introducing

Kubernetes

Agenda

Kubernetes API and K8s Features
Docker Swarm vs Kubernetes
Kubernetes Architecture – apiserver |etcd|Schedular | Master | Worker Nodes|
Kubernetes Installation Process (Linux/Ubuntu) | Kubeadm | Kubectl| kubelet |
Kubernetes Cluster setup – workloads | Pods deployment | Auto-scaling |
Running *first app on Kubernetes* - scenario with working example
Deployment of *Nginx web application* via CLI accessing from external network
Manifest | Pod definition file (YAML)
Pod Autoscaling | Rolling Updates | Pod CPU and Memory reservation |
Kubernetes Internal/External Services|ClusterIP| NodePort| LoadBalancer| Ingress
Resource Usage Monitoring | Resource Quota | Scenario with working example
Kubernetes Jobs | Secrets management | Creating Secrets |
Kubernetes Networking | Volumes| Storage |
Overview on Blue-Green Deployment/Canary
Kubernetes cluster used in multi-environment | Dev |QA | Staging | Prod Teams
Kubernetes App Services | Introduction to Azure | GCP | Practice Labs
Hands-on with | Production use-cases | live customer scenarios | Assignments

Kubernetes or K8s

Container + Orchestration

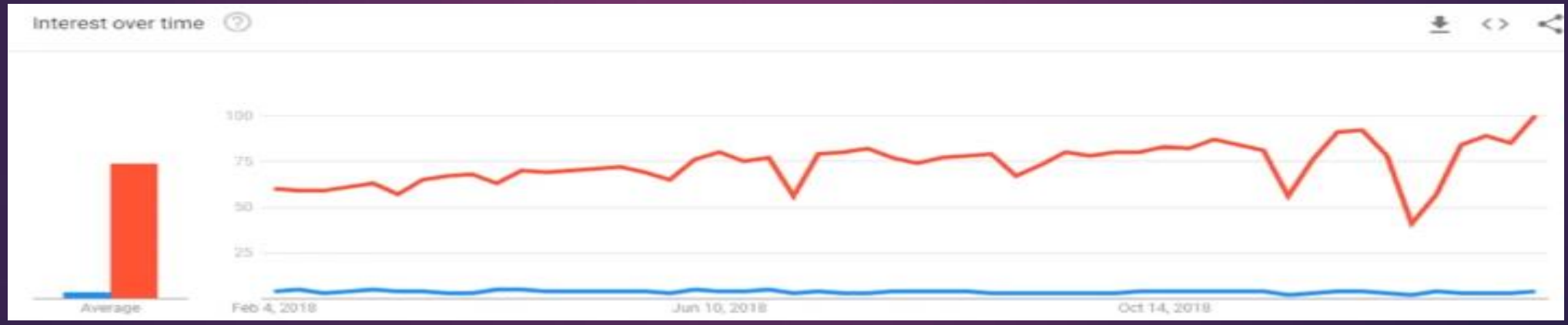
Introduction to Kubernetes

- Kubernetes is an open-source container orchestration software or a Tool.
- Originally it was developed by Google and was released in July 21st 2015.
- It is the ninth most active repository on GitHub in terms of number of commits.

