splunk>

Splunk User Group: Deploying Splunk on OpenShift

Eric Gardner (Splunk) – Sr. Solutions Engineer
Matthew Bach (RedHat) – Sr. Specialist Solutions Architect
Public Sector - DoD

splunk> turn data into doing™

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or plans of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results may differ materially. The forward-looking statements made in the this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, it may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements made herein.

In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only, and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionalities described or to include any such feature or functionality in a future release.

#whoami

Eric Gardner ericg@splunk.com

- 20+ years in IT (ITOPS focus)
- Worked with DoD/INTEL since leaving the Army in 1999.
- Spend my time traveling and usually planning travel when not actually doing it. Lately spending lots of time fighting bamboo.
- Based out of Bridgton, ME (that's about 1 hour north-west of Portland and Stephen King's stomping grounds)

#whoami

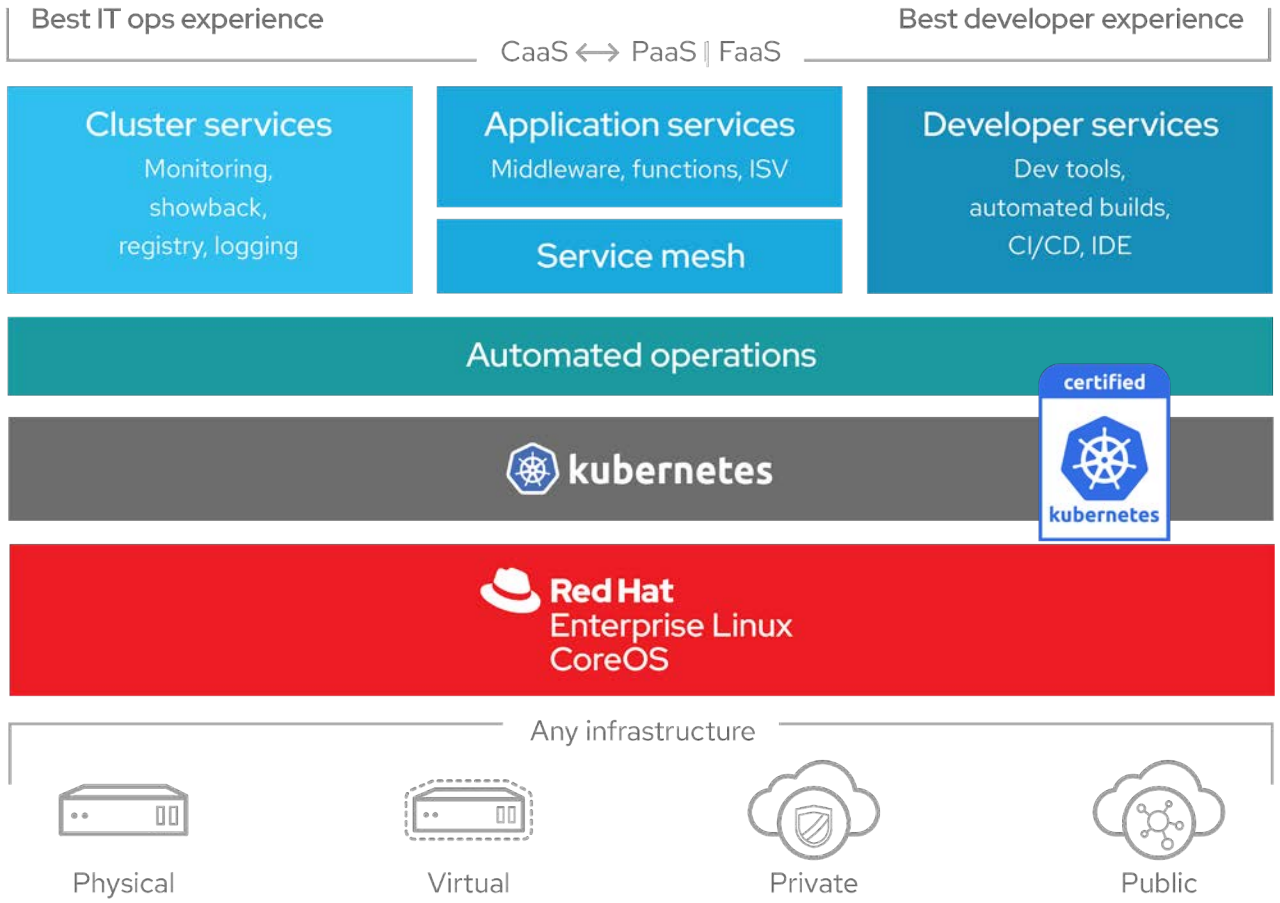
Matthew Bach: mbach@redhat.com

uid=1000(mbach) gid=1000(mbach) groups=1000(mbach),10(wheel)

