



Advanced Cluster Management

John Gammon
Senior Account Solution
Architect

Aly Ibrahim
Cloud App Dev Solutions
Architect

Lunch & Learn Agenda

- **Advanced Cluster Management Presentation** (~ 25 minutes)
 - Why Advanced Cluster Management?
 - Features of Advanced Cluster Management?
- **Demo** ----- (~30 minutes)
 - Cluster Lifecycle Management
 - Policy and Governance
 - Application Lifecycle Deployments

John Gammon
Senior Account Solution Architect

Aly Ibrahim
Cloud App Dev Solutions Architect

Why Advanced Cluster Management?

Educated Prediction of Future Conditions

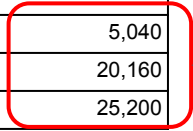
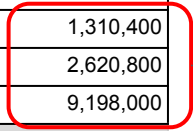
If we accept that enterprise kubernetes is going to grow,, what can we predict?

Let's do the Math

For Your Average Large Company

Expected Pace of Change

Years	2	3	4	5
For Every 1000 Applications	1,000	1,000	1,000	1,000
% Containerized	20.00%	30.00%	40.00%	60.00%
Containerized Apps	200	300	400	600
Number of Kubernetes Clusters (Dev/Test/Prod)	10	20	25	30
Sub-Total Containerized Apps	2,000	6,000	10,000	18,000
Concurrency Factor	1.40	1.40	1.40	1.40
Total Containerized Apps	2,800	8,400	14,000	25,200
Annual Frequency of Change				
Slow (1 per week)	145,600	436,800	728,000	1,310,400
Medium (2 per week)	291,200	873,600	1,456,000	2,620,800
Fast (daily)	1,022,000	3,066,000	5,110,000	9,198,000
Volume of Daily Pipelines				
Slow (1 per week)	560	1,680	2,800	5,040
Medium (2 per week)	2,240	6,720	11,200	20,160
Fast (daily)	2,800	8,400	14,000	25,200

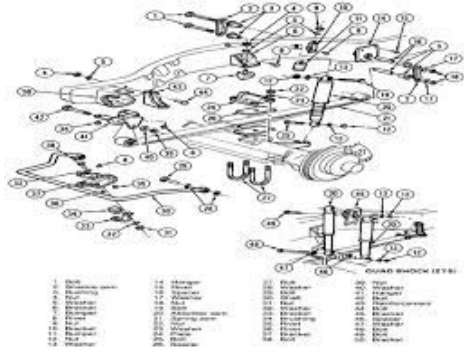
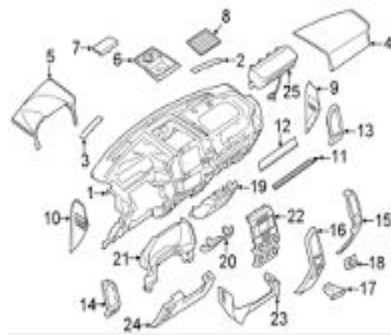
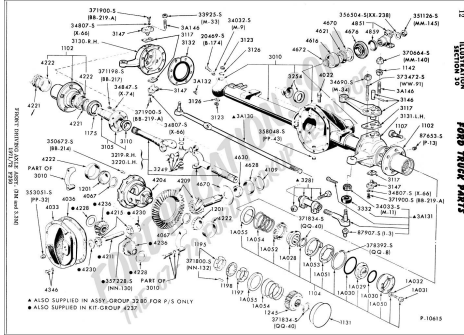


Enterprise Kubernetes

CONFIDENTIAL designer

xKS/DIY?

