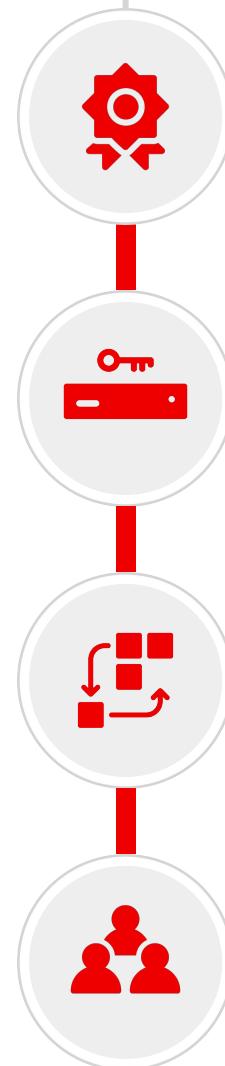




RED HAT QUAY TECHNICAL DECK

Detailed Feature Descriptions and Operational Guidance

- We start at 02:00 PM CEST



Industry-leading, **trusted**, and **open source** registry platform operating at scale since 2014

Built to **efficiently manage content** under governance and security **controls** globally

Runs **everywhere**, easy to **integrate** and **automate** but works best with **OpenShift**

Developed in **collaboration** with a broad open source, customer, and ecosystem **community**

Red Hat Quay Key Features

Massive Scale Testing Quay.io
Real Time Garbage Collection
Automated Squashing

SCALABILITY

Seamless Git Integration
Build Workers
Webhooks

BUILD AUTOMATION

Extensible API
Webhooks, OAuth
Robot Accounts

INTEGRATION

Vulnerability Scanning
Logging & Auditing
Notifications & Alerting

SECURITY

REGISTRY

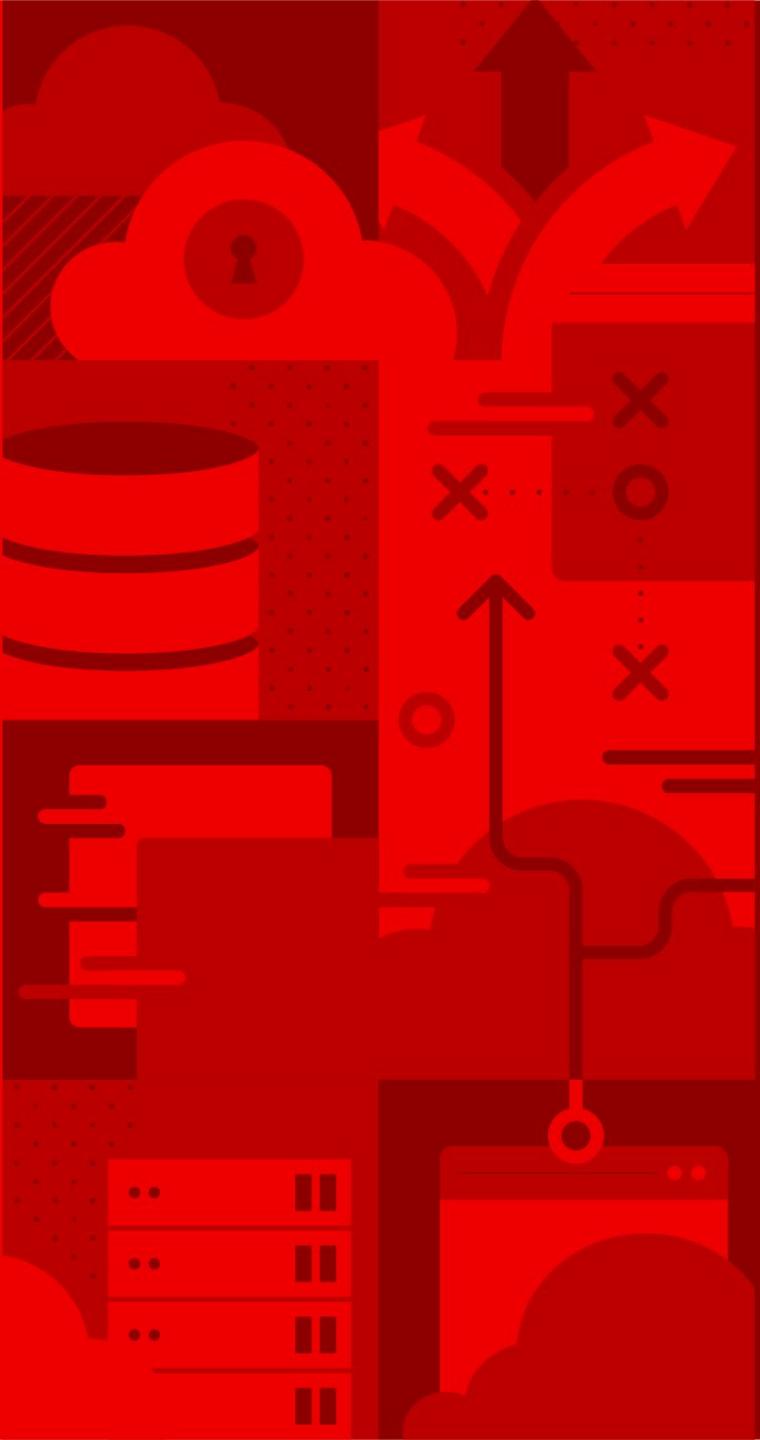
High Availability
Full Standards / Spec Support
Long-Term Protocol Support
Application Registry
Enterprise Grade Support
Regular Updates