- Information Systems Technician in the US Navy from 2004 - 2017
- Have been a Red Hatter ever since ~4.5 years
- RH OpenShift & Ansible Specialist
- Cover DoD customers (SOF, Missile Defense, 4th Estate)
- Based in VA Beach, VA
- Spend my time raising kids, tinkering with tech, video games, and mountain biking

Red Hat OpenShift

OpenShift 4 - A smarter Kubernetes platform



Automated, full-stack installation from the container host to application services

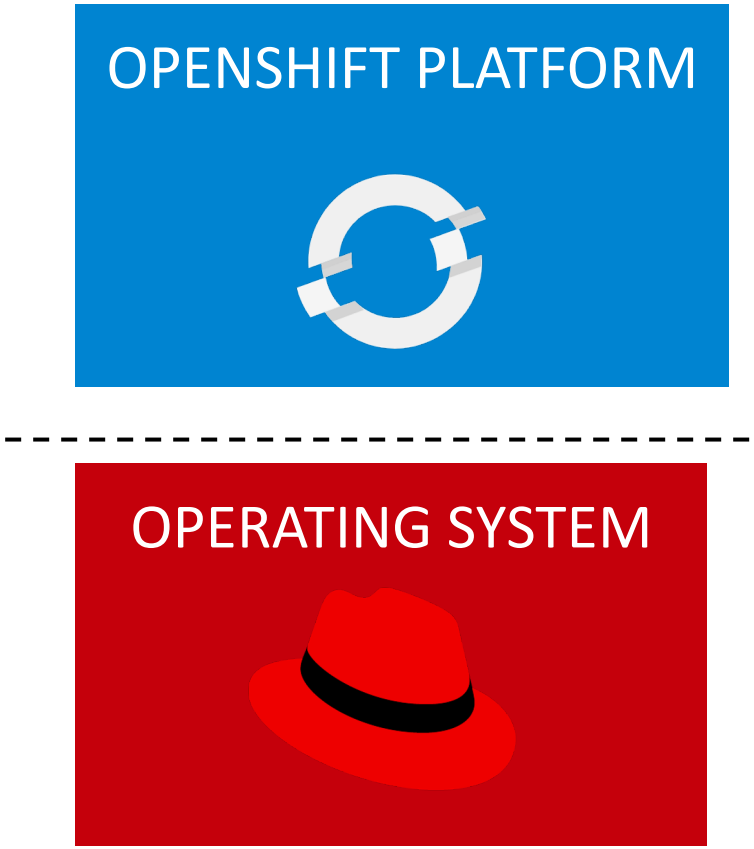
Seamless Kubernetes deployment to any cloud or on-premises environment

Autoscaling of cloud resources

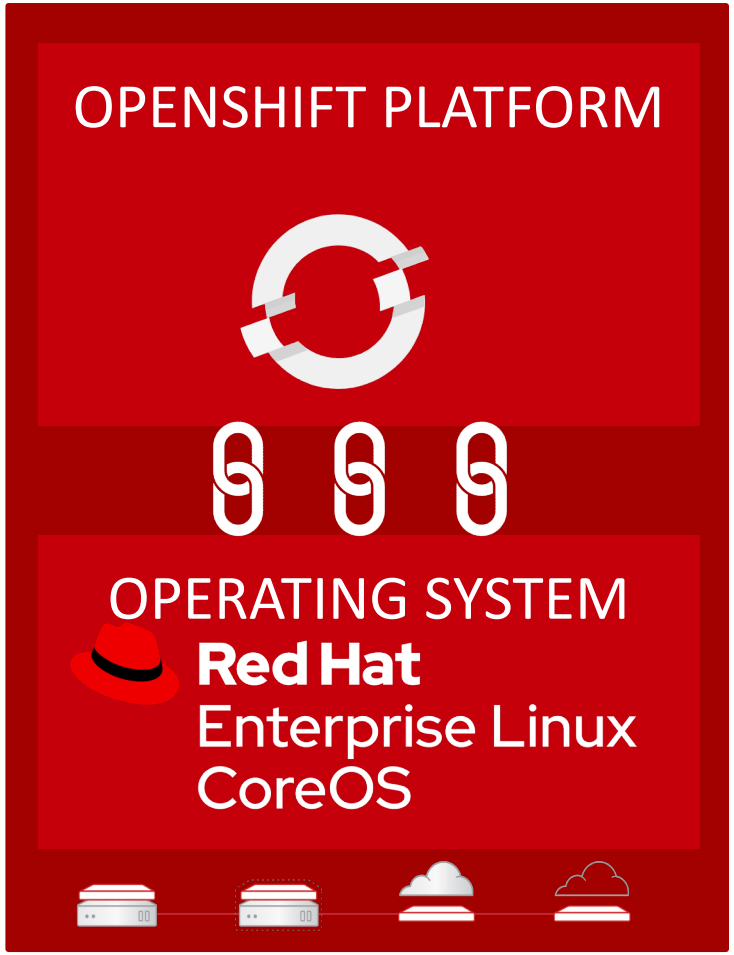
One-click updates for platform, services, and applications

