**Kubernetes: (similar to Docker Swarm developed by Docker Swarm)**

* Is an open-source, portable for automatic container deployment and management.
* Kubernetes Is a container management system developed by Google. (2014).
* Kubernetes eliminates many of the manual processes involved in deploying and scaling containerized applications.

**Features of Kubernetes**:

Automates various manual processes and controls server hosting and launching.

Manages containers, offers security, networking and storage services.

It monitors and continuously checks the health of nodes and containers.

Automatic rollback for changes that go wrong.

Mounts and adds a storage system to run apps.

**Hardware component – Node**:

* A node is the smallest unit of hardware in Kubernetes. It is a representation of a single machine in the cluster.
* A node is a physical machine in a datacenter (or) virtual machine hosted on a cloud provider like google cloud platform.

**Hardware component – Cluster**:

* Kubernetes does not work with individual nodes. It works with the cluster as whole.
* Nodes combine their resources to form a powerful machine known as cluster.
* When a node is added or removed, the cluster shifts around the work as necessary.

**Hardware component – Persistent Volumes:**

* To store data permanently. / Kubernetes uses Persistent Volumes.
* Nodes combine their resources to form a powerful machine known as cluster.
* When a node is added or removed, the cluster shifts around the work as necessary.

**Software component – Container**

* Containers are used everywhere as they create a self-contained environment for the execution of programs.
* The programs are bundled up into a single file (known as container) and then shared on the internet.
* Multiple programs are added to a single container. Limit to one process per container.

**Software component – Pods**:

* A pod represents a group of one or more application containers bundled up together and are highly scalable.
* If a pod fails, Kubernetes automatically deploys new replicas of pod to the cluster.
* Pods provides two different types of shared resources-networking and storage.

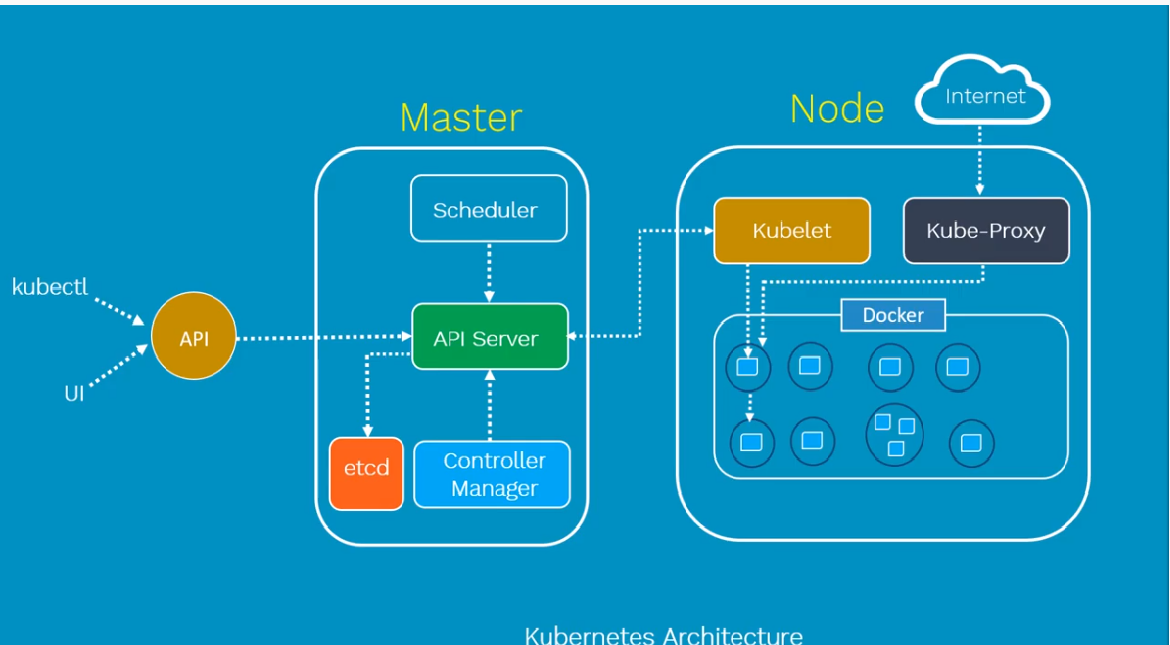
**Software component – Deployment**:

* Pods cannot be launched on a cluster directly, instead they are managed by one more layer of abstraction- the deployment.
* A deployment’s fundamental purpose I to indicate how many pods are running at same time.
* The manual management of pods is eradicated when deployment is used.

**Software component – Ingress:**

* Ingress allows access to Kubernetes services from outside the cluster.
* The multiple was to add ingress to the cluster can done through either an ingress controller or a load balancer.
* It can provide load balancing, SSL termination and name – based virtual hosting.

**Kubernetes Architecture:**



**Master Node:**

* The master node is the most vital component responsible for Kubernetes architecture.
* It is the central controlling unit of Kubernetes and manages workload and communications across the clusters.
* The Master node has various components, each having its process. They are

1. Cluster Store: store configuration details, receive commands, manage network rules.
2. Controller Manager: actually, perform the tasks that are stored in cluster store.
3. Scheduler: actually, controls distribution of workload.
4. API Server: control all operations on the cluster and take instruction.

**Slave Node:**

* Contains all different nodes and pods.
* Each pod contains one or more containers.
* Kubernetes Node contains,

1. Docker: one of basic requirement of node is Docker/responsible for pulling down and running container from Docker images
2. Kebelet: is responsible for managing pods and their contains. It deals with pods specifications which are defined in JSON or YAML format.
3. Kubernetes Proxy: is tool that interact with master environment within the Kubernetes network and allows all the instructions pushed to kubelet and docker to perform effectively.

**Configuring Kubernetes by Kubeadm:**

Installing:

Creating Kubernetes cluster (master and worker nodes):

Steps:

* Create VMs which are part of k8s cluster (means 4 Centos VMs instances).
* Disable SELinux and SWP on all nodes.

swapoff -a // is for performance related

setenforce 0

* Install Kubeadm, Kubectl and Docker on all nodes.

start and enable docker and kubelet on all nodes

yum update -y

yum install -y docker

systemctl start docker

systemctl enable docker

systemctl status docker

yum install -y kubeadm kubelet –disableexculdes=Kubernetes

systemctl enable kubelet && systemctl kubelet

* Initialize the master node.

Kubeadm init –pod-network-cidr=10.240.0.0/16

* Configuring Pod networks (we use flannel).
* Join worker nodes to the cluster.

Add the copied the out put form Kubeadm init command.

|  |  |
| --- | --- |
| Kubernetes | Docker |
| Kubernetes is an open-source platform used for  maintaining and deploying a group of containers | Docker is a tool used to automate the deployment of applications in light weight containers so that applications can work efficiently in different environments |
| Applications are deployed as a combination of pods, deployments and services | Applications are deployed in the form of services |
| Autoscaling feature is available and is about managing the infrastructure. | Autoscaling feature not available and is to manage solution. |