vulnerability management tools
- Container runtime analysis tools
- Security Information and Event Monitoring (SIEM)

And use open source & open standards

More about OpenShift Primed Partners

LOOKING INTO THE NOT SO DISTANT FUTURE

CONTAINER CHALLENGES

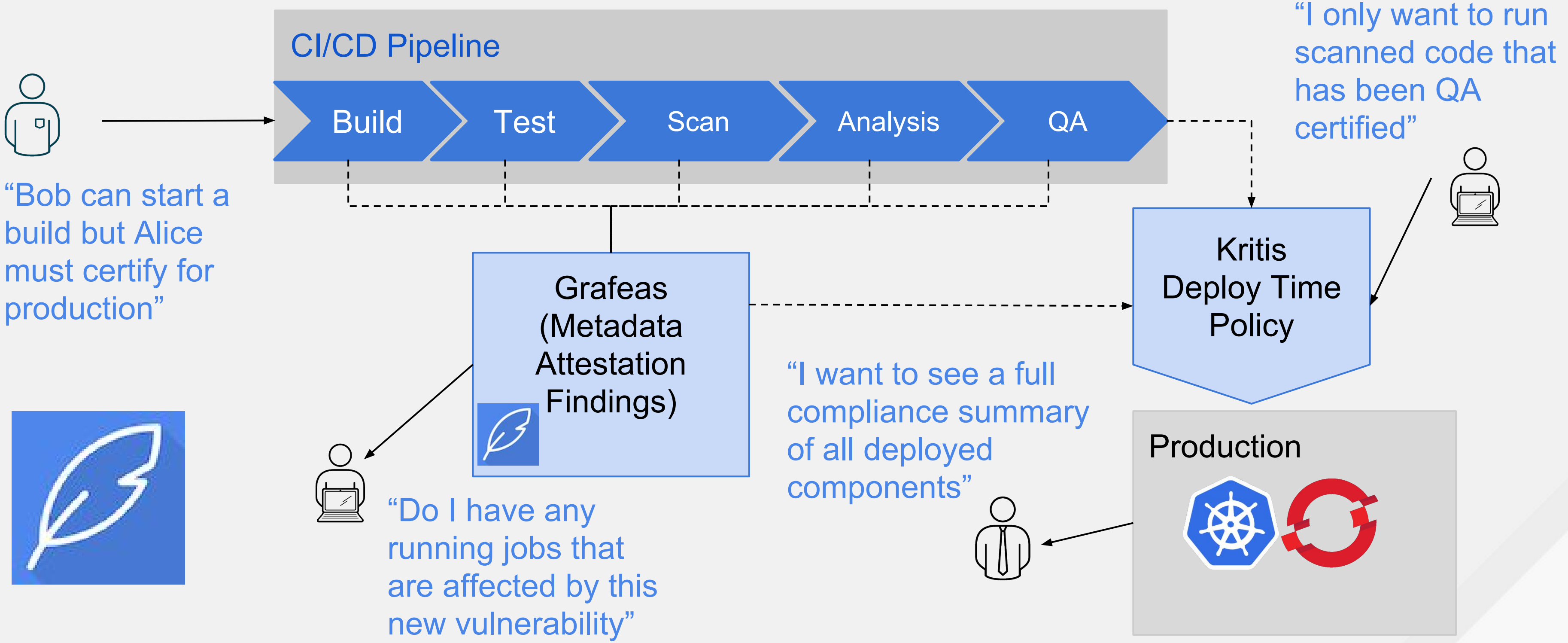
Enterprise Build, Pipeline and Runtime concerns

Excellent progress with containers so far, but much to be done

- Supply chain needs further security policy services
- Microservices have special networking and governance needs
- Build and runtime tools and services need decoupling

ATTESTATION OF SECURITY POLICY

Grafeas (Scribe) and Kritis (Judge)



CI/CD Pipeline

Build

Test

Scan

Analysis

QA

“Bob can start a build but Alice must certify for production”

Grafeas (Metadata Attestation Findings)

Kritis Deploy Time Policy

“I only want to run scanned code that has been QA certified”

“I want to see a full compliance summary of all deployed components”

“Do I have any running jobs that are affected by this new vulnerability”

Production

ISTIO AND MICROSERVICES

Connect, manage, and secure microservices.

Network of microservices that make up applications often called a service mesh.