Full-stack automated install

OPENSHIFT 3 & 4



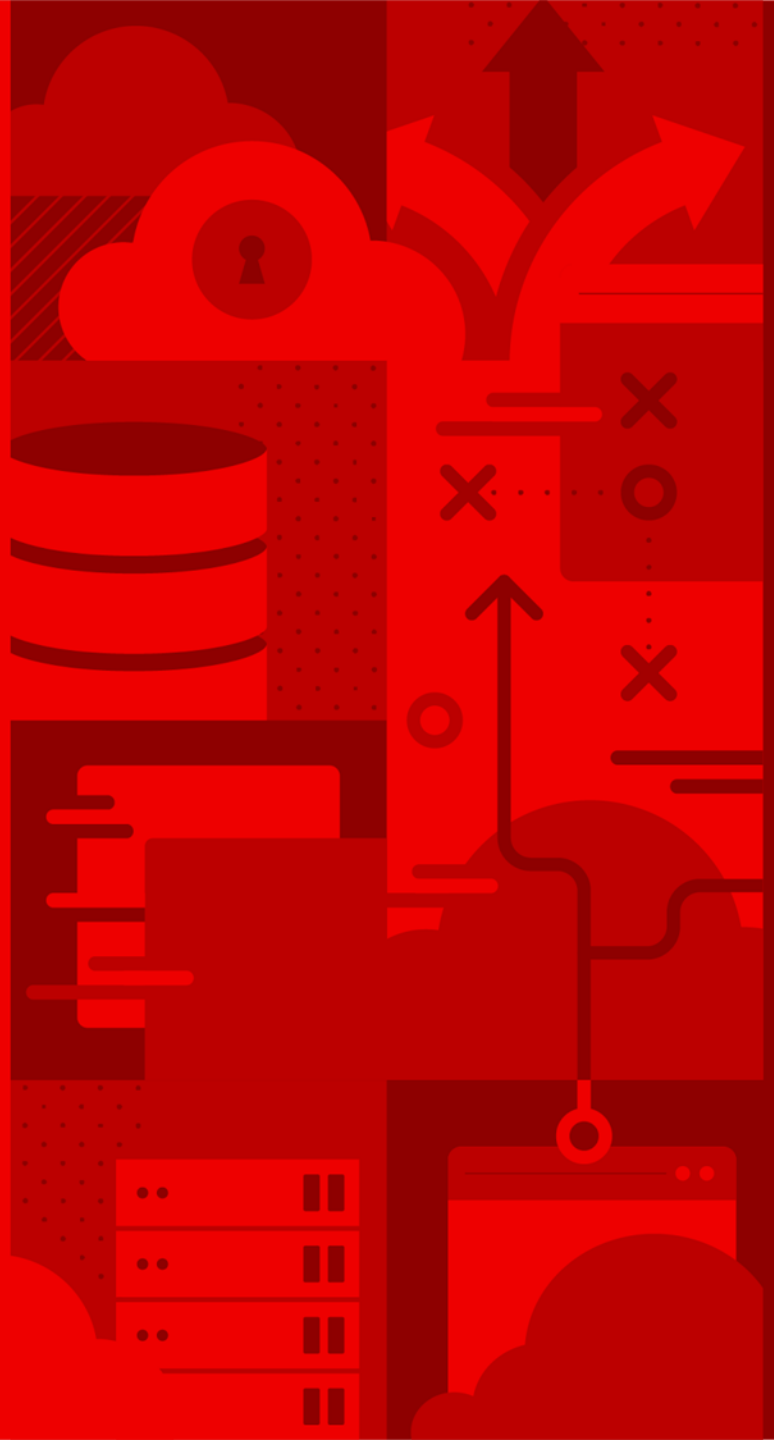
OPENSHIFT 4 (only)