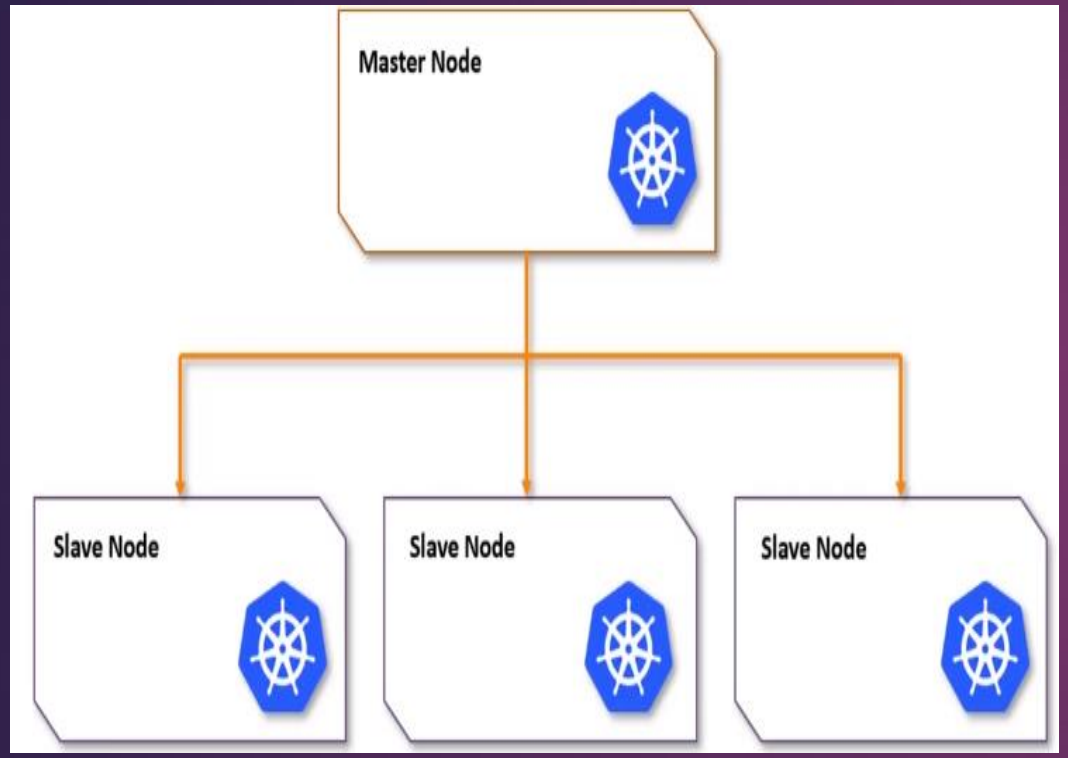
Kubernetes Features

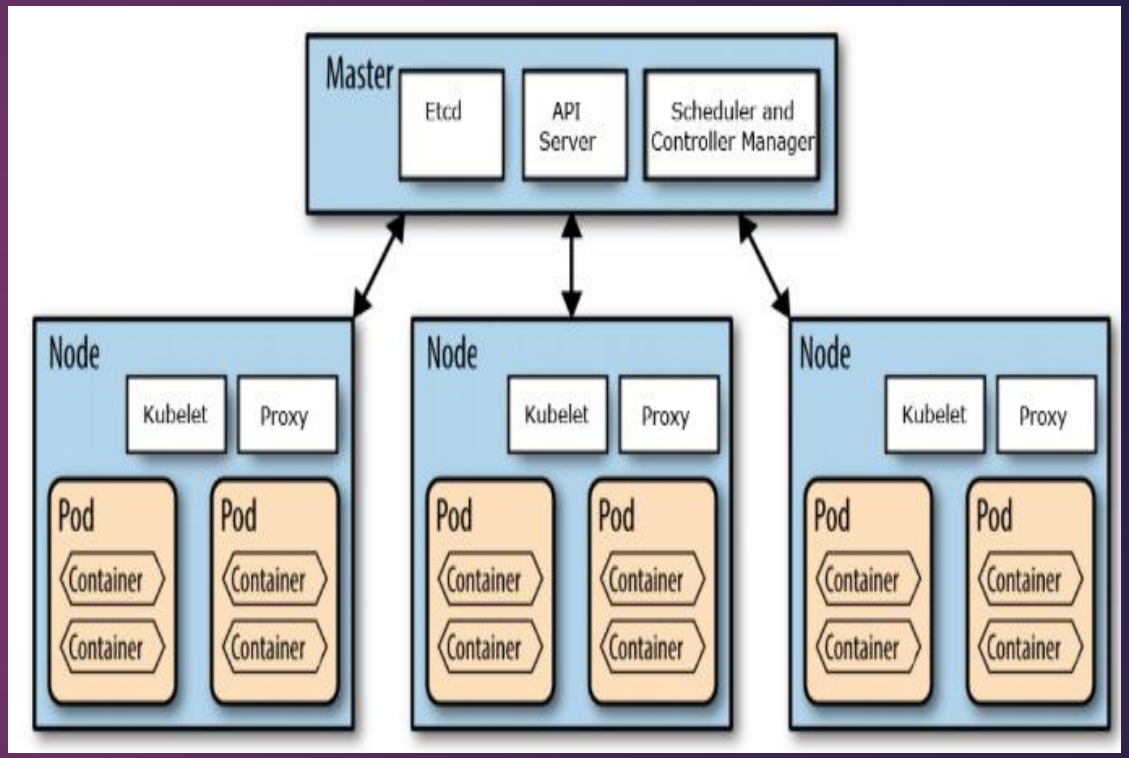
- Pods
- Replication Controller
- Storage Management
- Resource Monitoring
- Health Checks
- Networking
- Secret Management
- Rolling Updates
- Service Discovery