- Traffic Management
- Observability
- Policies and enforcement
- Service identity and security

OCI BASED INNOVATIONS

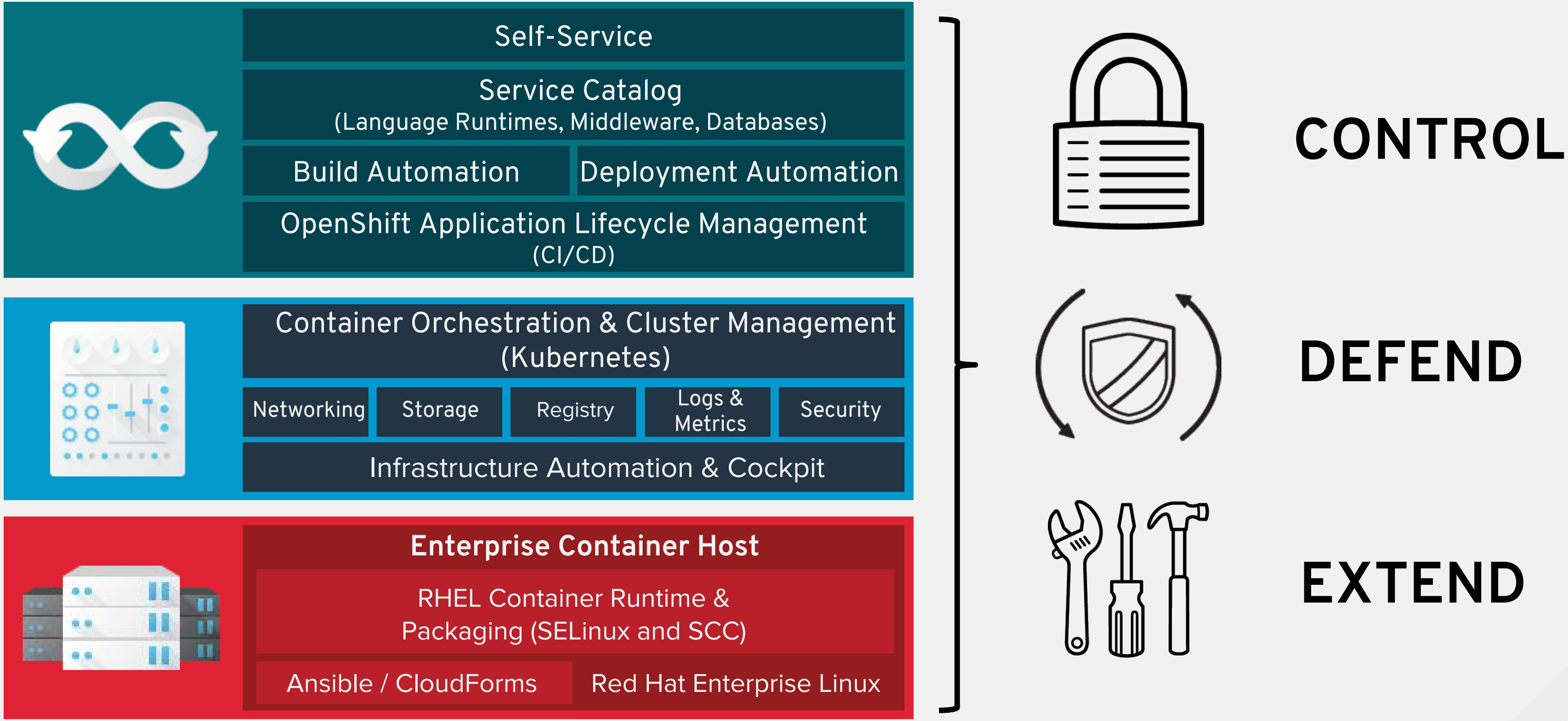
cri-o

- A lightweight, OCI-compliant container runtime designed for Kubernetes
- Runs any OCI / Docker container from any OCI / Docker registry
- Focus on stability and life cycle *with* the platform
- Improve container security & performance at scale

buildah

- OCI-compliant, daemon-less tool for building/modifying OCI/Docker images.
- Enables fine-grain control over the commands and content of each image layer
- Container host utils. can optionally be leveraged as part of the build
- Can use a Dockerfile
- Shares the underlying image and storage components with CRI-O

BRINGING IT ALL TOGETHER



Self-Service

Service Catalog
(Language Runtimes, Middleware, Databases)

Build Automation | Deployment Automation

OpenShift Application Lifecycle Management
(CI/CD)