Domain or Application Specific Knowledge

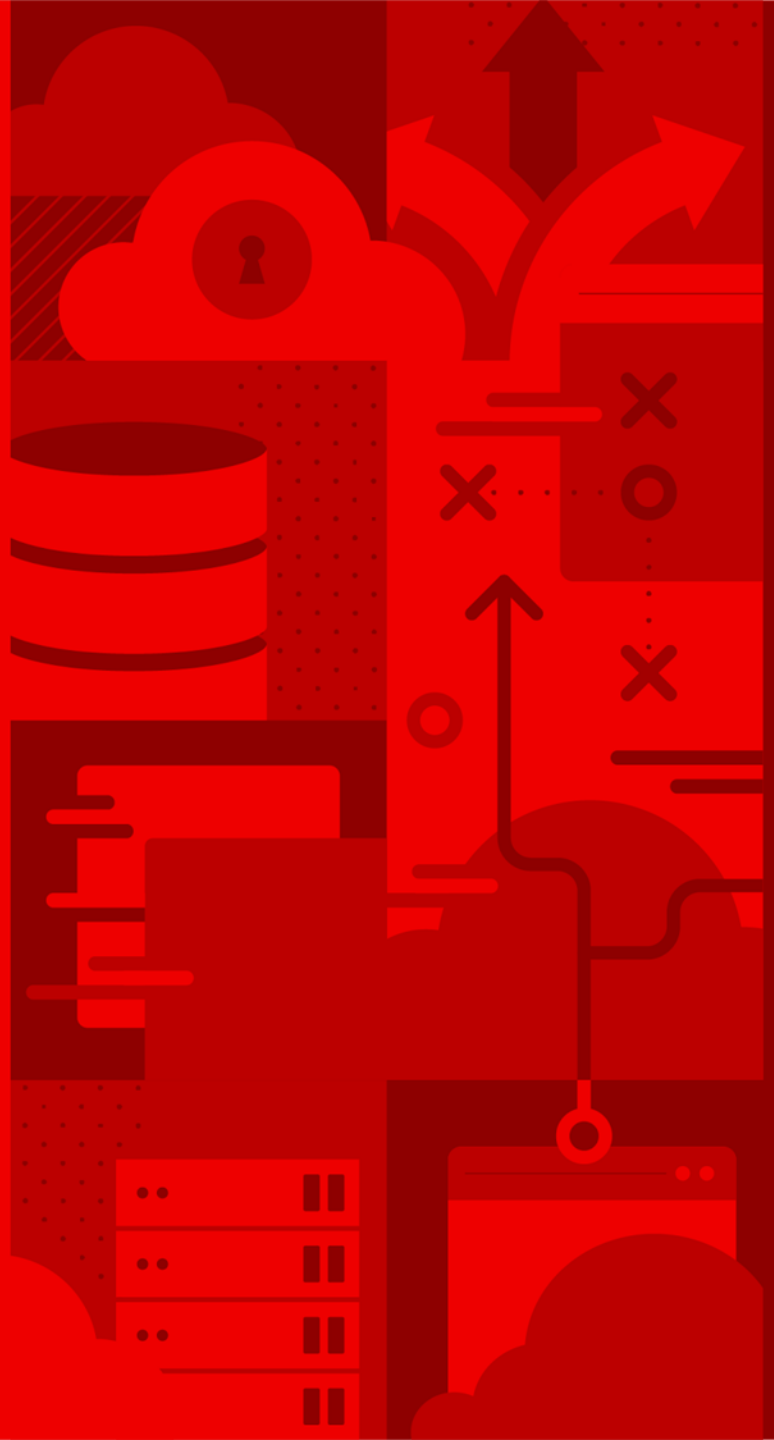
real-world experience with managing your application(s)

- Install
- Self Heal
- Scale
- Update
- Backup
- Clean Up
- Observability
- Resiliency



What's an Operator

An operator represents human operational knowledge in software, to reliably manage an application.

