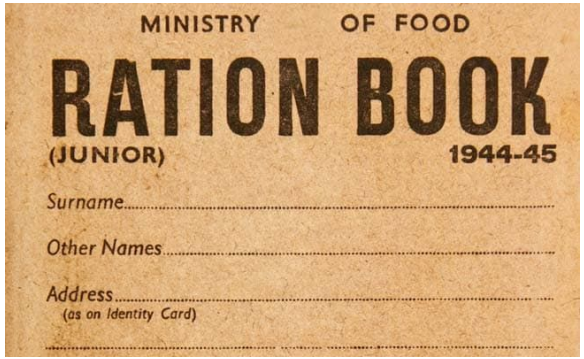
Sizing your Applications on OpenShift

Veer Muchandi
Chief Architect - Container Solutions, NACS
May 8, 2019

So you got an OpenShift Cluster running and you want to use it across multiple teams



Next steps ..
Preventing over-usage
Resource allocations

Constrain Resource Consumption using Quotas

- CPU requests and limits
- Memory requests and limits
- Ephemeral Storage requests and limits(tech preview)
- Storage requests (or for storage class)
- # of Persistent Volume Claims (or for a storage class)
- # of Pods
- # of Replication Controllers
- # of Services
- # of Secrets
- # of ConfigMaps
- # of openshift ImageStreams

Multi-Project Quotas

Cluster Resource Quota
- Based on Labels or Annotations or both
- Applied across all projects matching the labels

Use cases:
- Applying quotas on user
- Applying quotas on a team

Forcing Quotas

You can mandate quota definitions on certain resources.

Example: If a user creates a PVC for the gold storage class, it won't be accepted if quota is not defined for this in the project

```
admissionConfig:
  pluginConfig:
    ResourceQuota:
      configuration:
        apiVersion: resourcequota.admission.k8s.io/v1alpha1
        kind: Configuration
        limitedResources:
        - resource: persistentvolumeclaims
        matchContains:
        - gold.storageclass.storage.k8s.io/requests.storage
```

Limit the max

Enforce specifying min

Limit Ranges

- Compute resource constraints at
  - Pod, Container (CPU and Memory Min, Max, MaxLimitRequestRatio, Default)
  - Image (max size)
  - ImageStream (max image tags, max images)
  - PVC level (min, max)

- All resource create/modification requests are evaluated against LimitRange Object

Quality of Service

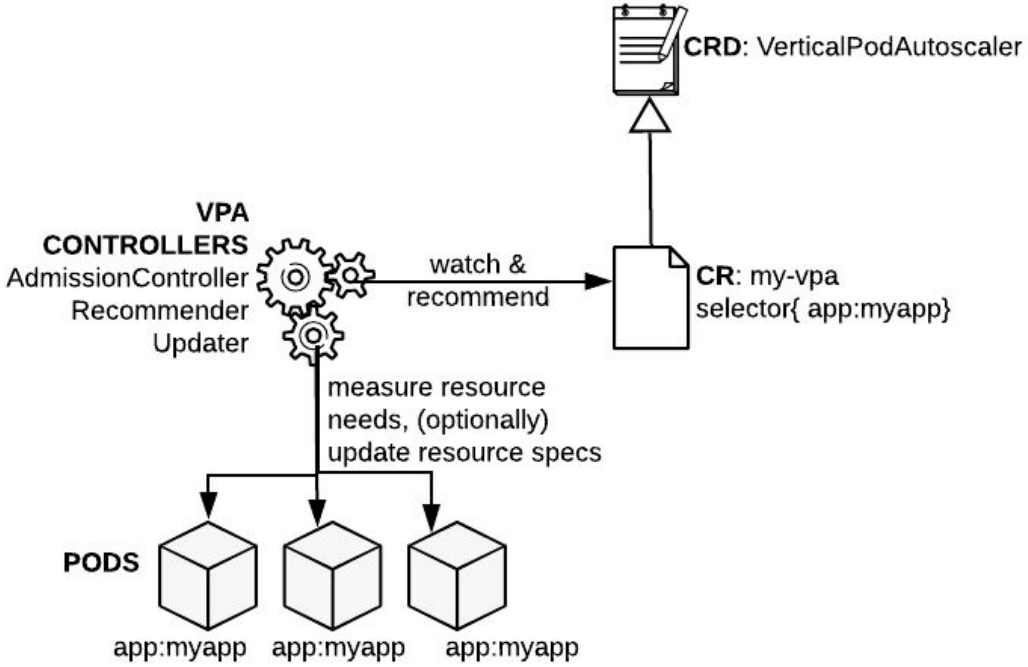
Request : at the time creation

Limit: max size

Request, Limit Unspecified	Best Effort
Request < Limit	Burstable
Request = Limit	Guaranteed

How to determine right size for Pods?

