https://medium.com/@Alibaba\_Cloud/kubernetes-resource-quotas-f2161607444e

<https://dzone.com/articles/kubernetes-namespaces-resource-quota-and-limits-fo>

<https://sysdig.com/blog/kubernetes-limits-requests/>

By default, containers run with unbounded compute resources on a Kubernetes cluster. With Resource quotas, cluster administrators can restrict the resource consumption and creation on a namespace basis. Within a namespace, a Pod or Container can consume as much CPU and memory as defined by the namespace’s resource quota. There is a concern that one Pod or Container could monopolize all of the resources. Limit Range is a policy to constrain resource by Pod or Container in a namespace.

Kubernetes supports 2 CPU quotas:

( One Millicores is 1/1000 of a CPU, therefore 1000m equals 1 CPU. 1000m equals one CPU on all computers. )

requests --> a Pod requests an amount of CPU resources

limit --> a Pod defines the limit of CPU resources it will use

Basic Quota:

apiVersion: v1

kind: ResourceQuota

metadata:

name: count-quotas

spec:

hard:

pods: "2"

######################################

apiVersion: v1

kind: ResourceQuota

metadata:

name: object-counts

spec:

hard:

requests.cpu: "1000m"

limits.cpu: "2000m"

#######################################

More Resources:

apiVersion: v1

kind: ResourceQuota

metadata:

name: quota2

spec:

hard:

cpu: "20"

memory: 1Gi

pods: "10"

replicationcontrollers: "20"

resourcequotas: "10"

services: "5"

#############################################

apiVersion: v1

kind: LimitRange

metadata:

name: limit-mem-cpu-per-container

spec:

limits:

- max:

cpu: "800m"

memory: "1Gi"

min:

cpu: "100m"

memory: "99Mi"

default:

cpu: "700m"

memory: "900Mi"

defaultRequest:

cpu: "110m"

memory: "111Mi"

type: Container

####################################################

apiVersion: v1

kind: LimitRange

metadata:

name: limits

spec:

limits:

- default:

cpu: 200m

memory: 512Mi

defaultRequest:

cpu: 100m

memory: 256Mi

type: Container

apiVersion: v1

kind: LimitRange

metadata:

name: cpu-memory-min-max-demo-lr

spec:

limits:

- max:

cpu: "200m"

memory: 128Mi

min:

cpu: "100m"

memory: 100Mi

type: Container

#No need to provide default and default values when min and max are given as by default the POD/Container will take max CPU,Memory as default.

apiVersion: v1

kind: Pod

metadata:

name: couchbase-pod

spec:

containers:

- name: couchbase

image: couchbase

ports:

- containerPort: 8091

resources:

limits:

cpu: "500m"

memory: 256Mi

apiVersion: v1

kind: Pod

metadata:

name: couchbase-pod1

spec:

containers:

- name: couchbase

image: couchbase

ports:

- containerPort: 8091

resources:

limits:

cpu: "500m"

memory: 1Gi

TOTAL YAML FOR NAMESPACE:

apiVersion: v1

kind: Namespace

metadata:

name: development

labels:

name: development

---

apiVersion: v1

kind: ResourceQuota

metadata:

name: object-counts

namespace: development

spec:

Hard:

#compute-resources

requests.cpu: "1000m"

limits.cpu: "2000m"

requests.memory: 1Gi

limits.memory: 2Gi

#object-counts

pods: "10"

replicationcontrollers: "20"

resourcequotas: "10"

services: "5"

---

apiVersion: v1

kind: LimitRange

metadata:

name: cpu-memory-min-max-demo-lr

namespace: development

spec:

limits:

- max:

cpu: "200m"

memory: 128Mi

min:

cpu: "100m"

memory: 100Mi

type: Container