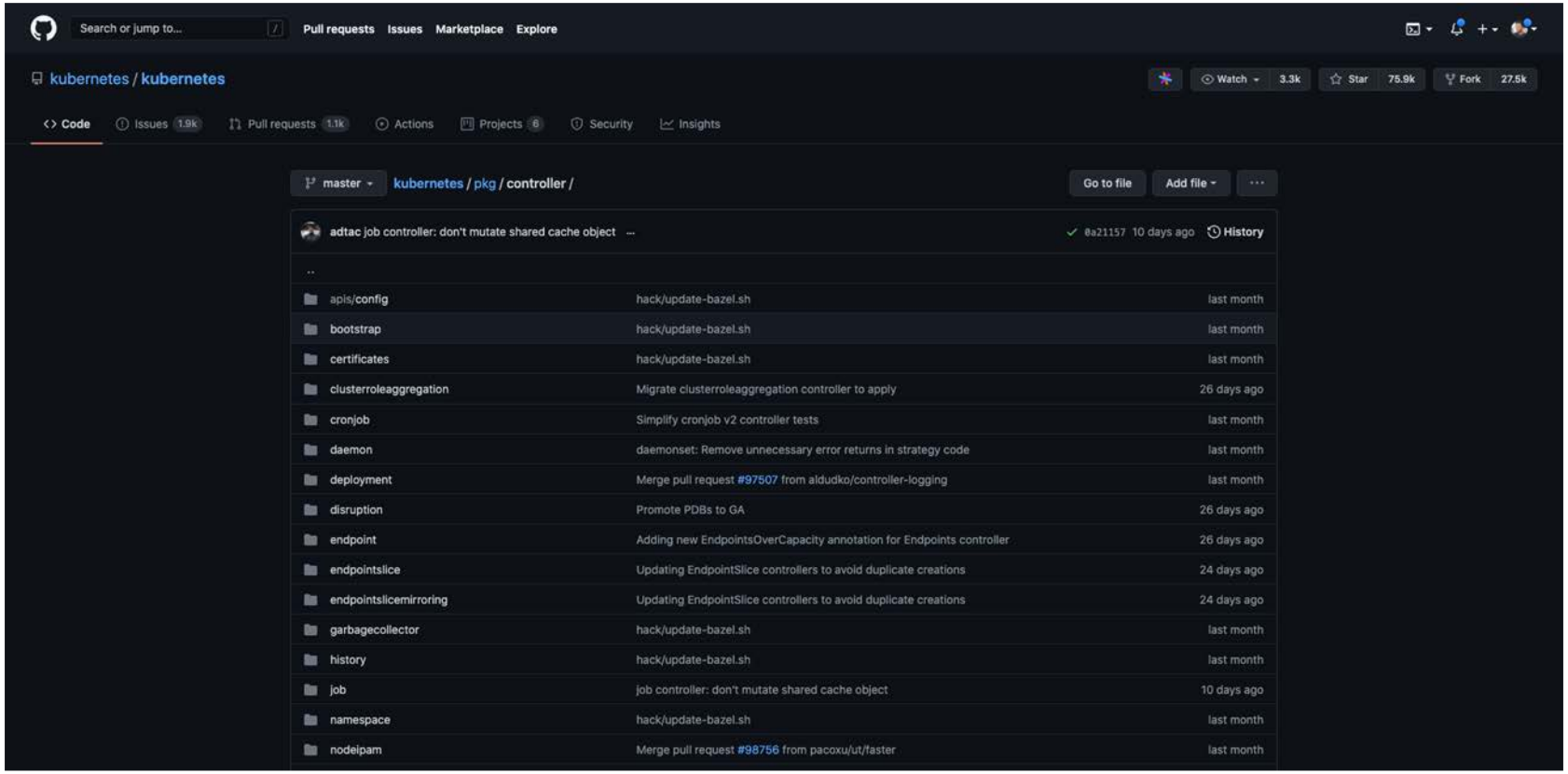


Operators

How you run and manage a particular or *specific* piece of complex software.