CONTENT DISTRIBUTION

Geo-Replication
Repository Mirroring
Air-Gapped Environments

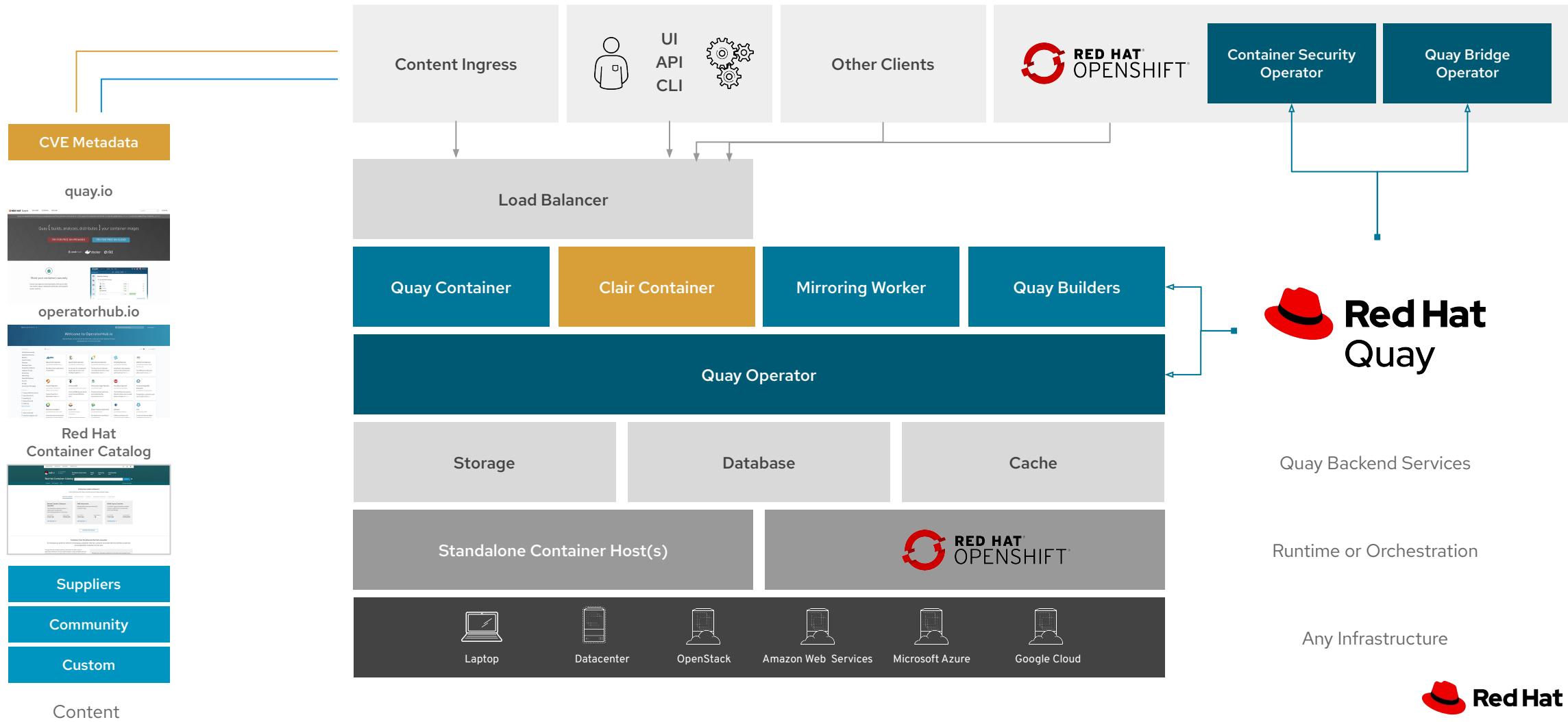
ACCESS CONTROL

Authentication Providers
Fine-Grained RBAC
Organizations & Teams

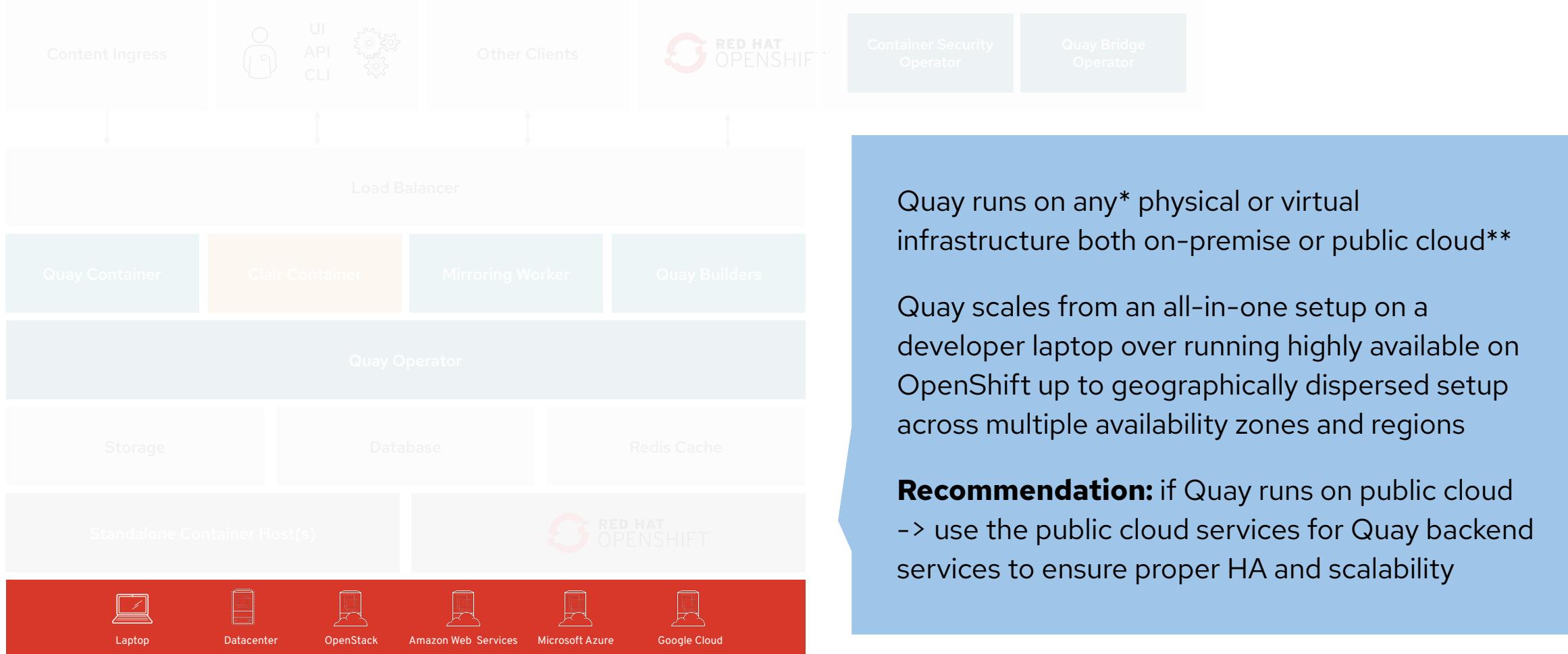


Quay Architecture

Red Hat Quay Architecture



Prerequisite: Infrastructure

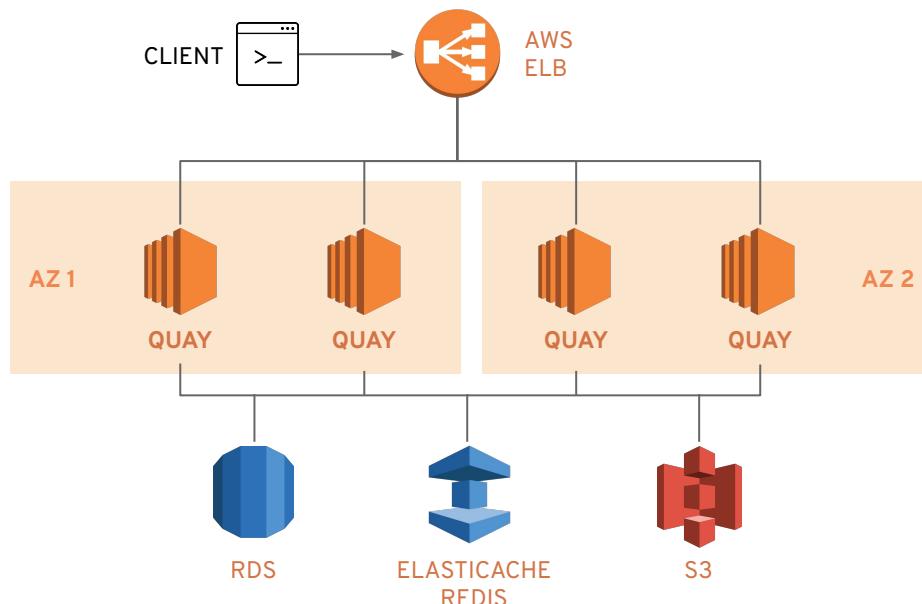


* Further details can be found in the Quay 3.x tested configuration matrix: <https://access.redhat.com/articles/4067991>

** Further details can be found in the Quay Support Policy: <https://access.redhat.com/support/policy/updates/rhquay/policies>

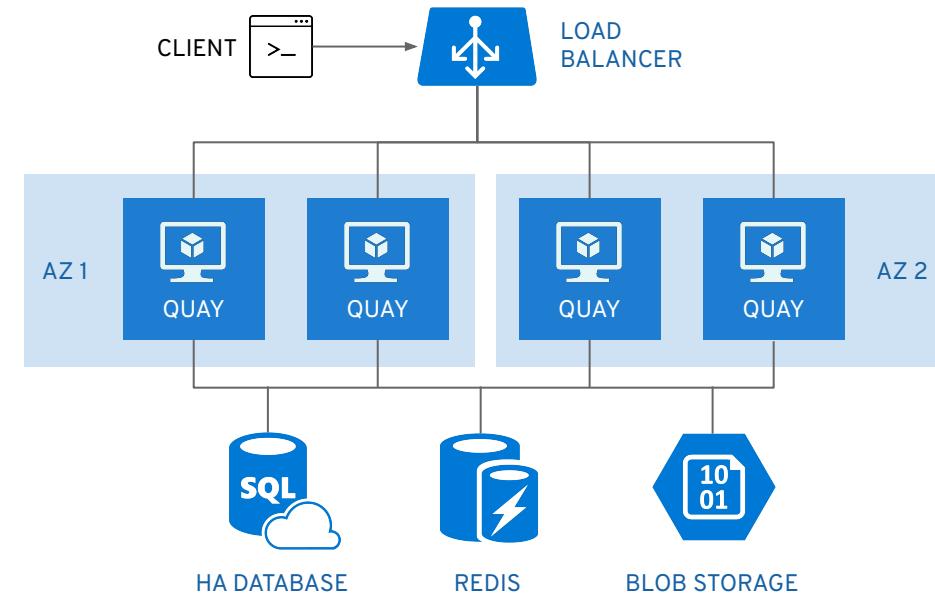
Running Red Hat Quay on Public Cloud

Full list of tested and supported configurations can be found inside the Red Hat Quay Tested Integrations Matrix: <https://access.redhat.com/articles/4067991>



If Quay runs on AWS you can use:

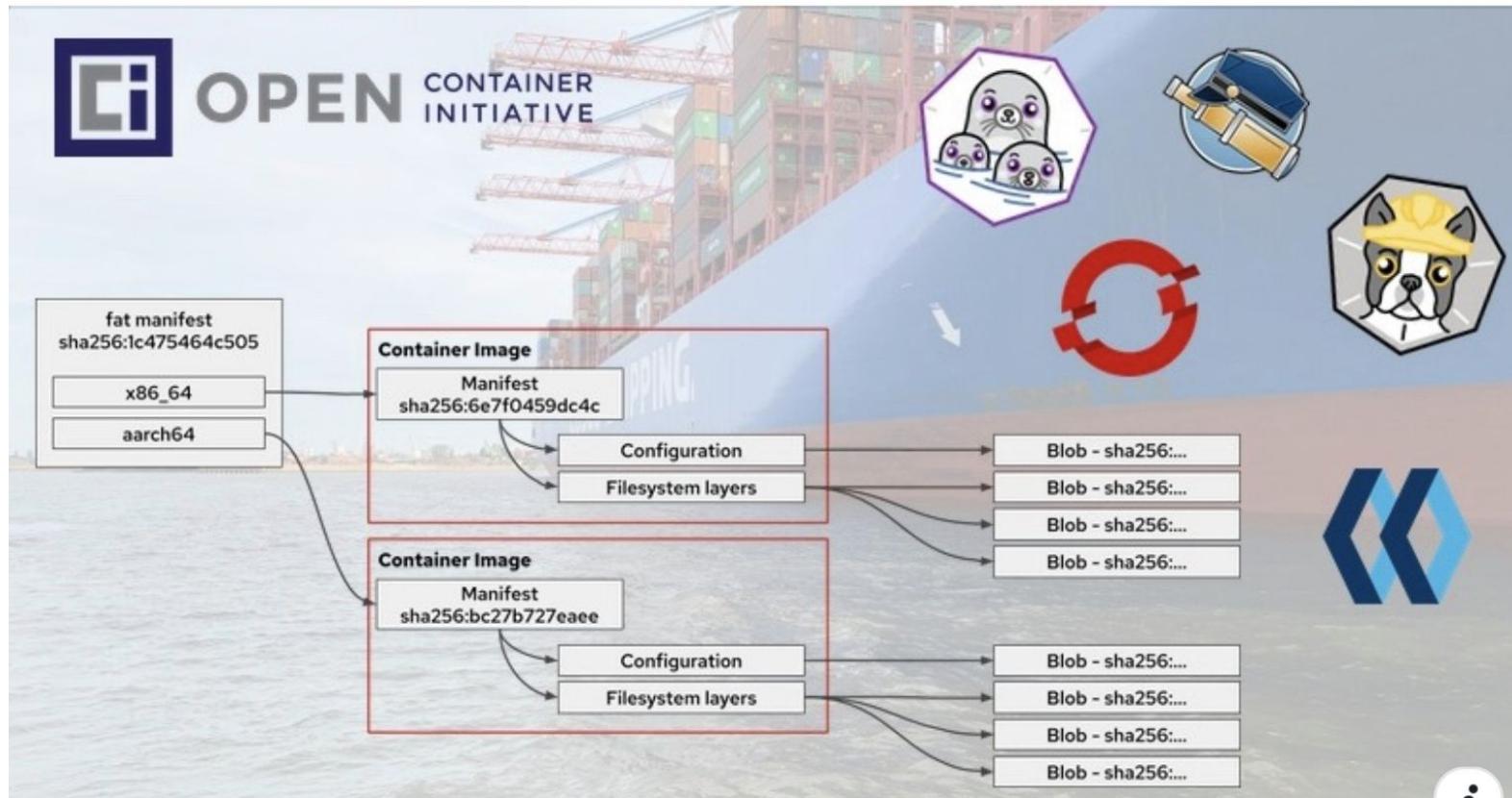
- AWS Elastic Load Balancer
- AWS S3 (hot) blob storage
- AWS RDS database
- AWS ElastiCache Redis
- EC2 VMs recommendation: M3.Large or M4.XLarge



