KUBERNETES

==> It is a open-source container orchestration platform.

==> It will never create container, but it will manage the containers.

==> It solves problems like auto scaling , self healing , load balancing

==> It will automates how containerized applications deploy , scaling and managing across the cluster.

CLUSTER

Cluster ==> combination of nodes

Node ==> vm

Pod ==> a smallest deployable unit which are going to manage the containers.

==> Cluster is a combination of nodes to run , manage and scale containerized applications using kubernetes.

IT HAS TWO TYPES OF NODES:

1) MASTER NODE:

*It is the heart of the cluster

*It will manage the entire cluster and actual state.

*It will not run the application containers, where it will controls and monitor everything.

===> Main components inside the Master Node:

1.API server:

* This is the entry point to the cluster

*Firstly request goes through this api server and it validate and update the cluster state.

*It communicate with etcd.

2.etcd:

*It stores the entire Kubernetes information.

3.Controller Manager:

*It will ensure cluster always in desired state.

*It runs background controllers like Deployment controller , Replicaset controller and node controller.

* so its work is to changes the actual state to match the desired state

4.Scheduler:

*It will schedule cluster tasks.

*It will decide which node a pod should run on

*Then assignes pod to a worker node.

2) WORKER NODE:

*Its a machine where the application actually run.

* It will create number of worker node

*It is where the workload is going to deployed is called worker node

*Each worker node contains:

1.kubelet:

*It is an agent running on each worker node.

* It talks with API server

*It ensures container in pod are running

==>It will tell the container runtime to start the container image.

2.k-proxy:

*It will contain network info like node ip,pod ip and svc ip.

*It handles the networking , service routing and load balancing between pods.

3.container runtime:

* It will provide the container env to run the environment.

*It is actually runs the container.