

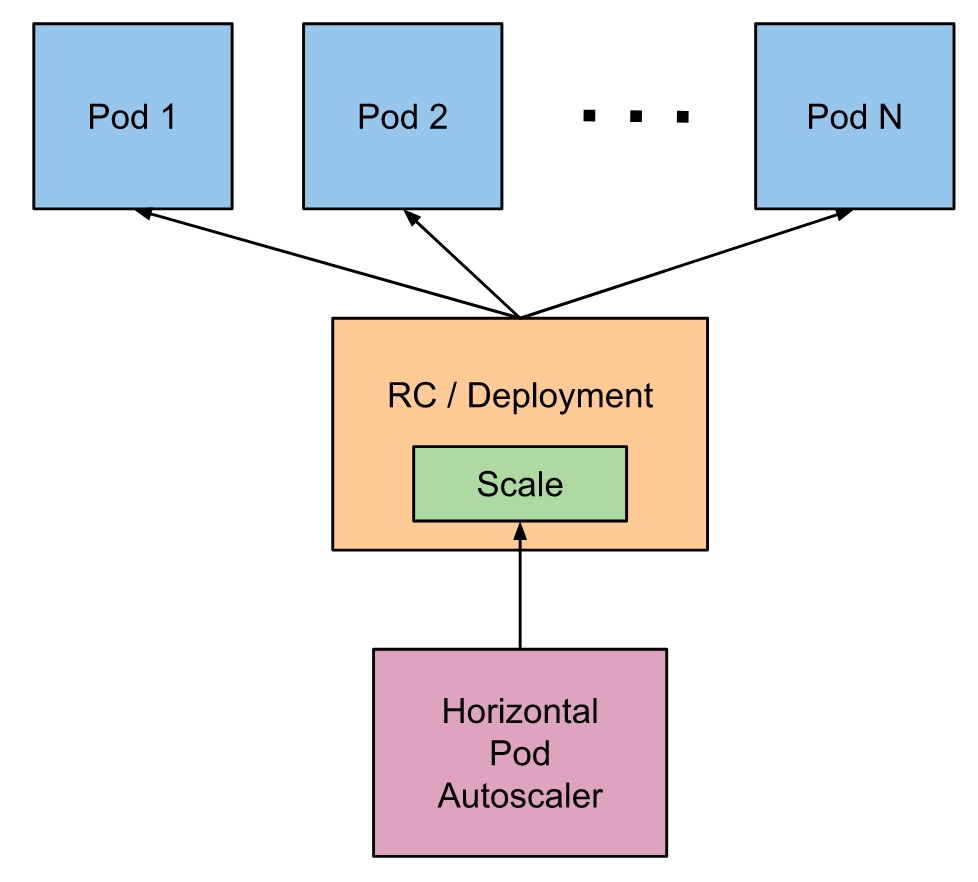
**Autoscaling**

**Autoscaling** is one of the key features in Kubernetes cluster. It is a feature in which the cluster is capable of increasing the number of nodes as the demand for service response increases and decrease the number of nodes as the requirement decreases. This feature of auto scaling is currently supported in Google Cloud Engine (GCE) and Google Container Engine (GKE) and will start with AWS pretty soon.

In order to set up scalable infrastructure in GCE, we need to first have an active GCE project with features of Google cloud monitoring, google cloud logging, and stackdriver enabled.

**Horizontal Pod Autoscaler**

Horizontal Pod Autoscaler automatically scales the number of pods in a replication controller, deployment or replica set based on observed CPU utilization (or, with beta support, on some other, application-provided metrics).



### CPU Based Scaling

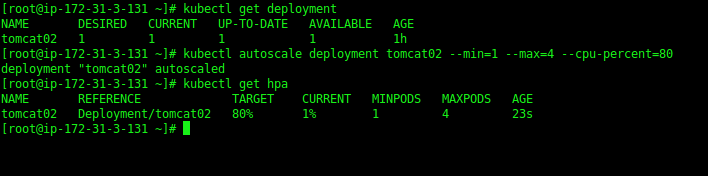
With Horizontal Pod Autoscaling, Kubernetes automatically scales the number of pods in a replication controller, deployment or replica set based on observed CPU utilisation.

Execute the command: “kubectl get deployment” to get the existing deployments.

Create a Horizontal Pod Autoscaler i.e. hpa for a particular deployment using command:

kubectl autoscale deployment <deployment-name> --min=2 --max=5 --cpu-percent=80

Execute “kubectl get hpa” to get the available hpa in your cluster.



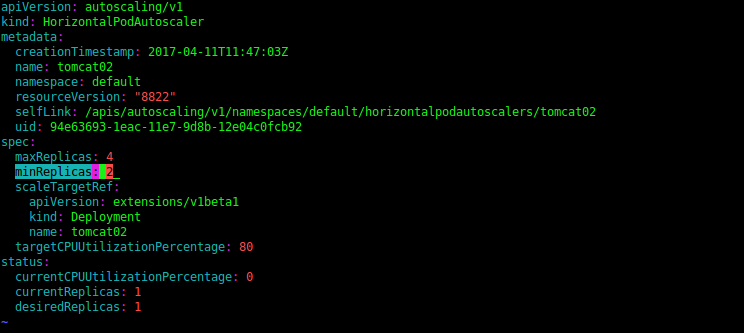
So, now we have a hpa running for our deployment “tomcat02”. It compares the arithmetic mean of the pods’ CPU utilization with the target defined in Spec.CPUUtilization, and adjusts the replicas of the Scale if needed to match the target (preserving condition: MinReplicas <= Replicas <= MaxReplicas).

But, how you will update the minimum no. of replicas in an existing HPA? In our case, currently, we have set the minimum no. of replicas to 1, what if we need to update the min. no. of replicas to 2. In this scenerio, just we need to get the hpa in yaml format and update the yaml file. Here’s an example,

kubectl get hpa/tomcat02 -o yaml > tomcat-hpa.yaml

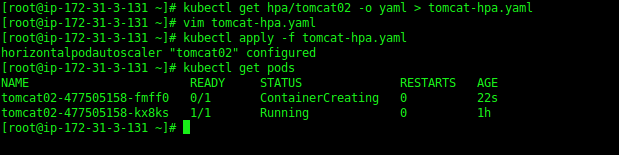
vim tomcat-hpa.yaml

Update the count as highlighted in below screenshot:



Save the yaml file and apply the changes:

kubectl apply -f tomcat-hpa.yaml



Once the changes have been applied, it will launch one more pod as shown in above screenshot.