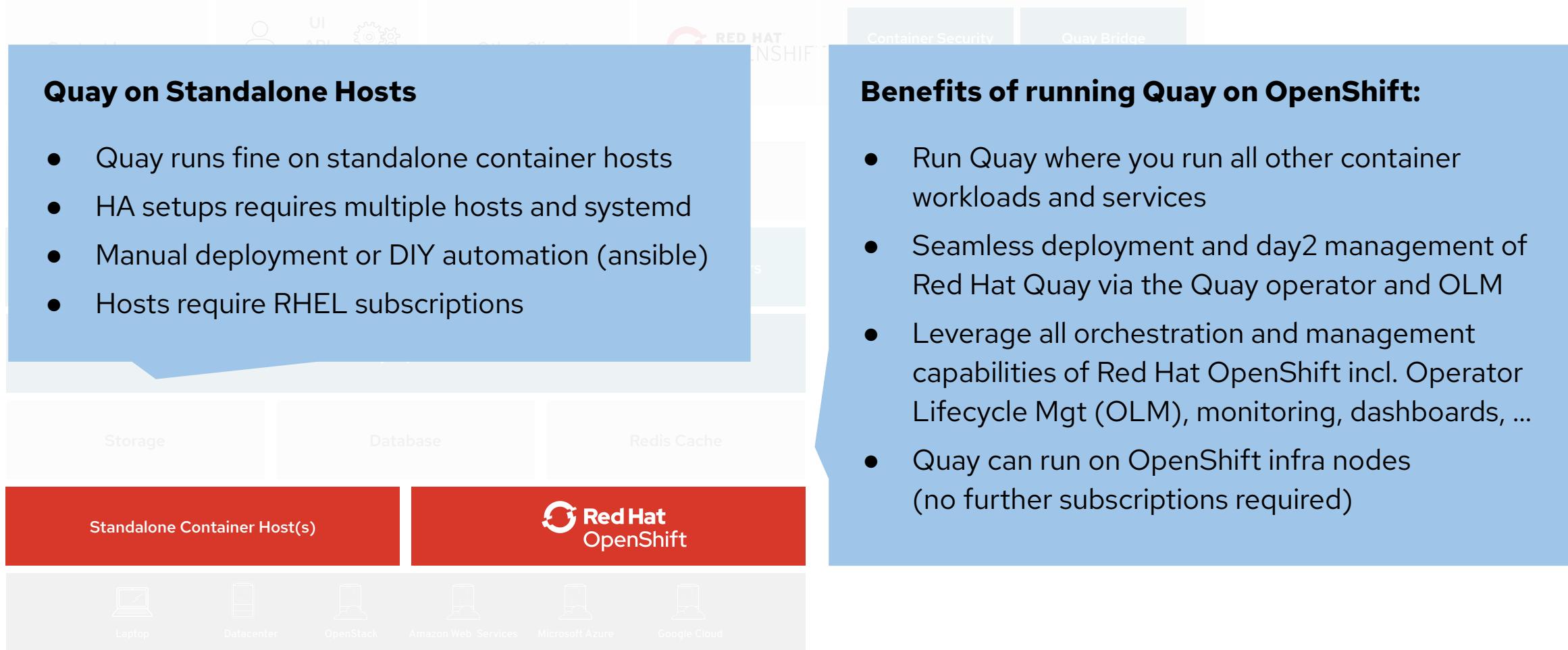
If Quay runs on MS Azure you can use:

- Azure managed services such as HA PostgreSQL
- Azure Blob Storage must be hot storage (not Azure Cool Blob Storage)
- Azure Cache for Redis

Multi Architecture Containers



Prerequisite: Container Runtime or Orchestration

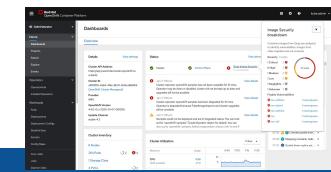


Red Hat Quay works best with OpenShift

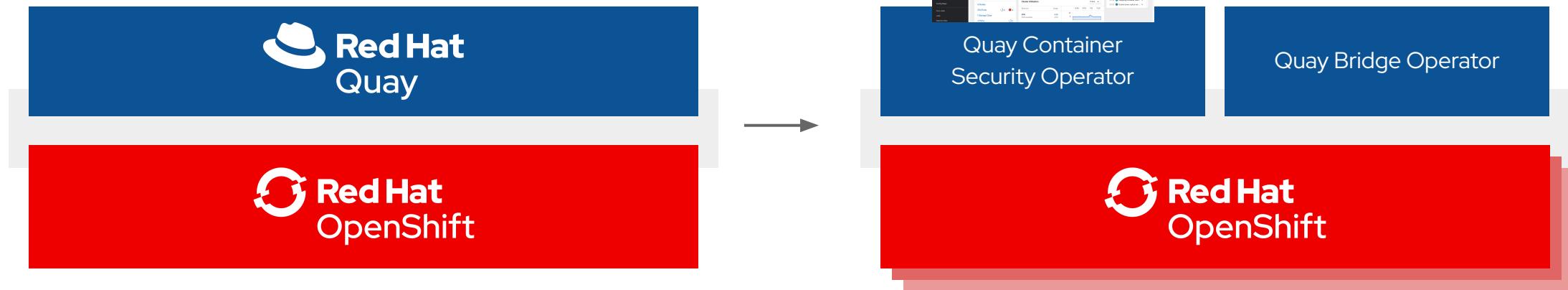
Red Hat Quay runs on any infrastructure
but **runs best on OpenShift**

The **Quay Operator** ensures seamless deployment
and management of Quay running on OpenShift

CSO brings Quay / Clair
vulnerability data into the
OpenShift Console



The **Quay Bridge Operator** ensures
seamless integration and
user experience for using
Quay **with** OpenShift



Quay serves content to **one or many OpenShift clusters**, wherever they're running.

With or without using the OpenShift internal registry but leveraging all OpenShift capabilities.

Benefits of running Quay on OpenShift



- **Zero to Hero** - Simplified deployment of Quay and associated components means that you can start using the product immediately
- **Scalability** - Leverage cluster compute capacity to manage expected demand
- **Simplified Networking** - Diverse ingress options using well established patterns for any application deployed on the platform
- **Centralized configuration management** - Configurations stored in etcd provide a centralized source of truth
- **Repeatability** - Consistency regardless of the number of replicas of Quay / Clair
- **Expanded Options** - Additional solutions that are specifically designed to take advantage of an OpenShift deployment

Quay Sizing Recommendations

- Scalability of Quay is one of its key strengths since the same code base runs on a developer laptop with a PoC sizing, as a typical mid-size deployment with ~2,000 users serving content to dozens of kubernetes clusters up to thousands of clusters world-wide (Quay.io)
- As for any other product there are no “typical sizing recommendations” since sizing heavily depends on a multitude of factors (no of users / images / concurrent pulls and pushes, etc.)
- **Stateless** components can be **scaled-out** (will cause more load on backend services though)
 - Auto-scaling on k8s deployments currently tech-preview, future via Quay operator
 - Note: Scaling out stateless components will add load to stateful components
- **Minimum** requirements as documented in the Quay Product Docs:
 - Quay: min 4GB, recommended 6GB, 2 or more vCPUs
 - Clair: recommended 2GB RAM, 2 or more vCPUs
 - Clair database requirements for security metadata: min 200MB
 - Storage depends on no of images, recommended min 30GB