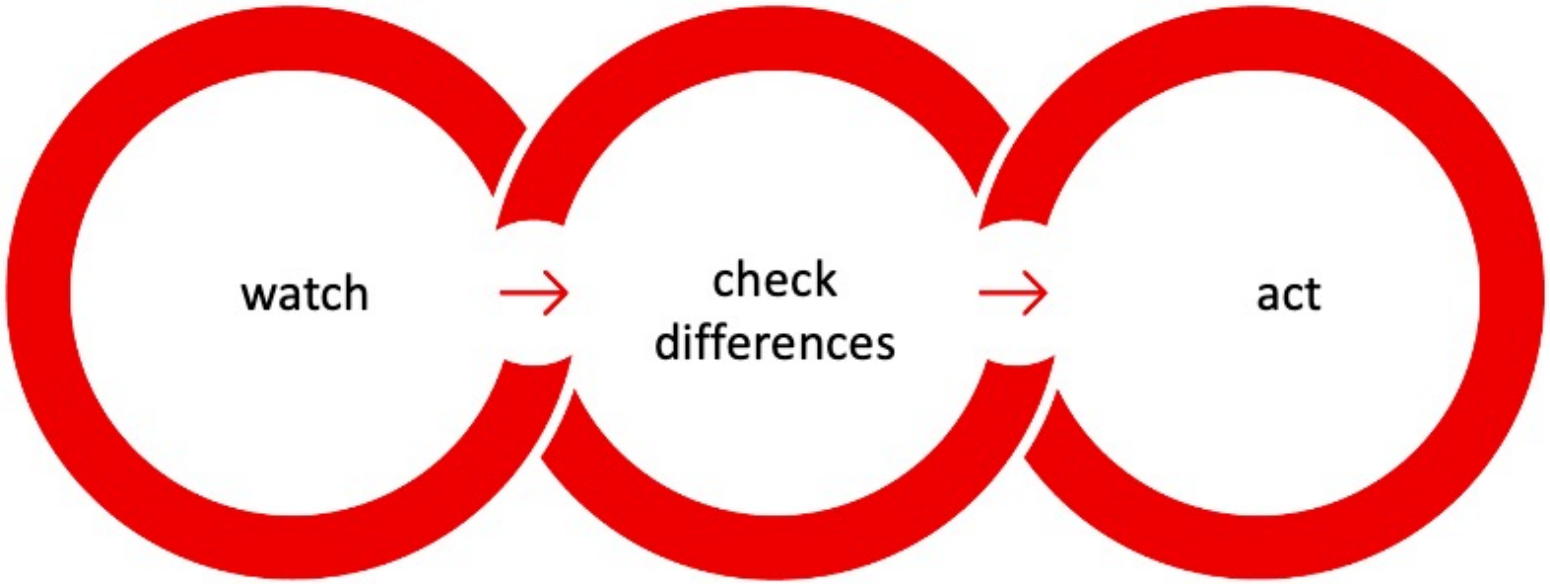
Controllers in Kubernetes

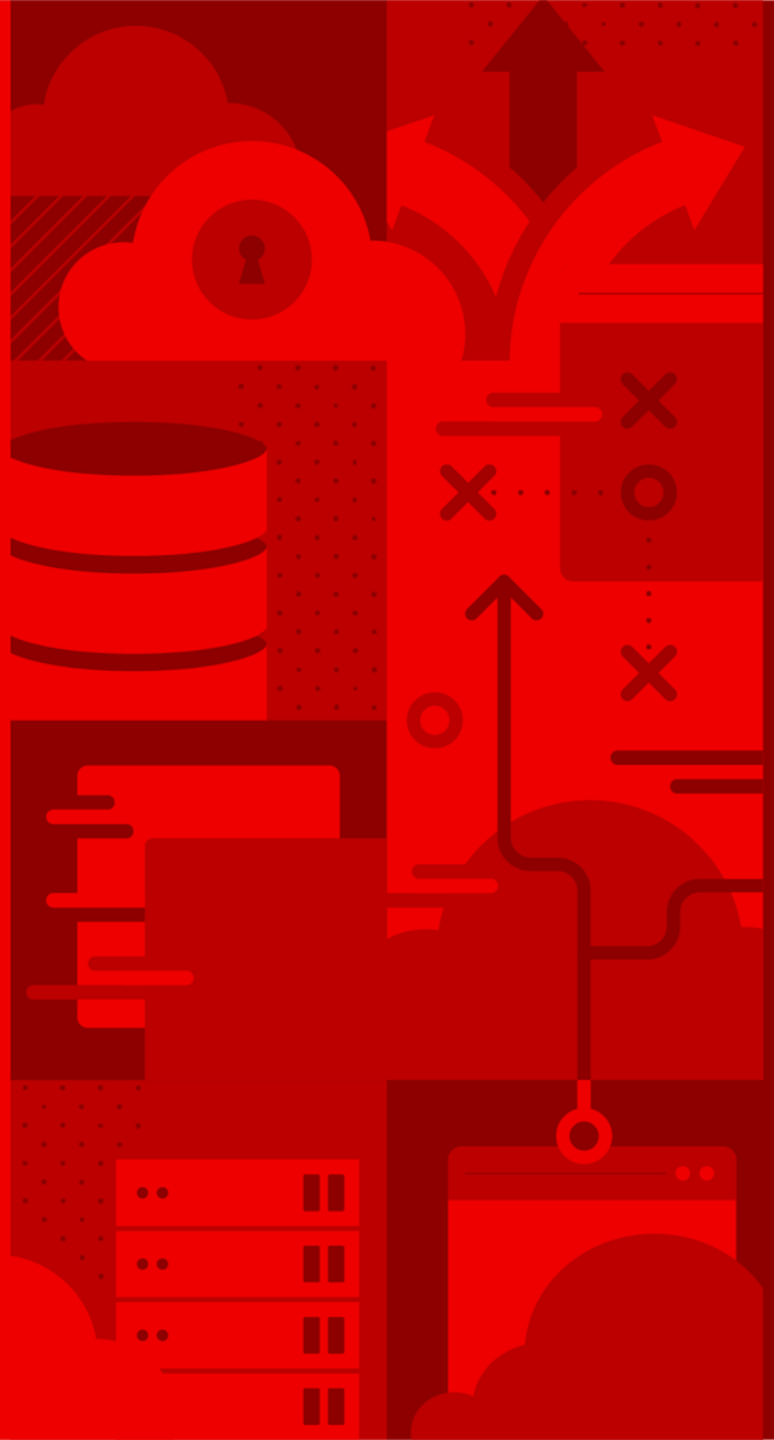
https://github.com/kubernetes/kubernetes/tree/master/pkg/controller