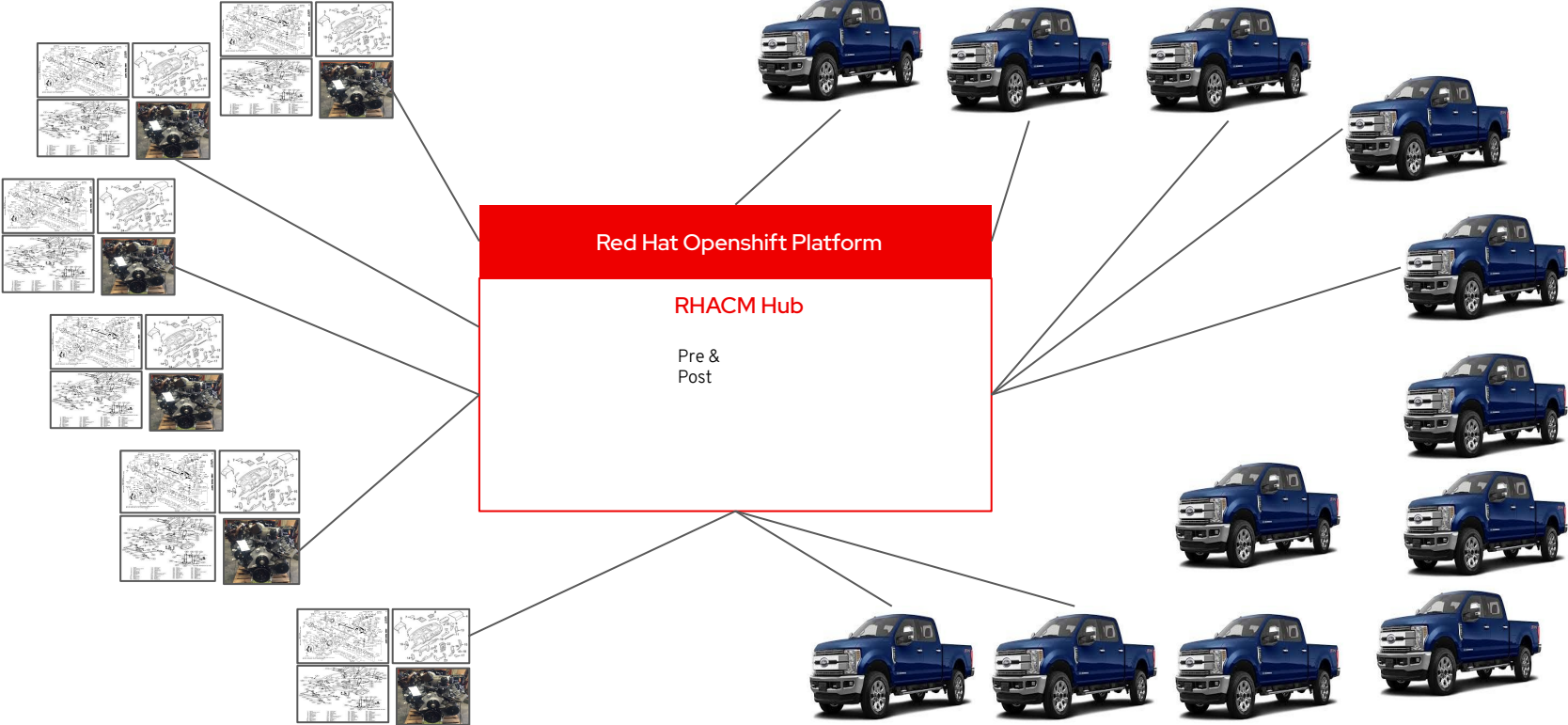
OpenShift?



OR



Advanced Cluster Management to the Rescue



Benefits

Red Hat OpenShift and Red Hat Advanced Cluster Management for Kubernetes



Accelerate development to production

Self-service provisioning allows app dev teams to request clusters directly from a catalog removing central IT as a bottleneck.



Increase application availability

Placement rules can allow quick deployment of clusters across distributed locations for availability, capacity, and security reasons.



Reduce costs

Centralized management of clusters reduces operational cost, makes the environment consistent, and removes the need to manually manage individual clusters.



Ease compliance

Policies can be written by the security team and enforced at each cluster, allowing environments to conform to your policy.