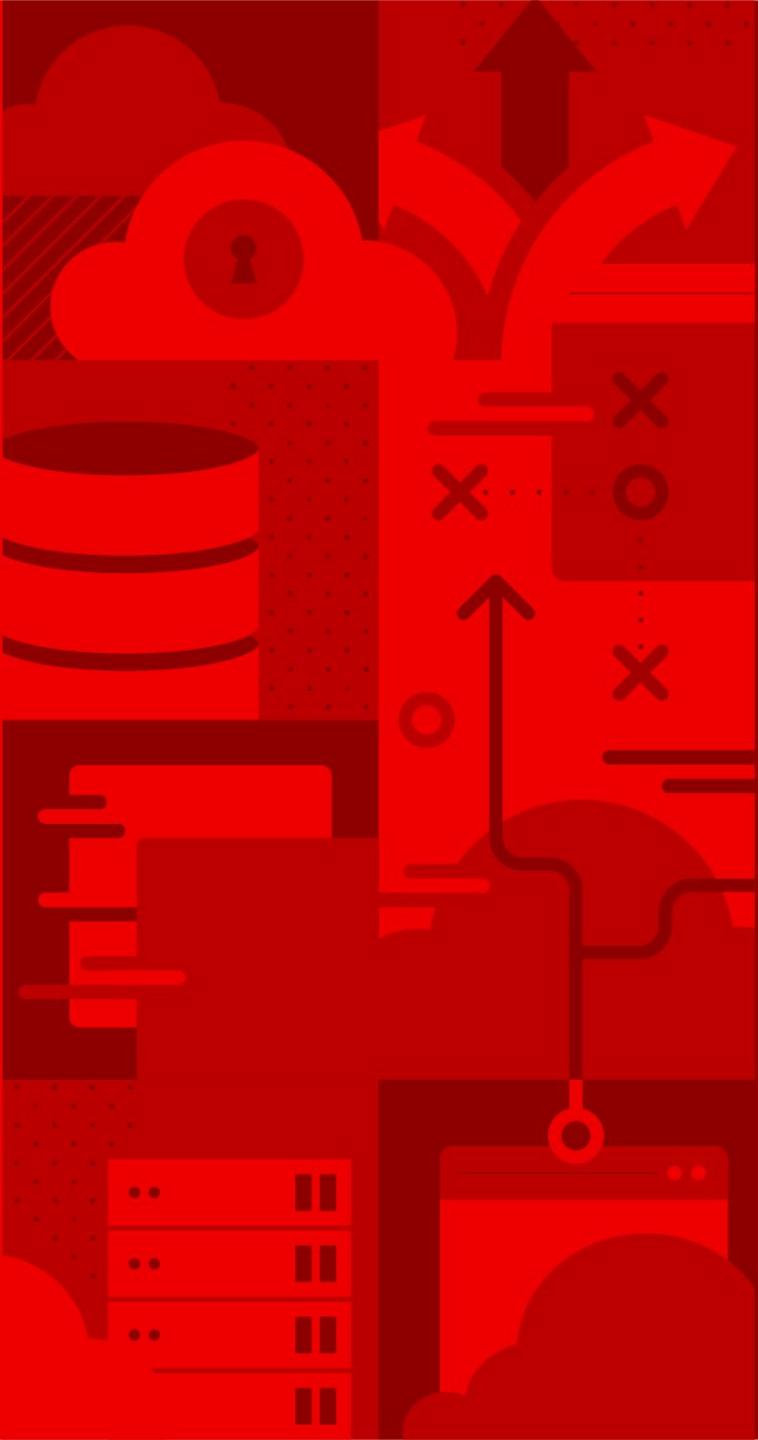
Quay Sample Sizings

Note: Those are sample sizings of existing Quay deployments. Whether a specific deployment runs fine with the same metrics depends on too many other factors as well not shown here.

Metric	Minimum Setup	Mid/Large Setup	XXXXL (Quay.io)
No of Quay containers by default	1	4	15
No of Quay containers max at scale-out	N/A	8	30
No of Clair containers by default	1	3	10
No of Clair containers max at scale-out	N/A	6	15
No of mirroring pods ¹ (to mirror 100 repos)	1	5-10	N/A
Database sizing		4-8 Cores / 6-32 GB RAM	32 cores 244GB, 1+ TB disk
Storage Backend Sizing	10-20 GB	1 - 20 TB	50+ TB up to PB
Redis Cache Sizing ²		2 Cores / 2-4 GB RAM	4 cores / 28 GB RAM
Underlying node sizing (phys or virtual)	2-4 Cores / 6 GB RAM	4-6 Cores, 12-16 GB RAM	Quay: 13 cores 56GB RAM Clair: 2 cores 4 GB RAM

¹ see repository mirroring section for further details on sizing & related recommendations

² since Redis cache is only used for Quay builders the sizing can be very tiny if builders aren't used



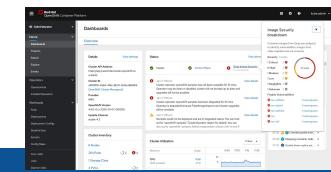
Quay and OpenShift

Red Hat Quay works best with OpenShift

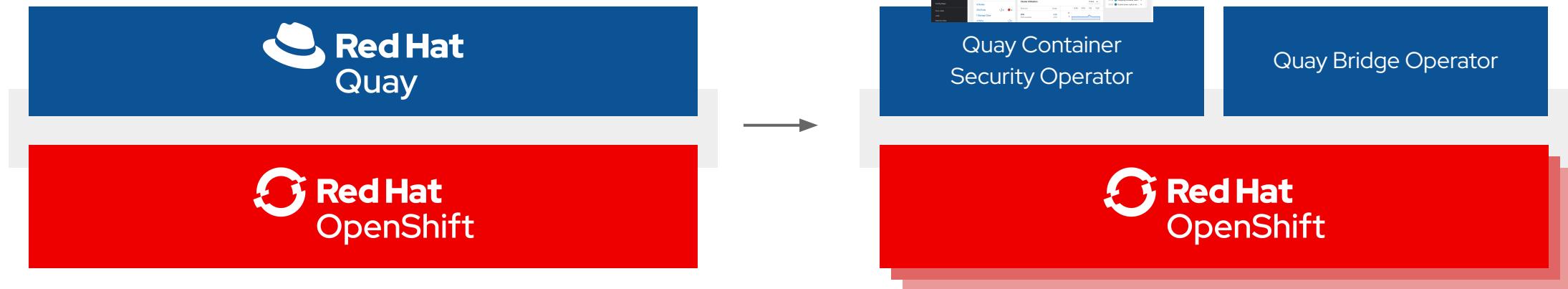
Red Hat Quay runs on any infrastructure
but **runs best on OpenShift**

The **Quay Operator** ensures seamless deployment
and management of Quay running on OpenShift

CSO brings Quay / Clair
vulnerability data into the
OpenShift Console



The **Quay Bridge Operator** ensures
seamless integration and
user experience for using
Quay **with** OpenShift



Quay serves content to **one or many OpenShift clusters**, wherever they're running.

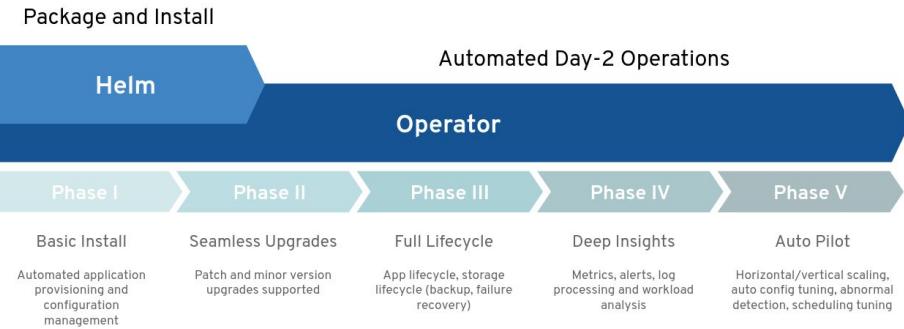
With or without using the OpenShift internal registry but leveraging all OpenShift capabilities.

Benefits of running Quay on OpenShift



- **Zero to Hero** - Simplified deployment of Quay and associated components means that you can start using the product immediately
- **Scalability** - Leverage cluster compute capacity to manage expected demand
- **Simplified Networking** - Diverse ingress options using well established patterns for any application deployed on the platform
- **Centralized configuration management** - Configurations stored in etcd provide a centralized source of truth
- **Repeatability** - Consistency regardless of the number of replicas of Quay / Clair
- **Expanded Options** - Additional solutions that are specifically designed to take advantage of an OpenShift deployment

Quay - Focus on Operators



Focus and direction for the Quay product are kubernetes operators and running Quay on OpenShift / kubernetes given the advantages of operators compared with its alternatives

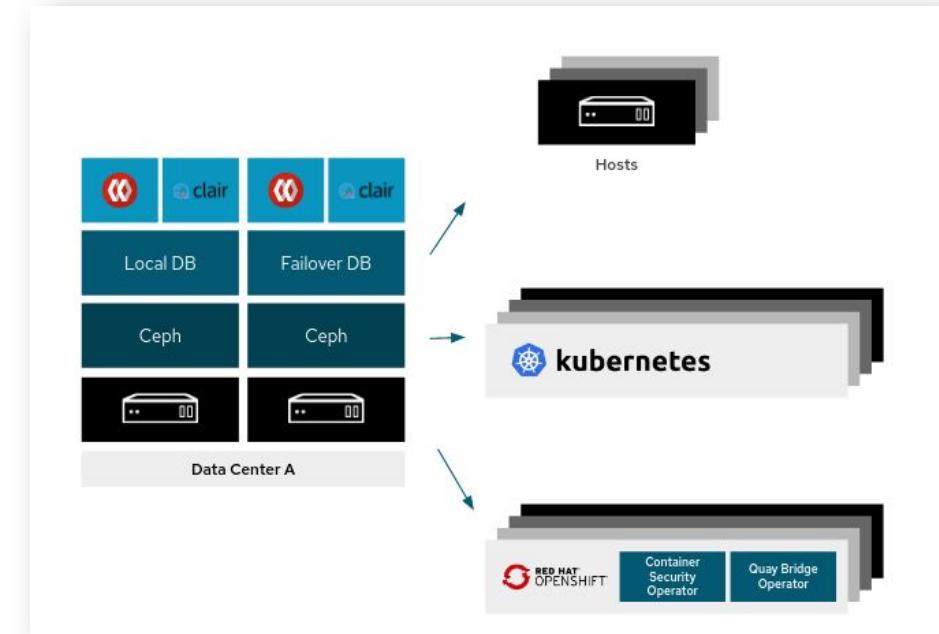
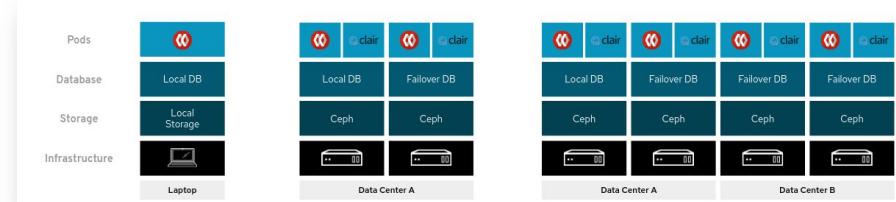
Maintaining another deployment and management tooling for non-k8s deployments is not feasible and not aligned to our prioritization and roadmap (Quay v4 will run on k8s by default)

The screenshot shows the Red Hat OpenShift Container Platform interface. The top navigation bar includes the Red Hat logo, 'OpenShift Container Platform', and user information: 'Administrator' and 'Project: all projects'. A search bar at the top right contains the text 'quay'. The left sidebar is titled 'Operators' and lists several categories: Home, Overview, Projects, Search, Explore, Events, and OperatorHub (which is currently selected). Under OperatorHub, there are sections for Installed Operators, Workloads, Networking, Storage, and Builds. The main content area displays three operator cards:

- Quay** (Community): Red Hat Quay is a private container registry that stores, builds, and deploys container...
- Quay Bridge Operator**: Enhance OCP using Red Hat Quay container registry
- Red Hat Quay** (provided by Red Hat): Red Hat Quay is a private container registry that stores, builds, and deploys container...