Welcome to Vertical Pod Autoscaler (beta)



Modes:

Off: recommend resources in status of VPA CR
Initial: Adjust container resource spec for new pods
Auto: Adjust container resource spec for running pods

Vertical Pod Autoscaler

```
apiVersion: autoscaling.k8s.io/v1beta1
kind: VerticalPodAutoscaler
metadata:
 name: my-vpa
spec:
 selector:
   matchLabels:
      app: my-app
```

Selects pods based on the matchLabels

Vertical Pod Autoscaler

```
status:
  conditions:
  - lastTransitionTime:
2019-04-22T20:15:11Z
    status: "True"
    type: RecommendationProvided
  recommendation:
    containerRecommendations:
    - containerName: mycontainer
      lowerBound:
        cpu: 25m
        memory: 262144k
      target:
        cpu: 25m
        memory: 262144k
      uncappedTarget:
        cpu: 25m
        memory: 262144k
      upperBound:
        cpu: 3179m
         memory: "6813174422"
```

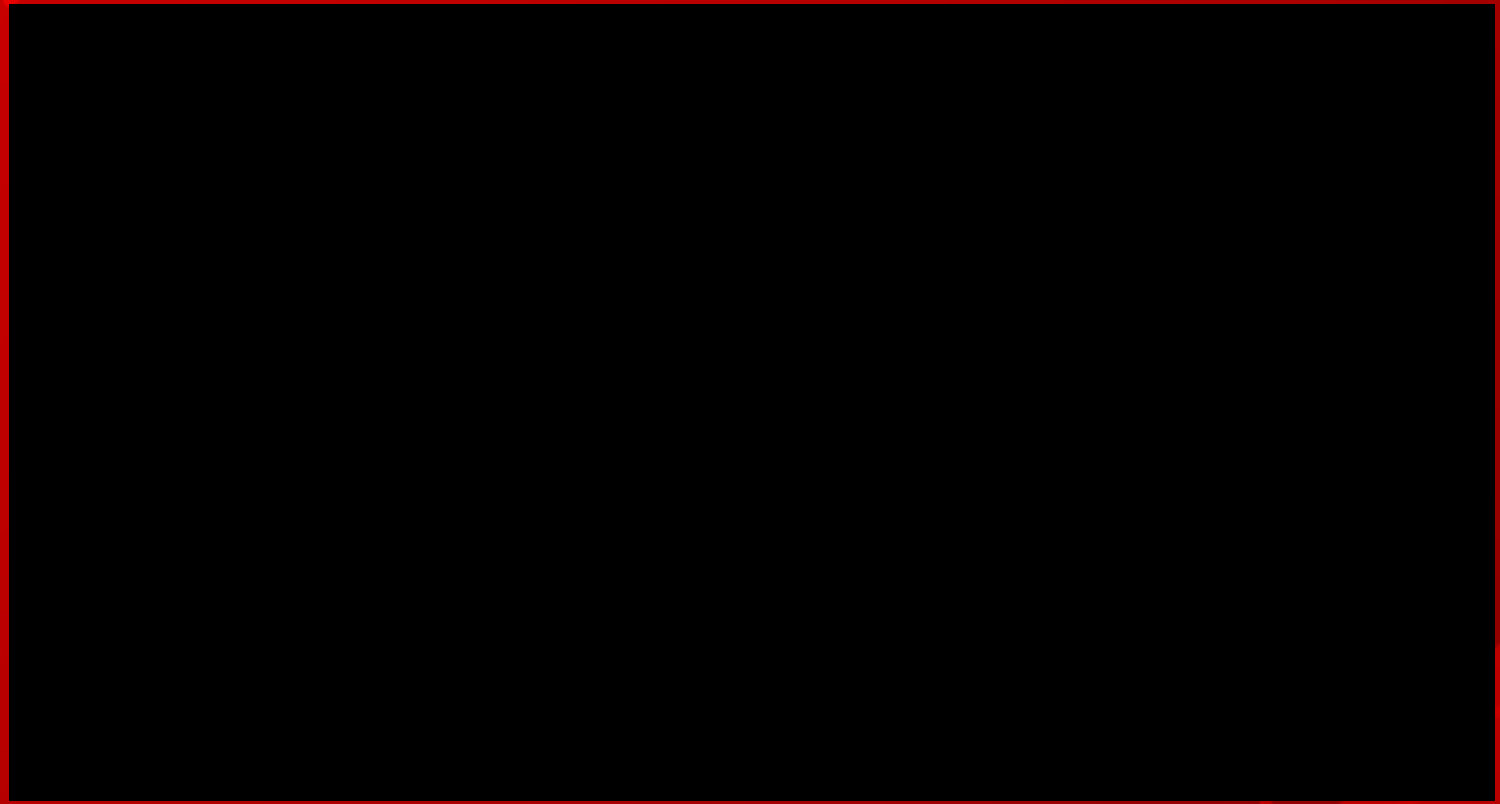
lowerBound: Min resources to set for a container

upperBound: Max resources to be set

target: VPA's recommended values, considering additional constraints

uncappedTarget: VPA recommendation without constraints

Demo



RED HAT SUMMIT

THANK YOU


linkedin.com/company/Red-Hat


facebook.com/RedHatinc

youtube.com/user/RedHatVideos


twitter.com/RedHat