Container Orchestration & Cluster Management
(Kubernetes)

Networking | Storage | Registry | Logs & Metrics | Security

Infrastructure Automation & Cockpit

Enterprise Container Host

RHEL Container Runtime &
Packaging (SELinux and SCC)

Ansible / CloudForms | Red Hat Enterprise Linux

CONTROL

DEFEND

EXTEND

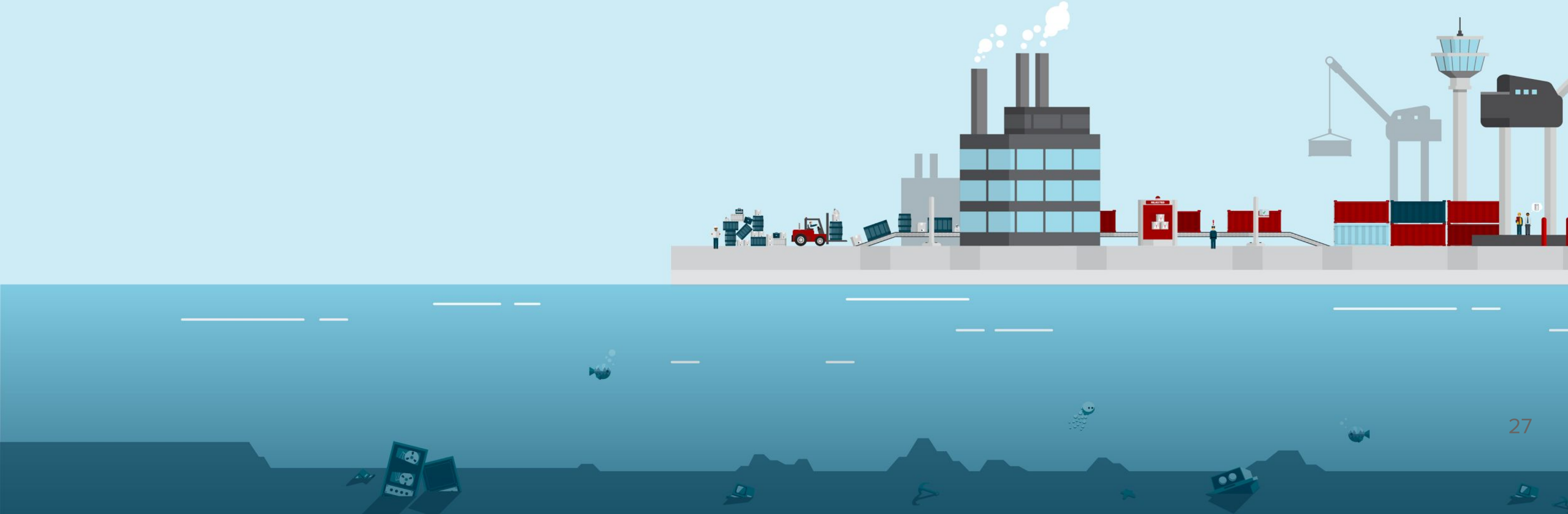
ADDITIONAL RESOURCES

Ten Layers of Container Security

Openshift Security Guide

Container Image Signing Integration Guide

THANK YOU



FURTHER READING

Container Image Signing

Verify provenance of images

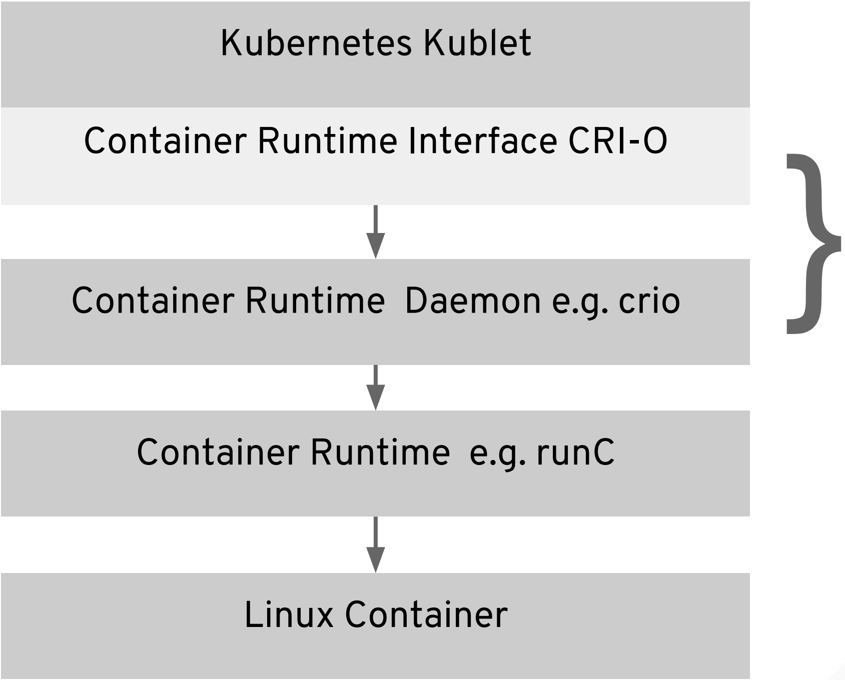
Registry independent

Supports multiple signatures

Enforce signatures at node level via signing trust policy

Supported in OpenShift v3.6 with improved integration on the roadmap

OCI CRI-O



Kubernetes Kublet

Container Runtime Interface CRI-O

Container Runtime Daemon e.g. crio

Container Runtime e.g. runC

Linux Container

OCI Compliance

SECURING THE OPERATIONS - LOGGING

EFK Stack (FEK?)