Quay Deployment Examples

- Quay can run on standalone container hosts or OpenShift (recommended)
- A Quay deployment can be distributed across multiple DCs or even OCP clusters (geo-repl)
- Typically Quay is used for **more than one / many OpenShift clusters**
- Components which can run **on-cluster**: Quay, Clair, mirroring workers
- Components which should / must run **off cluster** (today): Quay builders, databases (if not an operator), storage



Quay Builders on OpenShift

- Quay builders require a docker runtime and do not work with buildah yet
- As of today (Quay 3.3) the Quay builders can't run on OpenShift 3 + 4 and therefore should run off-cluster (also for security reasons)
- Preferably on bare metal due to performance reasons
- Technically Quay builders can run on OCP 4 bare metal, documentation and a small enhancement of the Quay config app targeted for Quay 3.4

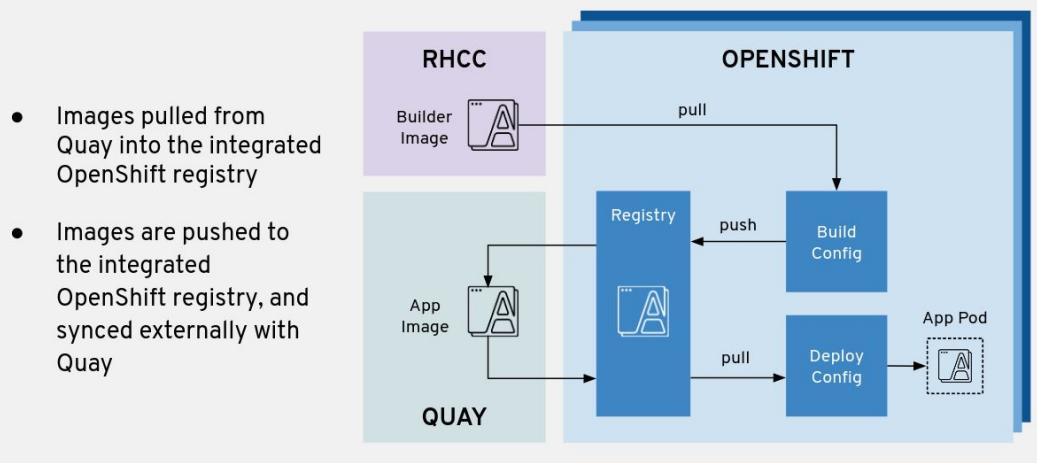
OpenShift and External Registries

- OpenShift can utilize an external container registry as a source for operations on the platform
 - Build source and output
 - Runtime content
- From an OpenShift point of view, Quay as with any external registry is not as deeply integrated as the OpenShift internal registry
 - No automatic RBAC isolation based on OpenShift cluster permissions
 - No real-time automatic ImageStream notifications and updates

Using Quay With or Without Internal Registry

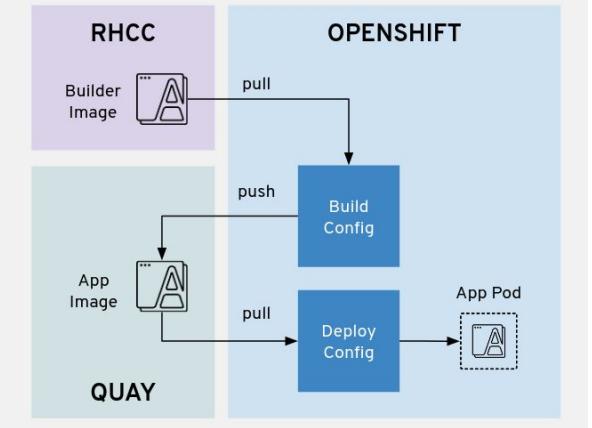
- Quay can be used as an external registry in front of an entire OpenShift cluster with its registry
- Quay also can be used directly without using the internal registry which requires a couple of changes (secrets, build and deployment configs) which are **partially** done automatically by QBO

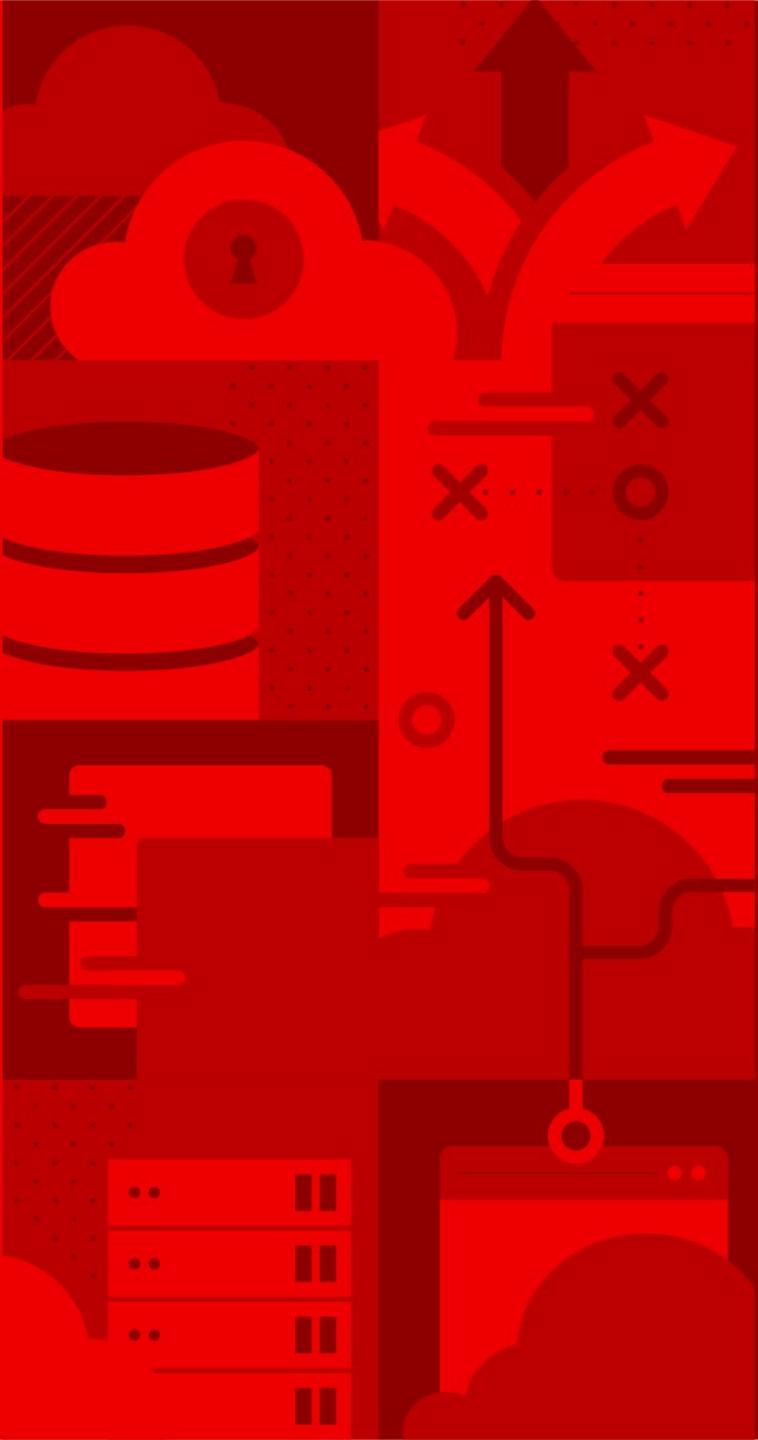
Quay as Upstream Registry with OpenShift



Quay as OpenShift Registry

- Images are pushed directly by builds to Quay
- Images are pulled directly from Quay