Advanced Cluster Management

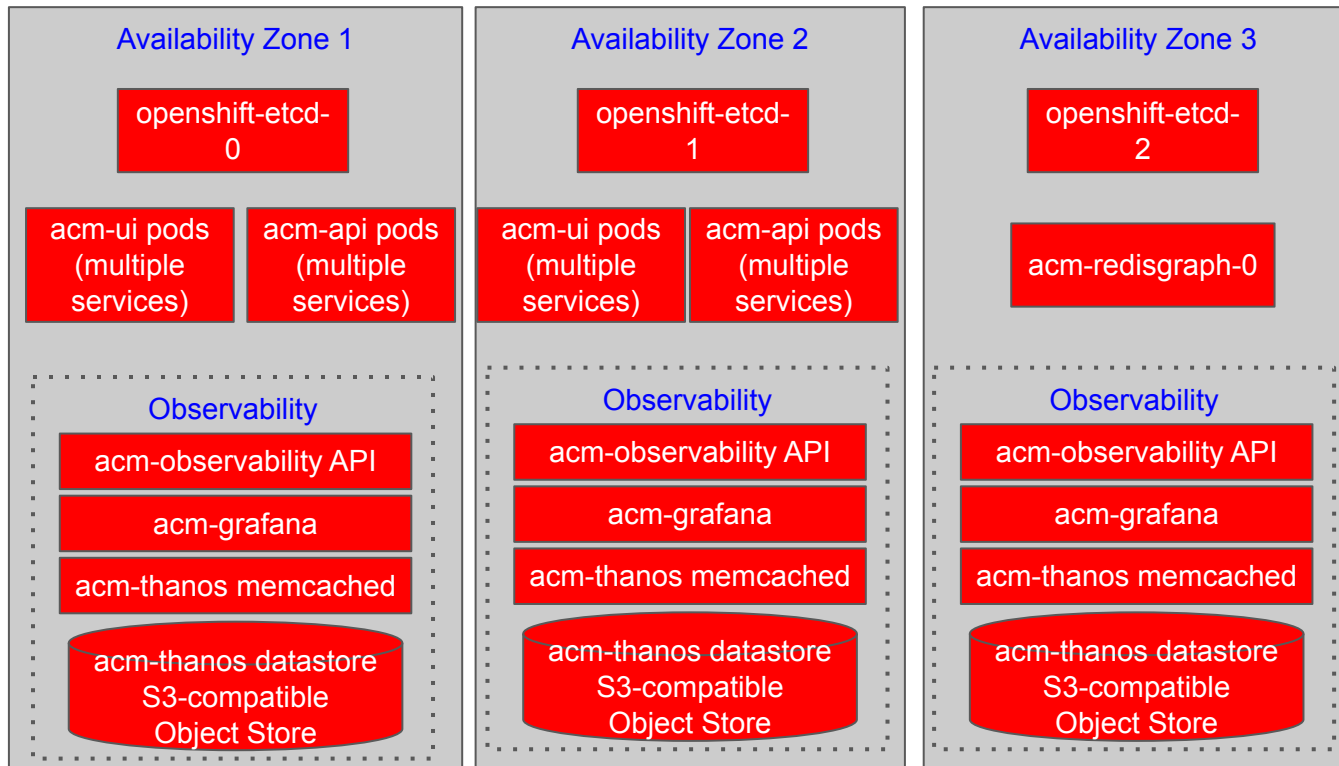
Advanced Cluster Management for Kubernetes

Hub Sizing Requirements

2.0

+ About 20Gi of persistent storage

OpenShift Node Role	Availability Zones	Data Stores	Total reserved memory (lower bound)	Total reserved CPU (lower bound)
Master	3	etcd x 3	<i>Per OpenShift sizing guidelines</i>	<i>Per OpenShift sizing guidelines</i>
Worker	3	redisgraph/redis x 1	12Gi	6 CPU



High Availability

- Fault domains spread pods across AZs via podAntiAffinity
- Stateful datastores require 3 replicas
- All Stateless UI & API services will be run with at least 2 replicas to support rolling updates and fault domain outages
- Thanos requires an S3 object store that can be run inside or outside the cluster
- Redis/RedisGraph provides an in-memory index for search; search data re-indexed in case of Pod or Node failure

