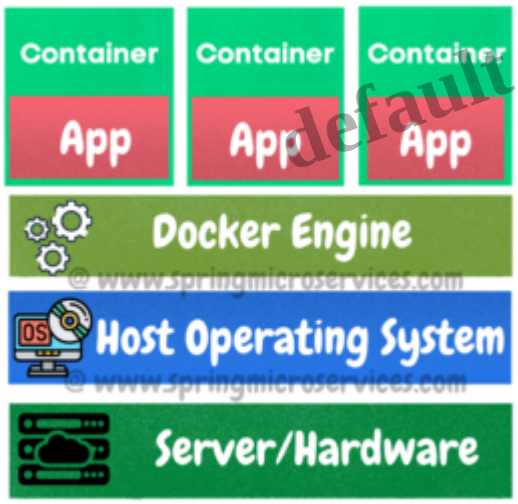
What is Kubernetes?

Description

Introduction

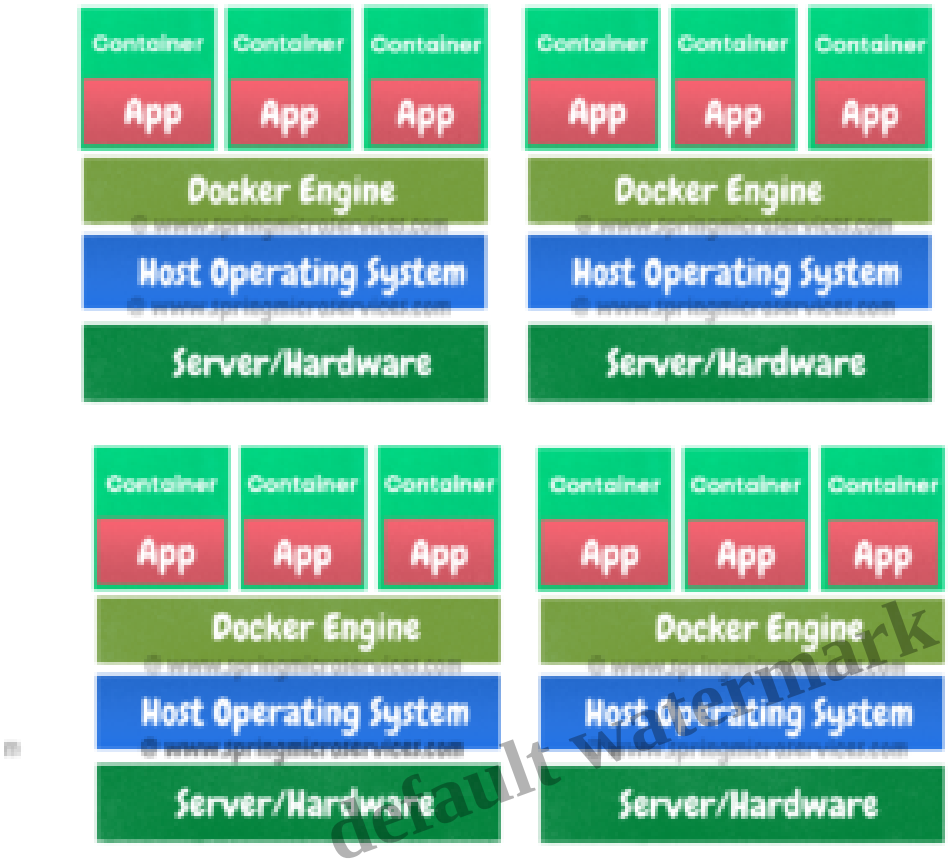
Before going over what is Kubernetes, do you know what are containers? If you are not aware of Containers, I would recommend you to go over our post on What are container.

Kubernetes is an extensible, portable, and open-source platform designed by Google in 2014. It is mainly used to automate the deployment, scaling, and operations of the container-based applications across the cluster of nodes. Kubernetes is also designed for managing the services of containerized apps using different methods which provide the scalability, predictability, and high availability. Nowadays many organizations have started adopting Kubernetes-based infrastructure. This technique or concept works with many container tools, like docker**,** and follows the client-server architecture.



Container Architecture with a single server

In the above container architecture image you see there is a single server. Consider if there are more servers and that keeps increasing, then managing those will be a difficult task. In the below image we have 4 servers and 12 containers.



Container architecture with multiple server and multiple containers

We need to know what is deployed to which node/server and if at all we want to deploy to a specific node/server, we need to have some configuration in place to help us with all these tasks. All these problems are addressed by Kubernetes ðŸ˜‡

So Kubernetes is also known as an container Orchestration *(Planning or coordination of the elements to produce a desired effect)* open source platform. Kubernetes is now part of cloud native computing foundation (CNCF) which is managing it. Kubernetes groups containers that makes up an application into logical units for easy management and discovery. Very often you will keep hearing the word “Pod”. Pods are nodes which are actually servers where different containers can be deployed. Pod can have either a single container or multiple containers.

Features of Kubernetes

Horizontal Scaling – This feature uses aÂ HorizontalPodAutoscalarÂ which automatically increases or decreases the number of pods in a deployment, replication controller, replica set, or stateful set on the basis of observed CPU utilization. Scaling can be controlled via configuration.

Self Healing – Whenever the container fails, Kubernetes tries to restart them automatically. Whenever a node dies it reschedules or replaces a container. It will automatically kill the old containers and removes out of the network, if they are not responding to the user requests. All the decision is taken based on the health checks done by Kubernetes which it does internally.

Load balancing & service discovery – This is one of the feature which I liked where Kubernetes assigns the IP addresses and a DNS Name for a set of containers. Kubernetes can also take care of load balancing and service discovery like any other cloud platforms.

Automated Rollouts and Rolls backs – Kubernetes makes the changes progressively. If there are any issues in the deployment, Kubernetes will automatically rollback the deployments.

Storage Orchestration – Kubernetes can automatically mount the storage system according to your choice, be it local storage or cloud storage like AWS etc. Kubernetes provides an essential feature called ‘persistent storage**‘**Â for storing the data, which cannot be lost after the pod is killed or rescheduled. Kubernetes supports various storage systems for storing the data, such asÂ **Google Compute Engine’s Persistent Disks (GCE PD)**

or **Amazon Elastic Block Storage (EBS)**.Â It also provides the distributed file systems:Â NFS or GFS.

Batch Execution – Everything that runs on Kubernetes is a workload. A workload can be a single component or several. There are several built-in workload resources available. Kubernetes provides two workload resources to create batch transactions i.e. a Job object and a CronJob object.

Configuration Management – Kubernetes provides a feature to change your configuration without rebuilding your image, so there is no need to redeploy your application when we change the configuration. Configuration will be outside the image.

Category

1. Cloud
2. Kubernetes

Date Created
December 28, 2021
Author
kk-ravi144gmail-com