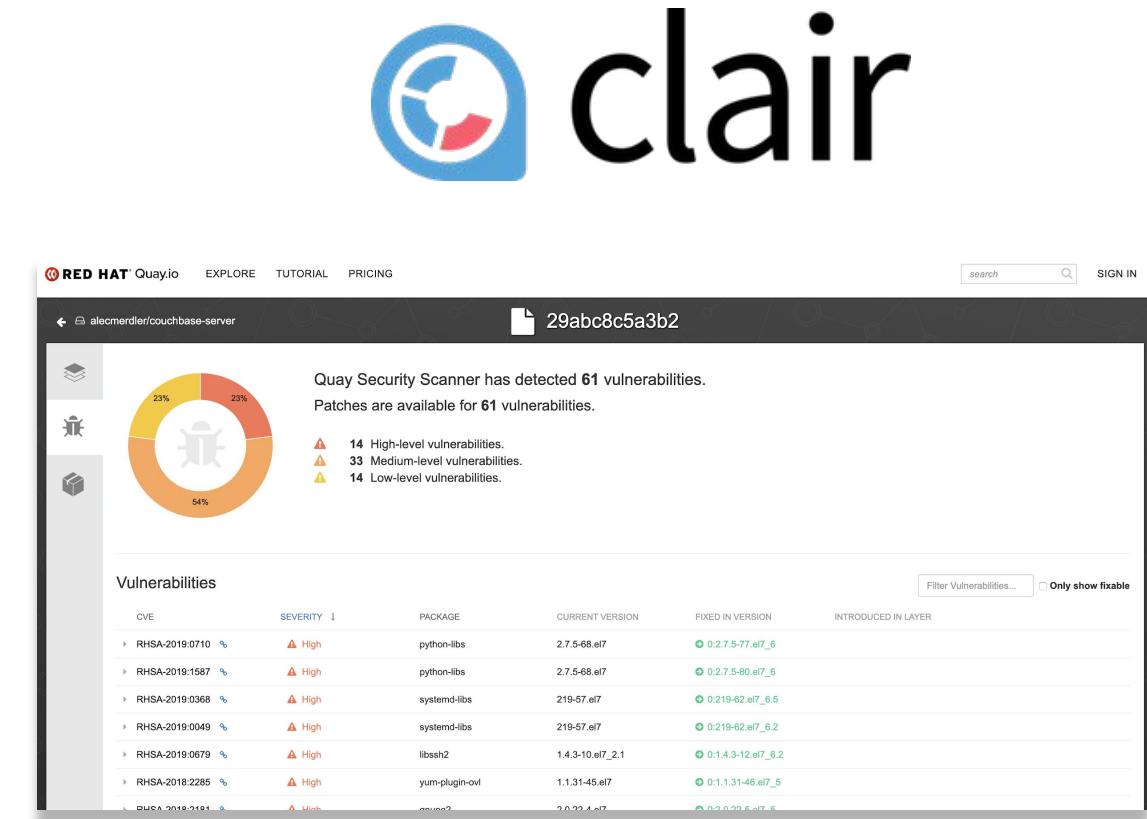




Built-In Vulnerability Scanning via Clair

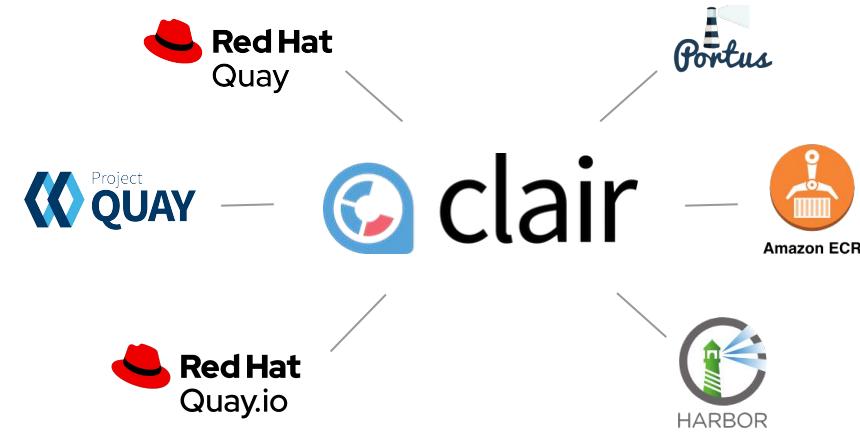
Clair Overview

- Clair is an open source tool for static analysis of vulnerabilities in application containers
- Developed by CoreOS for Quay and it's massive scale usage at Quay.io
- Used by various other projects and third party products
- Upstream Repositories:
<https://github.com/quay/clair>





The screenshot shows the Red Hat Quay interface for a Python image. It displays a pie chart of vulnerabilities by severity (High, Medium, Low, Negligible, Unknown) and a detailed table of specific CVEs and their fixes across various packages like systemd, libssh2, apt, and glibc.



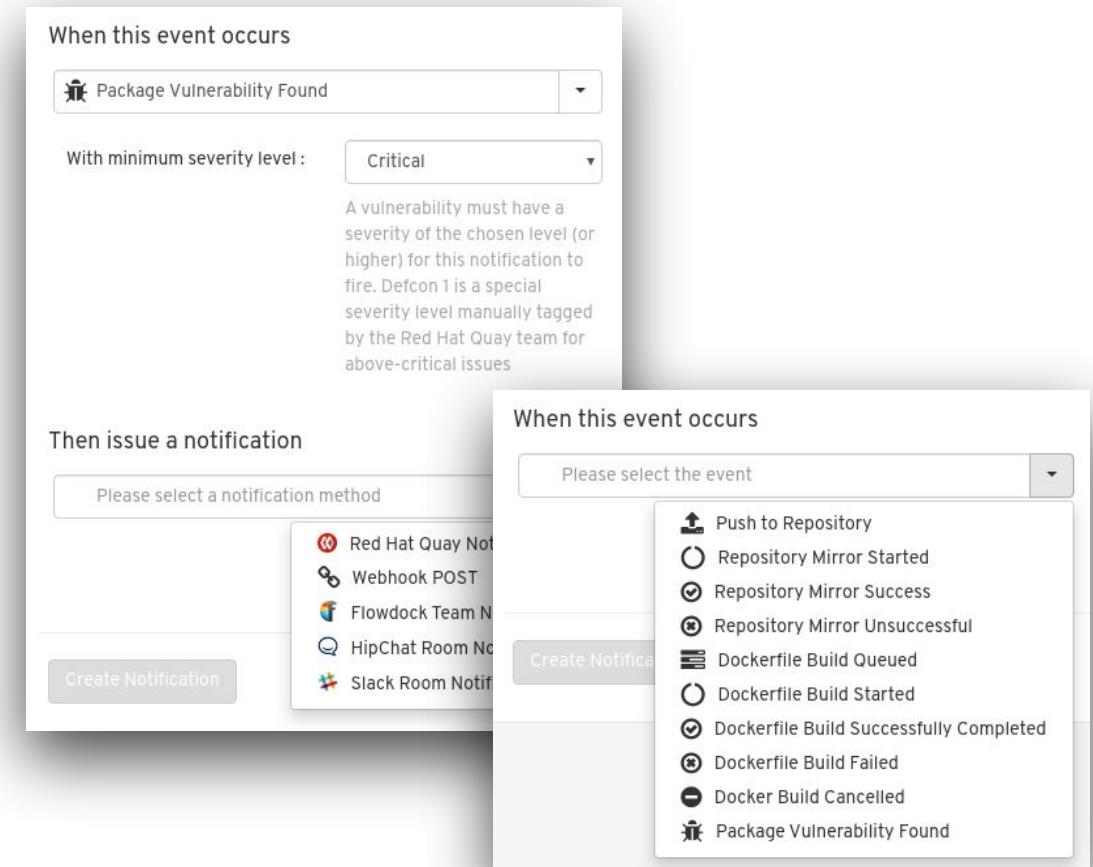
Clair v4 (Tech Preview with Quay 3.3)

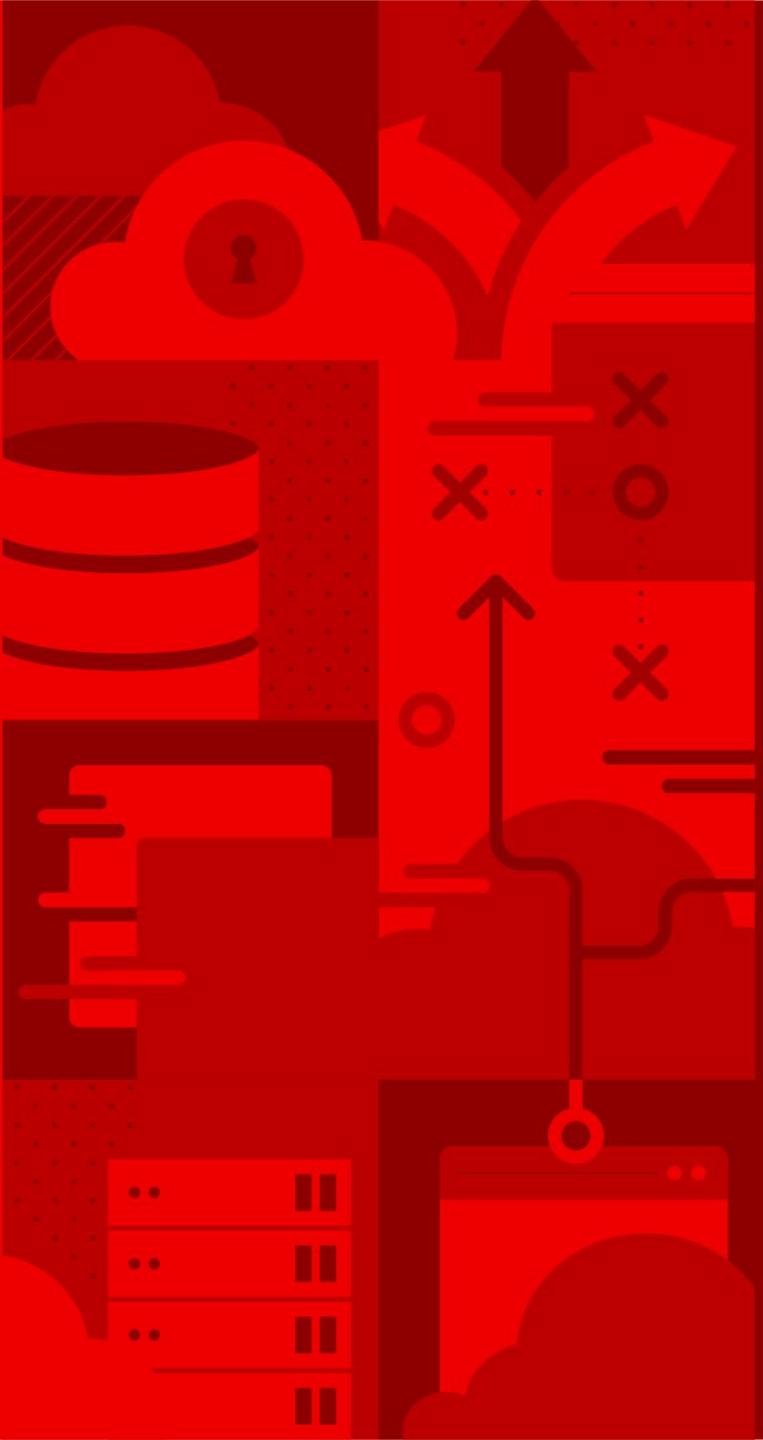
Clair v4 is the newest version of Clair after a massive refactoring in order to make several big enhancements possible. This includes:

- Support for programming language package managers (3.3: python)
- immutable data model & new manifest-oriented API
- Refocus on latest container specifications (OCI) (Content addressability)