Unified Multi-Cluster Management

Single Pane for all your Kubernetes Clusters

The screenshot displays the Red Hat ACM for Kubernetes interface. The top section shows an overview of clusters categorized by provider: Azure (1 cluster: 01 AKS), Amazon (1 cluster: 01 RHOC), auto-detect (2 clusters: 01 Other), and MyDataCenter (1 cluster: 01 RHOC). Below this, a summary bar indicates 4 Apps, 5 Clusters, 3 Kubernetes types, 1 Region, 17 Nodes, and 646 Pods. A cluster compliance gauge shows 100% compliance. The bottom section is a 'Clusters' table with the following data:

Name	Namespace	Labels	Endpoint	Status	Nodes	Kubernetes Version	Storage	Memory	CPU	
exec2-iks	mcm-exec2-iks	cloud=IBM datacenter=dal13 environment=dev name=exec2-iks region=US vendor=IKS	-	Offline	1	3.1.2-dev	-	33%	70%	
social-dev-1	mcm-social-dev-1	cloud=IBM datacenter=oregon environment=dev name=social-dev-1 owner=marketing region=us-west vendor=ICP	launch	Ready	1	3.1.2	v1.11.5+icp-ee	100%	62%	45%
social-dev-2	mcm-social-dev-2	cloud=IBM datacenter=oregon environment=dev name=social-dev-2 owner=marketing region=us-west vendor=ICP	launch	Offline	1	3.1.2	v1.11.1+icp-ee	100%	48%	47%
social-dev-gke	social-dev-gke	cloud=Google datacenter=us-central1-a environment=dev name=social-dev-gke owner=marketing region=US vendor=GKE	-	Ready	1	3.1.2-dev	v1.11.7-gke.12	-	6%	22%
social-prod-1	mcm-social-prod-1	cloud=IBM datacenter=oregon environment=Prod name=social-prod-1 owner=marketing region=us-west vendor=ICP	launch	Ready	1	3.1.2	v1.11.1+icp-ee	100%	52%	34%
social-prod-eks	social-prod-eks	cloud=AWS datacenter=us-east-1 environment=Prod name=social-prod-eks owner=marketing	-	Ready	1	3.1.2-dev	v1.11.8-eks-7c34c0	-	1%	10%

- **Centrally** create, update and delete Kubernetes clusters **across multiple** private and public clouds
- Search, find and modify **any** kubernetes resource across the **entire** domain.
- **Quickly** troubleshoot and resolve issues across your **federated** domain

Multi-Cluster Lifecycle Management

Creating & Importing Clusters

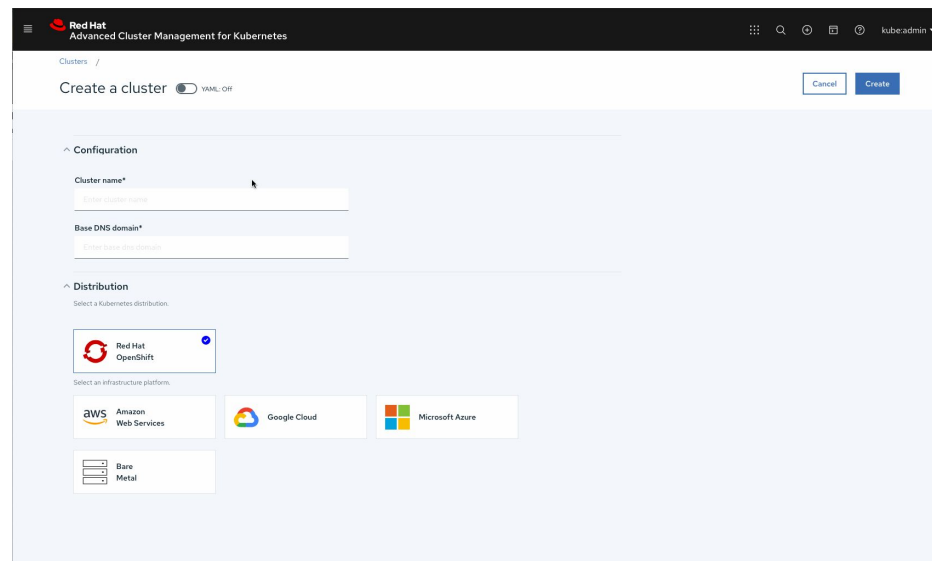
- **Create, Upgrade** and **Destroy** OCP clusters running on **Bare-metal** as well as public cloud
- Leverage [Hive API for OCP cluster deployment](#)
- Wizard or YAML based create cluster flow
- Launch to an OCP Console from ACM
- Access cluster login credentials and download kubeadmin configuration