Kubernetes Controller

control loop



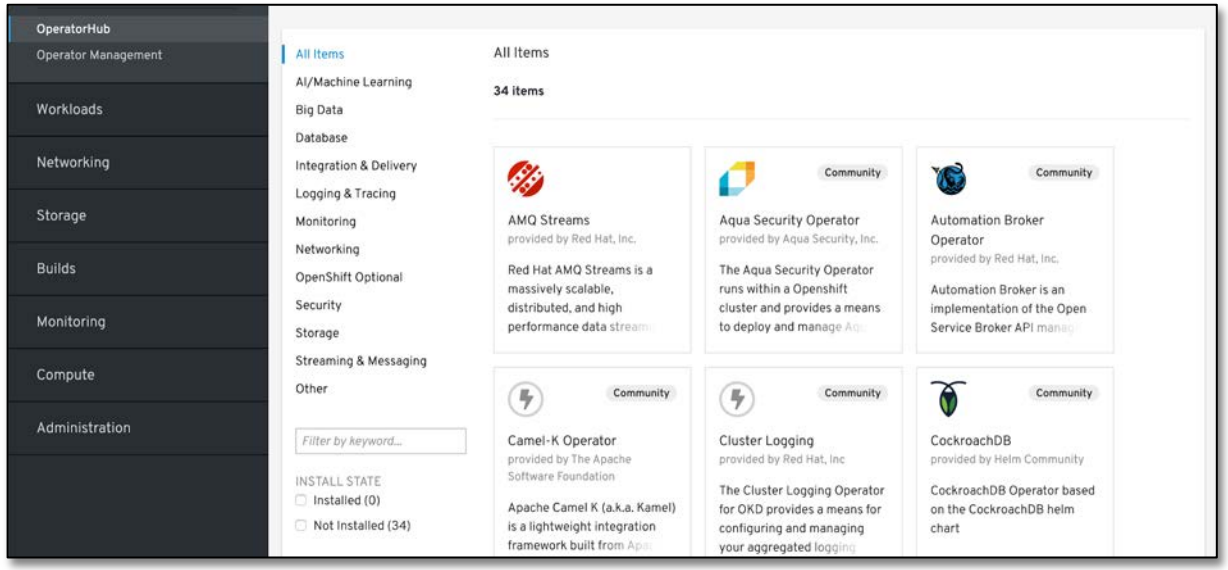


All operators ARE controllers but
NOT ALL controllers are operators

Controller ≠ Operator
Operator = Controller

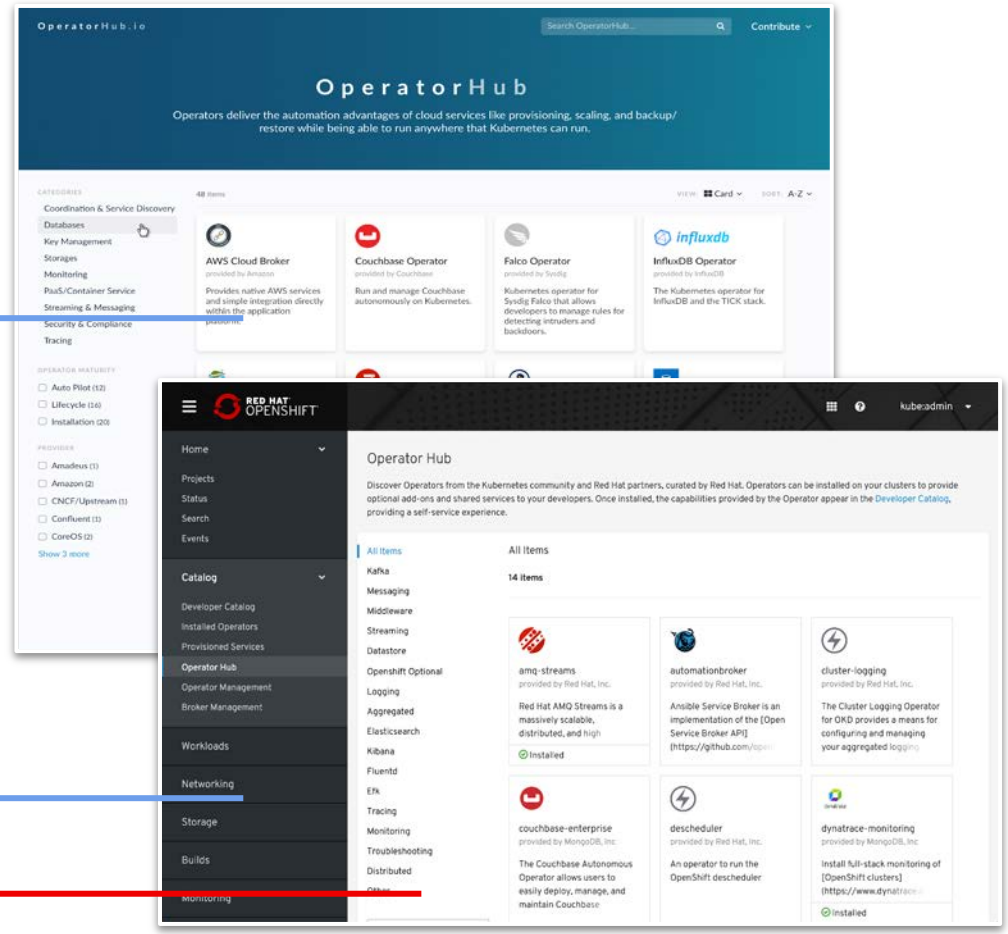
Goals

- Build an ecosystem of software on OpenShift that can be as easy, safe, and reliable to use and operate as a Cloud Service.
- Low-touch, remotely managed, one-click-updates.



OperatorHub and certified Operators

- OperatorHub.io launched by Red Hat, AWS, Microsoft and Google
- OpenShift Operator Certification
- OperatorHub integrated into OpenShift 4



COMMUNITY OPERATORS

OPENSHIFT CERTIFIED OPERATORS

Splunk Operator For Kubernetes Goals

- Eases Splunk Enterprise deployment on private or public cloud
- Simplifies scaling and management of Splunk Enterprise
- Aligns to Kubernetes best practices

Splunk Operator For Kubernetes Resources

- Splunk Operator for Kubernetes is hosted on Github here: https://github.com/splunk/splunk-operator
- Minimum Reference Hardware (minimum specs)
  - StandAlone: Minimum - *Each Standalone Pod: 12 Physical CPU Cores or 24 vCPU at 2Ghz or greater per core, 12GB RAM*
  - Search Head/Search Head Cluster: *Each Search Head Pod: 16 Physical CPU Cores or 32 vCPU at 2Ghz or greater per core, 12GB RAM.*