Notifications for Vulnerabilities found by Clair

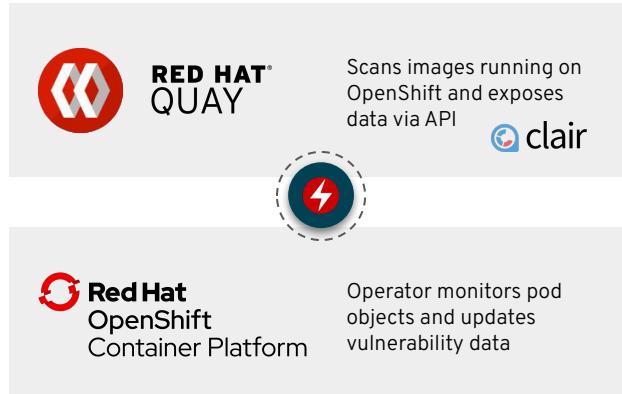
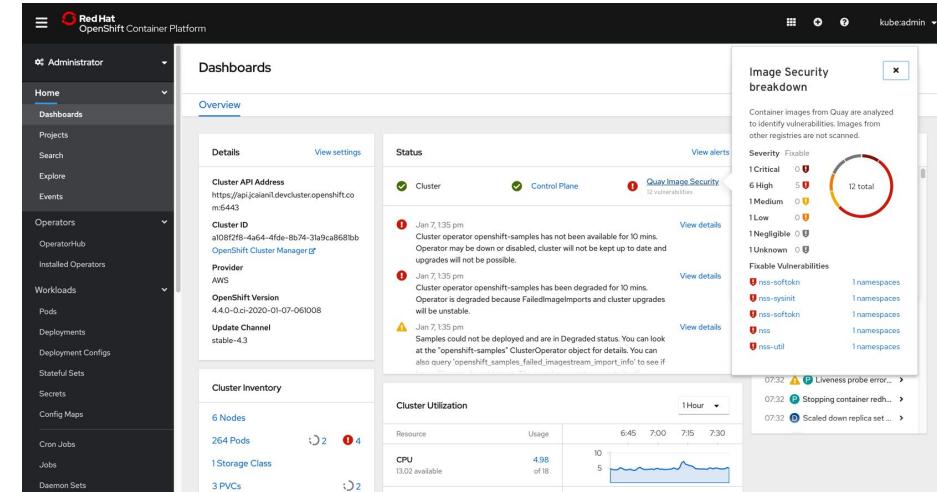
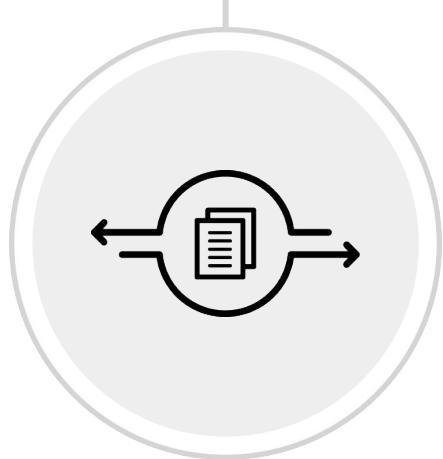
- **Quay triggers different notifications for various repository events** (depends on enabled features)
- This includes the event type “**Package Vulnerability Found**”
- Additional Filter can be applied for **Severity Level**
- **Various Notification Methods**
- Custom Notification Title (optional)





Container Security Operator and OpenShift Console Integration

Quay Container Security Operator (CSO)



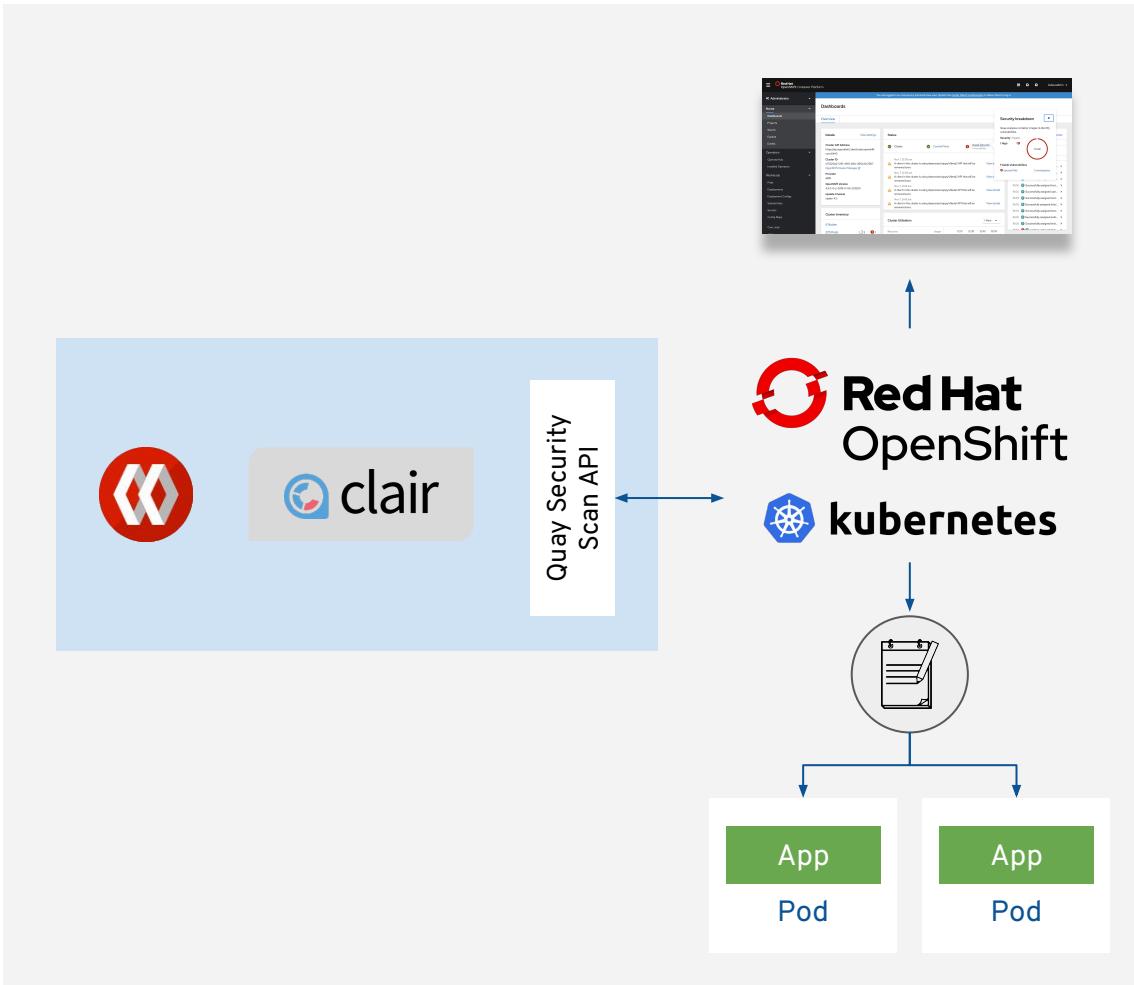
Container Security Operator - Vulnerability Data in OpenShift

Operator which runs on OpenShift and fetches vulnerability from Quay / Clair if Kubernetes pod objects change

Synchronous Updates of vulnerability information

Prerequisite to leverage / show vulnerability data in OpenShift Console

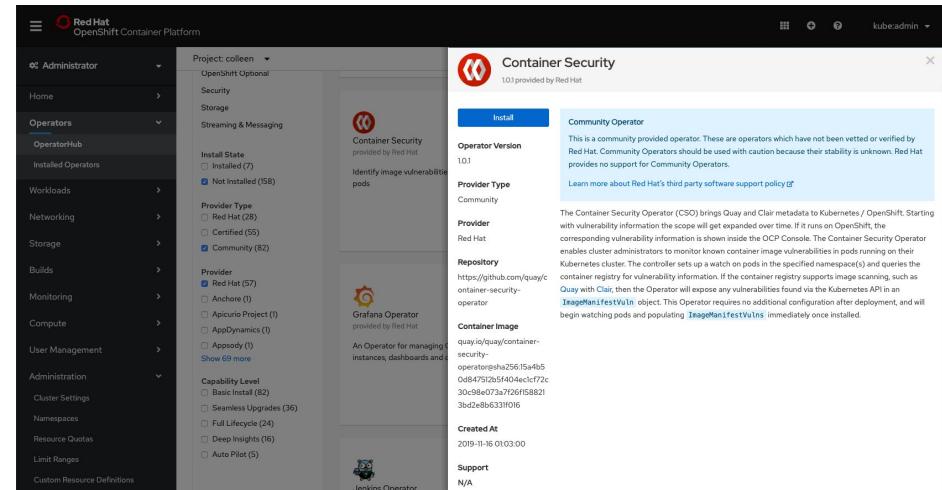
Container Security Operator (CSO)



- Container Security Operator (CSO) runs on OpenShift and watches pod objects
- Pod object changes triggering a data fetch from Quay/Clair and stores vulnerability information in CRs (by image manifest ID)
- CRs gets deleted if pod gets deleted
- Configurable interval to update vulnerability data from Quay / Clair (default: 5min)
- Data available via k8s CLI / APIs
- Supposed to be used by partner security products as well (consistent data ingress)

How to deploy the Container Security Operator (CSO)

- CSO supposed to run on all OCP clusters Quay is serving images to (not limited to the cluster Quay is running on)
- CSO available in OpenShift embedded operatorhub (upstream version in [operatorhub.io](#))
- Deployment via Operator Lifecycle Manager to ensure that OLM takes care of RBAC permissions, dependency resolution and automatic upgrades
- Works with both Red Hat Quay and Quay.io



```
securityLabeler:
  host: # Leave empty to use in-cluster config
  prometheusAddr: "0.0.0.0:8081"
  interval: 1m
  workers: 1
  labelPrefix: seccscan # Security labels' "namespace"
  namespaces: # List of namespaces to label in the cluster
    - default
    - dev
securityScanner:
  host: "https://quay.mycompany.com"
  apiVersion: 1
  type: "Quay"
```

OpenShift Console Integration via CSO

Image Manifest Vuln Details

Quay Security Scanner has detected 7 vulnerabilities.
Patches are available for 7 vulnerabilities.
7 High vulnerabilities.