IT Operations

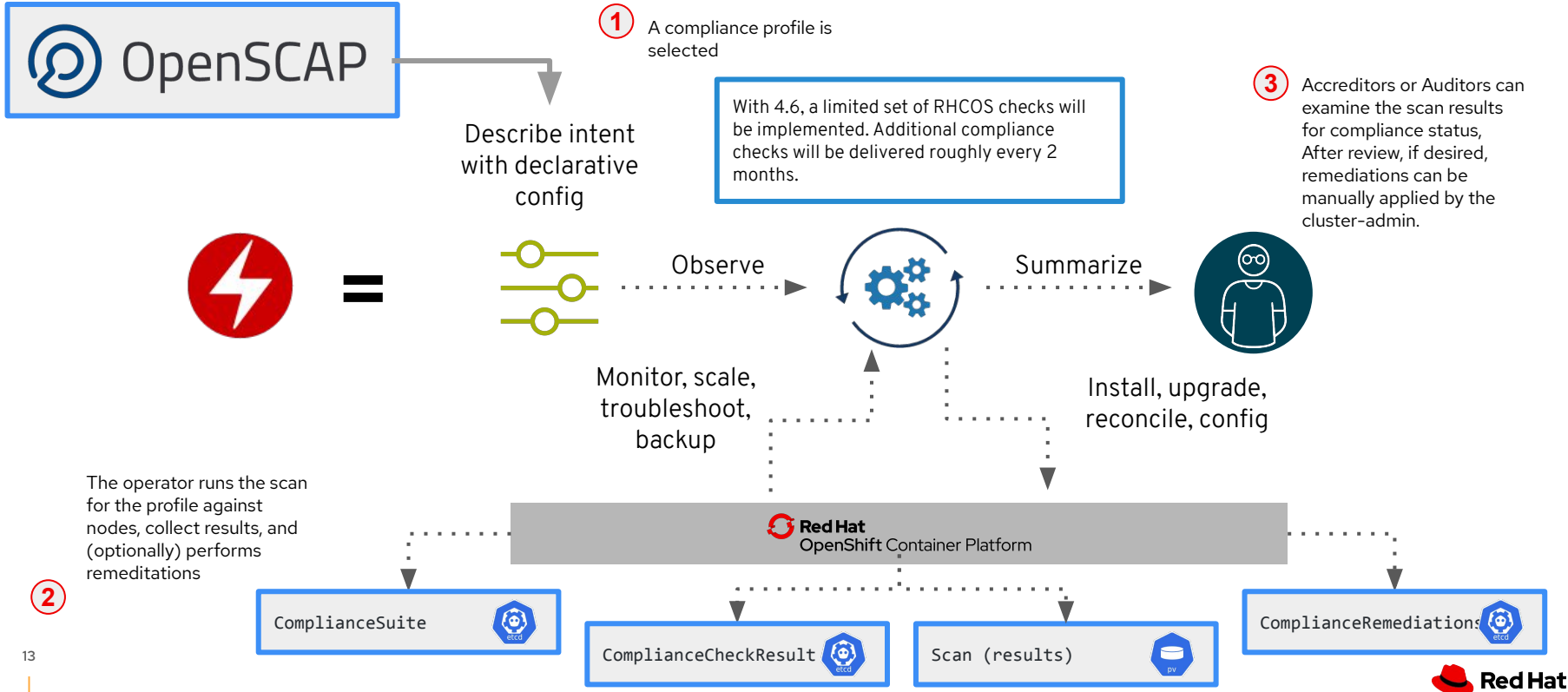


DevOps/SRE



For Each OpenShift Cluster

OpenShift Compliance Operator: Declarative Security Compliance (As of 10/22)