- ElasticSearch, Fluentd, Kibana
- Based on log aggregation
- Event system - all events container, system, kubernetes, captured by EFK and issues or errors
- Good for ad hoc analytics
- Good for post mortem forensics because of extensive log information

MONITORING: HAWKULAR

- REST API to store and retrieve availability, counter, and gauge measurements
- Visualization and alerting
- Application performance management
- Integration with ManageIQ (cloud mgmt)
- Most associated with large scale central IT teams with lots of apps

MONITORING: PROMETHEUS

- Time series data model identified by metric name and key/value pairs
- Collection happens via a pull model over HTTP
- Values reliability even under failure conditions over 100% accuracy
- Most associated with web-scale DevSecOps

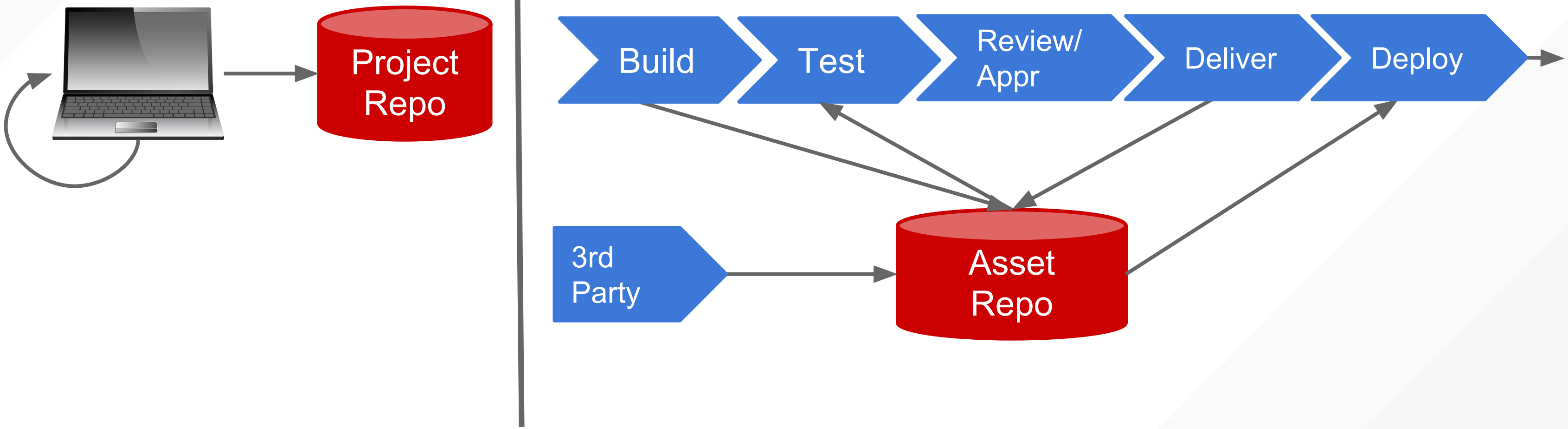
FUTURE OCI TOOLING

Developer
Workstation/Laptop

Container Platform
DataCenter

```
$ krane
```

Buildah
libpod
cri-o


Project
Repo
Build
Test
Review/
Appr
Deliver
Deploy
3rd
Party
Asset
Repo