Name	Namespace	Created
imc-controller-69c477c85-vg77q	knative-eventing	8 hours ago

Image Manifest Vulnerabilities

Image Name	Namespace	Highest Severity	Affected Pods	Fixable	Manifest
openshift/knative-knative-eventing-channel-controller	knative-eventing	High	1	7	08aed83c1bb
openshift/knative-knative-eventing-sources-controller	knative-eventing	High	1	7	32f3ca637fc
openshift/knative-knative-eventing-controller	knative-eventing	High	1	7	cc4ec0d71b1

Vulnerabilities

Vulnerability	Severity	Package	Current Version	Fixed in Version
RHSA-2019:4190	High	nss-softoken-freebl	3.44.0-5.el7	0:3.44.0-8.el7_7
RHSA-2019:4190	High	nss-util	3.44.0-3.el7	0:3.44.0-4.el7_7
RHSA-2019:4190	High	nss-tools	3.44.0-4.el7	0:3.44.0-7.el7_7
RHSA-2019:4190	High	nss-softoken	3.44.0-5.el7	0:3.44.0-8.el7_7

List view - easily view vulnerabilities of the images

- Highest severity
- Number of affected pods
- Number fixable
- Manifest SHA external link for viewing the vulnerability in Quay

Details view - see a list of vulnerabilities of an image

- Vulnerability info, severity, package, current package, and the fixed version.

Affected Pods tab - easy access to the affected pods to quickly update with the fixes.

Learn more about the new views we added to the OpenShift Console with OCP 4.4:
<https://blog.openshift.com/openshift-4-4-not-on-my-watch-image-vulnerabilities-list/>

Vulnerability Data inside the OpenShift Console

The screenshot shows the 'Image Manifest Vulnerabilities' section within the OpenShift Console. The left sidebar contains navigation links for Home, Operators, Workloads, Networking, Storage, Builds, Monitoring, Compute, User Management, and Administration. The main area displays a table of vulnerabilities with the following columns: Image Name, Namespace, Highest Severity, Affected Pods, Fixable, and Manifest. A search bar labeled 'Filter by name...' is located at the top right of the table area.

Image Name	Namespace	Highest Severity	Affected Pods	Fixable	Manifest
VULN alecmerdler/bad-pod	NS default	⚠ Medium	1	0	35c1c5688e7 ↗
VULN alecmerdler/bad-image	NS skynet	⚠ Unknown	1	1	4bc210f89d7 ↗
VULN alecmerdler/bad-pod	NS default	❗ Critical	1	0	7d4aae77622 ↗
VULN 3scale/3scale-operator	NS default	❗ High	1	24	9a6536efbb5 ↗
VULN alecmerdler/bad-image	NS skynet	⚠ Unknown	1	1	b025832c073 ↗
VULN alecmerdler/bad-pod	NS default	⚠ Low	1	0	e94c22ba519 ↗
VULN alecmerdler/bad-pod	NS default	❗ Defcon 1	1	0	f4cd12ac979 ↗

ImageManifestVuln list view

Vulnerability Data inside the OpenShift Console

CVE	Severity	Package	Current Version	Fixed in Version
RHSA-2019-4190	High	nss-softokn	3.36.0-5.el7_5	0:3.44.0-8.el7_7
RHSA-2019-4190	High	nss-sysinit	3.36.0-7.el7_6	0:3.44.0-7.el7_7
RHSA-2019-4190	High	nss-softokn-freebl	3.36.0-5.el7_5	0:3.44.0-8.el7_7
RHSA-2019-4190	High	nss-util	3.36.0-11.el7_6	0:3.44.0-4.el7_7
RHSA-2019-4190	High	nss	3.36.0-7.el7_6	0:3.44.0-7.el7_7
RHSA-2019-4190	High	nss-tools	3.36.0-7.el7_6	0:3.44.0-7.el7_7
RHSA-2019-2237	Medium	nss-softokn	3.36.0-5.el7_5	0:3.44.0-5.el7
RHSA-2019-2237	Medium	nss-sysinit	3.36.0-7.el7_6	0:3.44.0-4.el7
RHSA-2019-2237	Medium	nss-softokn-freebl	3.36.0-5.el7_5	0:3.44.0-5.el7
RHSA-2019-2237	Medium	nss-util	3.36.0-11.el7_6	0:3.44.0-3.el7
RHSA-2019-2304	Medium	openssl-libs	1:1.0.2k-16.el7_6.1	1:1.0.2k-19.el7
RHSA-2019-2118	Medium	glibc-common	2.17-260.el7_6.4	0:2.17-292.el7

ImageManifestVuln detail view

Vulnerability Data inside the OpenShift Console

The screenshot shows the OpenShift console interface. On the left is a dark sidebar menu with options: Home, Operators, Workloads, Networking, Storage, Builds, Monitoring, Compute, and User Management. The 'Workloads' section is expanded, showing 'Overview', 'YAML', and 'Affected Pods'. The 'Affected Pods' tab is selected, indicated by an underline. The main content area displays a table of affected pods. The table has columns: Name, Namespace, and Created. A single row is shown: '3scale-operator-7864b9bb5d-frhnt' in the Name column, 'default' in the Namespace column, and '11 days ago' in the Created column. There is also a small icon next to the pod name. At the top of the main content area, there is a breadcrumb navigation: 'ImageManifestVuln > ImageManifestVuln Details'. Below the breadcrumb, the text 'VULN 3scale/3scale-operator@9a6536efbb5' is displayed. A search bar at the bottom right of the main content area is labeled 'Filter by name...'. The overall theme is red and white.

Name	Namespace	Created
3scale-operator-7864b9bb5d-frhnt	default	11 days ago

ImageManifestVuln detail view (affected pods)

Vulnerability Data inside the OpenShift Console

The screenshot shows the OpenShift Console interface. On the left is a dark sidebar with a navigation menu:

- Administrator
- Home
- Operators
- Workloads
- Networking
- Storage
- Builds
- Monitoring
- Compute
- User Management

Next to the sidebar, the main content area has a header "Project: default". Below it, a breadcrumb path shows "ImageManifestVuln > ImageManifestVuln Details". A prominent blue button labeled "VULN" contains the text "3scale/3scale-operator@9a6536efbb5". The page title is "Affected Pods". There are three tabs: "Overview", "YAML", and "Affected Pods", with "Affected Pods" being the active tab. A search bar at the top right says "Filter by name...". The main content is a table listing a single pod:

Name	Namespace	Created
3scale-operator-7864b9bb5d-frhnt	default	11 days ago

Kebab action on Pods list view

OpenShift Console Vulnerability Information Enhancements

Project: knative-eventing

ImageManifestVuln > ImageManifestVuln Details

IMV openshift-knative/knative-e-evening-channel-controller@08aed83clbb

Details **YAML** **Affected Pods**

Image Manifest Vuln Details

Quay Security Scanner has detected 7 vulnerabilities.
Patches are available for 7 vulnerabilities.
7 High vulnerabilities.

7 total

Name	Registry
sha256:08aed83clbb9f510d0f2c4dc64993fa333bad32d90bc08e4fcfc82fb	quay.io/openshift/knative/knative-e-evening-channel-controller

Namespace knative-eventing

Labels knative-eventing/imc-controller-69c4775c85-vg77q=true

Annotations 0 Annotations

Created At Mar 23, 4:40 pm

Owner No owner

Vulnerabilities

Vulnerability	Severity	Package	Current Version	Fixed in Version
RHSA-2019-4190	High	nss-softoken-freebl	3.44.0-5.el7	0.344.0-8.el7_7
RHSA-2019-4190	High	nss-util	3.44.0-3.el7	0.344.0-4.el7_7
RHSA-2019-4190	High	nss-tools	3.44.0-4.el7	0.344.0-7.el7_7
RHSA-2019-4190	High	nss-softoken	3.44.0-5.el7	0.344.0-8.el7_7
RHSA-2019-4190	High	nss	3.44.0-4.el7	0.344.0-7.el7_7
RHSA-2019-4190	High	nss-sysinit	3.44.0-4.el7	0.344.0-7.el7_7
RHSA-2020-0227	High	sqlite	3.717-8.el7	0.3717-8.el7_71

Project: knative-eventing

Image Manifest Vulnerabilities

Filter by name...

Image Name	Namespace	Highest Severity	Affected Pods	Fixable	Manifest
IMV openshift-knative/knative-e-evening-channel-controller	NS knative-eventing	High	1	7	08aed83clbb
IMV openshift-knative/knative-e-evening-sources-controller	NS knative-eventing	High	1	7	32f3ca637fd
IMV openshift-knative/knative-e-evening-controller	NS knative-eventing	High	1	7	cc4ec0d71b8
IMV openshift-knative/knative-e-evening-webhook	NS knative-eventing	High	1	7	e3bb2c01ddf

Learn more about the new views which have been added to the OpenShift Console as part of OCP 4.4:
<https://www.openshift.com/blog/4.4-openshift-not-on-my-watch-image-vulnerabilities-list>

Try it out!

<https://access.redhat.com/products/red-hat-quay>

SUBSCRIPTIONS DOWNLOADS CONTAINERS SUPPORT CASES

Red Hat CUSTOMER PORTAL

Products & Services Tools Security Community

Red Hat Quay

Products & Services > Red Hat Quay

WHAT'S NEW GET STARTED KNOWLEDGE SUPPORT

Red Hat Quay 3

Release Notes ▶

News

Introducing Red Hat Quay V3: A container registry tailored for the enterprise
2019-06-19T11:28:20+00:00

Check out the Quay datasheet on redhat.com
2018-11-09T15:35:22+00:00

Red Hat® Quay is a secure, private container registry that builds, analyzes and distributes container images. It provides a high level of automation and customization. Red Hat Quay is also available as a hosted service called Quay.io.

REQUEST AN EVALUATION



On all Quay product pages you can find an evaluation form which grants you access to the software for a 90 day trial period.

Alternatively you can signup **for free** on Quay.io

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.



[linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)



[facebook.com/redhatinc](https://www.facebook.com/redhatinc)



twitter.com/RedHat