Policy based Governance, Risk and Compliance

Don't wait for your security team to tap you on the shoulder

The dashboard displays a summary of policy violations and security findings. The top navigation bar shows counts for Policy Violations (3), Cluster Violations (1), High Severity Findings (1), and Medium Severity Findings (1). The main content area is divided into 'Top violations' and 'Top security findings'. The 'Top violations' section lists three items: 'policy-cis', 'policy-grc', and 'policy-role'. The 'Top security findings' section shows one 'Policy violation finding' and a message stating 'No other security findings'. Below this, there is a 'Most impacted controls' section with a key and a diagram. A detailed view of a 'compliancePolicy' is shown in the foreground, featuring a table of details, a code editor with a YAML snippet, and a table of object templates.

Type	Detail
Name	policy-prod
Message	-
Status	-
Enforcement	-
Exclude Namespaces	kube*
Include Namespaces	default

```
31 - from:
32   - podSelector: {}
33   podSelector:
34     matchLabels: null
35   - complianceType: musthave
36   objectDefinition:
37     apiVersion: v1
38     kind: LimitRange
39     metadata:
40       name: mem-limit-range
41     spec:
42       limits:
43         - default:
44             memory: 512Mi
45           defaultRequest:
46             memory: 256Mi
47           type: Container
48       remediationAction: enforce
49
```

Name	Compliance Type	API version	Kind	Last Transition	Compliant
restricted-mcm	musthave	policy/v1beta1	PodSecurityPolicy	-	-
deny-from-other-namespaces	musthave	networking.k8s.io/v1	NetworkPolicy	-	-
mem-limit-range	musthave	v1	LimitRange	-	-

- **Centrally** set & enforce policies for security, applications, & infrastructure
- Quickly **visualize** detailed **auditing** on configuration of apps and clusters
- Built-in **CIS** compliance policies and audit checks
- **Immediate** visibility into your compliance posture based on **your** defined standards