  - Indexer Cluster: *Each Indexer Pod: 12 Physical CPU cores, or 24 vCPU at 2GHz or greater per core, 12GB RAM.*
- Important points
  - *Splunk Operator does not support vCPU licensing (ingest only)*
  - *Splunk Enterprise is supported by Splunk,Inc when deployed using the Operator (as of v8.2)*
  - *SmartStore (S2) storage architecture is required for support in an operational environment*

Read the Docs

- Getting started docs are found here:
  https://splunk.github.io/splunk-operator/

- In this demo the latest Splunk Operator for Kubernetes has already been installed in the “splunk” project using the OpenShift Client (oc):
  - oc new-project splunk (if it doesn’t already exist)
  - oc project splunk (to switch into the ”splunk” project)
  - oc adm policy add-scc-to-user nonroot -z default (allows nonroot user to run the Splunk service)
  - oc apply -f https://github.com/splunk/splunk-operator/releases/download/1.0.1/splunk-operator-install.yaml (installs the operator in the “splunk” project)

Note: the demo is a basic walkthrough of deploying a distributed Splunk Enterprise architecture in OpenShift. A lot of defaults will be accepted due to limited time. Please read the docs and use advanced features like core count, memory allocation, SmartStore, App Framework, etc. to get the best experience and supported configuration.

Demo

splunk> turn data into doing™