Policy based Governance, Risk and Compliance

Don't wait for your security team to tap you on the shoulder



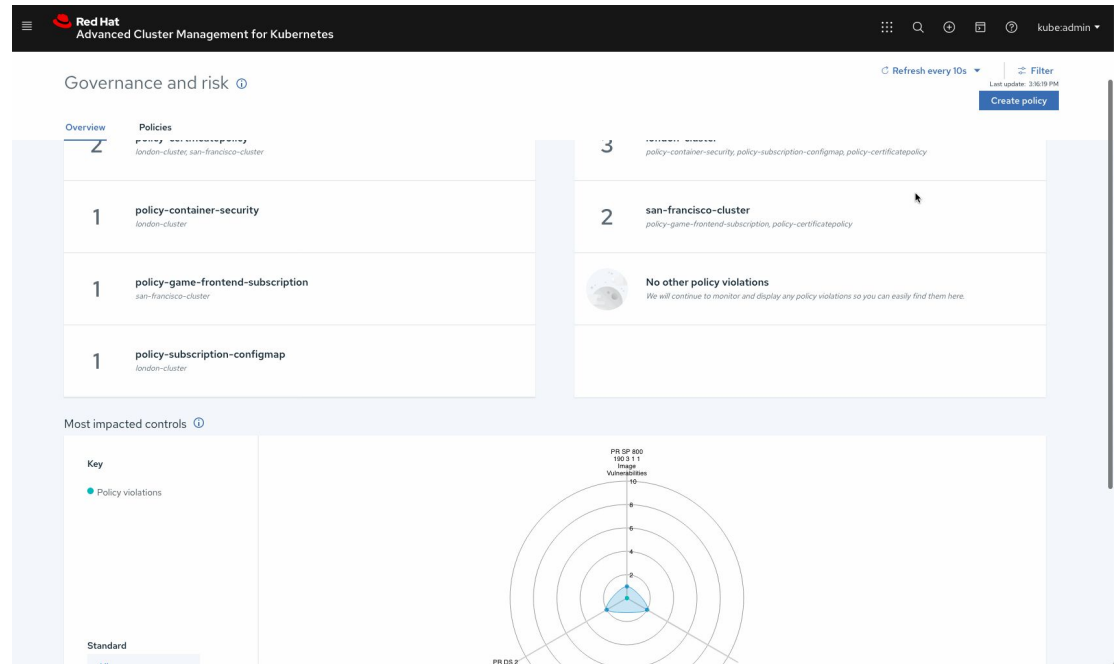
Security Ops



IT Operations



- Standard Policies out of the box
 - FISMA
 - HIPAA
 - NIST
 - PCI
- Leverage Different Categories to Represent more standards (if Needed)
- Use Labels to enforce policies against clusters
- Use **inform** to view policy violations
- Use **enforce** to view violations and automatically remediate



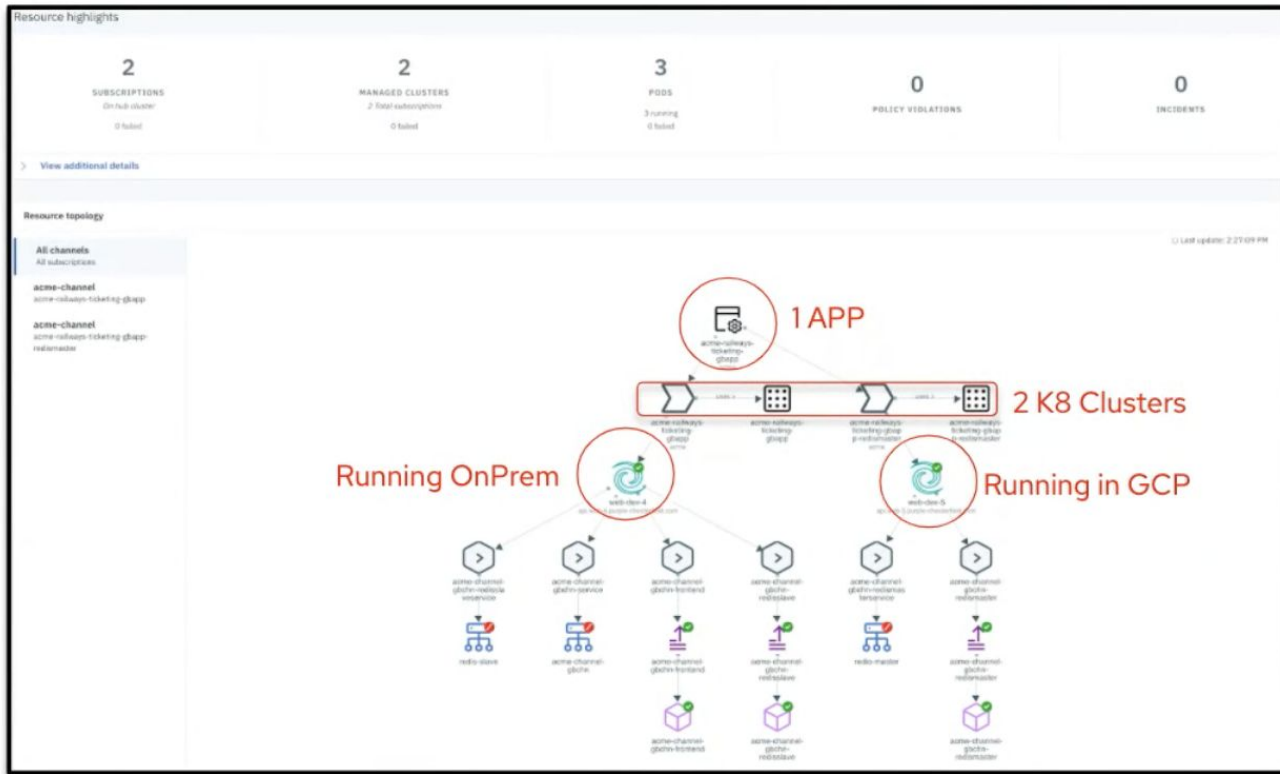
The screenshot shows the Red Hat Advanced Cluster Management for Kubernetes (ACM) interface. The top navigation bar includes the Red Hat logo, the product name, and a user profile dropdown for 'kubeadmin'. The main content area is titled 'Governance and risk' and features a 'Refresh every 10s' button and a 'Filter' dropdown. A 'Create policy' button is visible in the top right. The interface is divided into several sections:

- Overview:** A table listing policies across different clusters. The table has two columns: 'Policies' and 'Clusters'. The data is as follows:

Policies	Clusters
policy-container-security	london-cluster, san-francisco-cluster
policy-game-frontend-subscription	san-francisco-cluster
policy-subscription-configmap	london-cluster
- Policy List:** A list of policies with their counts and names:
 - 3 policy-container-security, policy-subscription-configmap, policy-certificatpolicy
 - 2 san-francisco-cluster, policy-game-frontend-subscription, policy-certificatpolicy
- No other policy violations:** A message stating 'We will continue to monitor and display any policy violations so you can easily find them here.'
- Most impacted controls:** A section with a key for 'Policy violations' and a radar chart. The chart shows a score of 10 for 'PR.DS.2' and 'PR.GP.800'.

Advanced Application Lifecycle Management

Simplify your Application Lifecycle

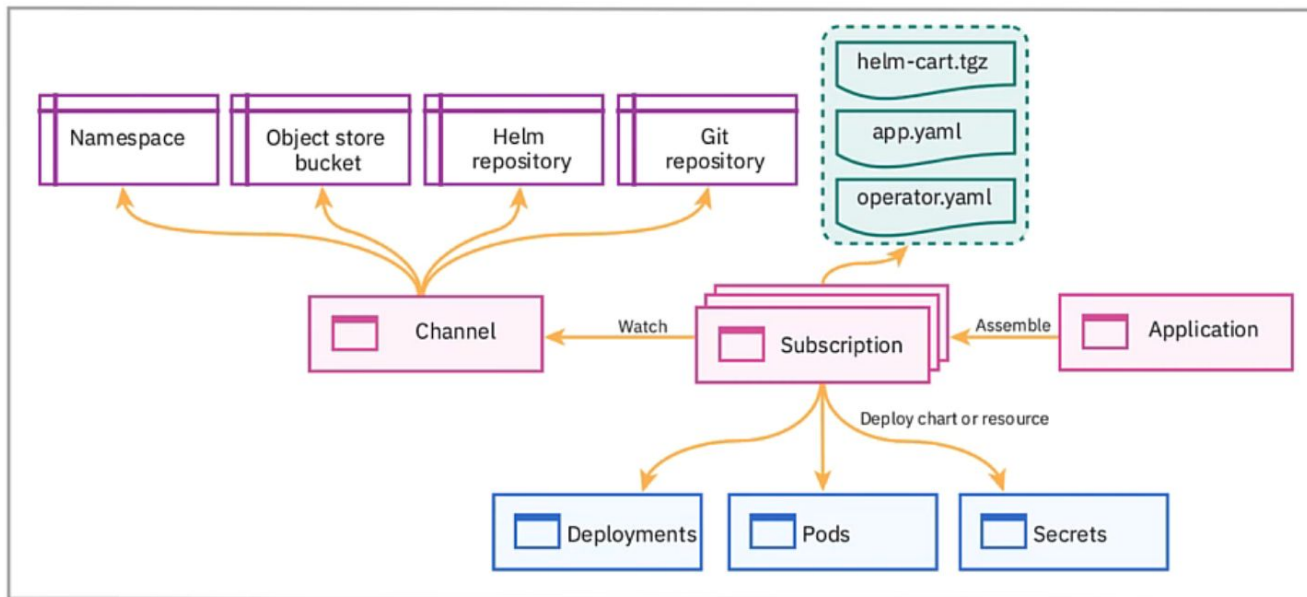


Easily Deploy Applications at **Scale**

Deploy Applications from **Multiple** Sources

Quickly **visualize** application relationships **across** clusters and those that **span** clusters

Application LifeCycle Management



Integration Architecture Overview for Application Life Cycle