Docker Swarm vs Kubernetes



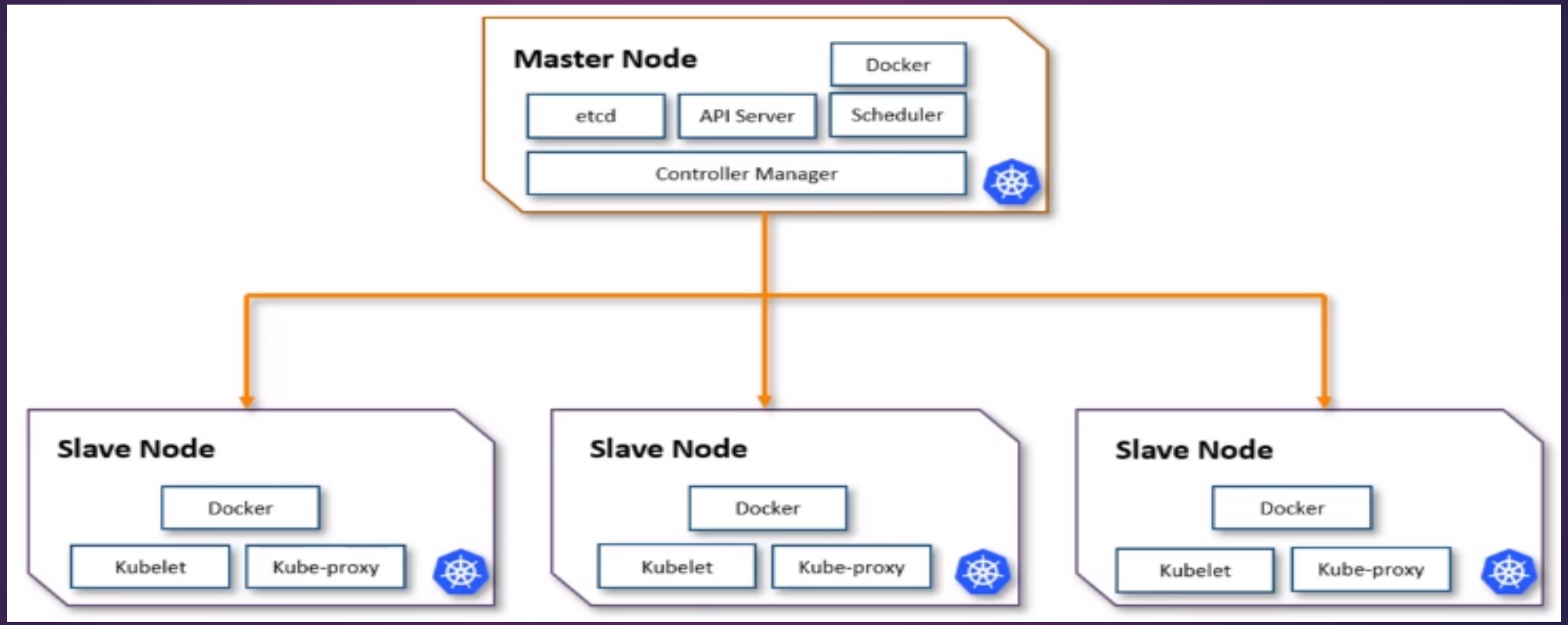
Kubernetes Architecture





Kubernetes Architecture

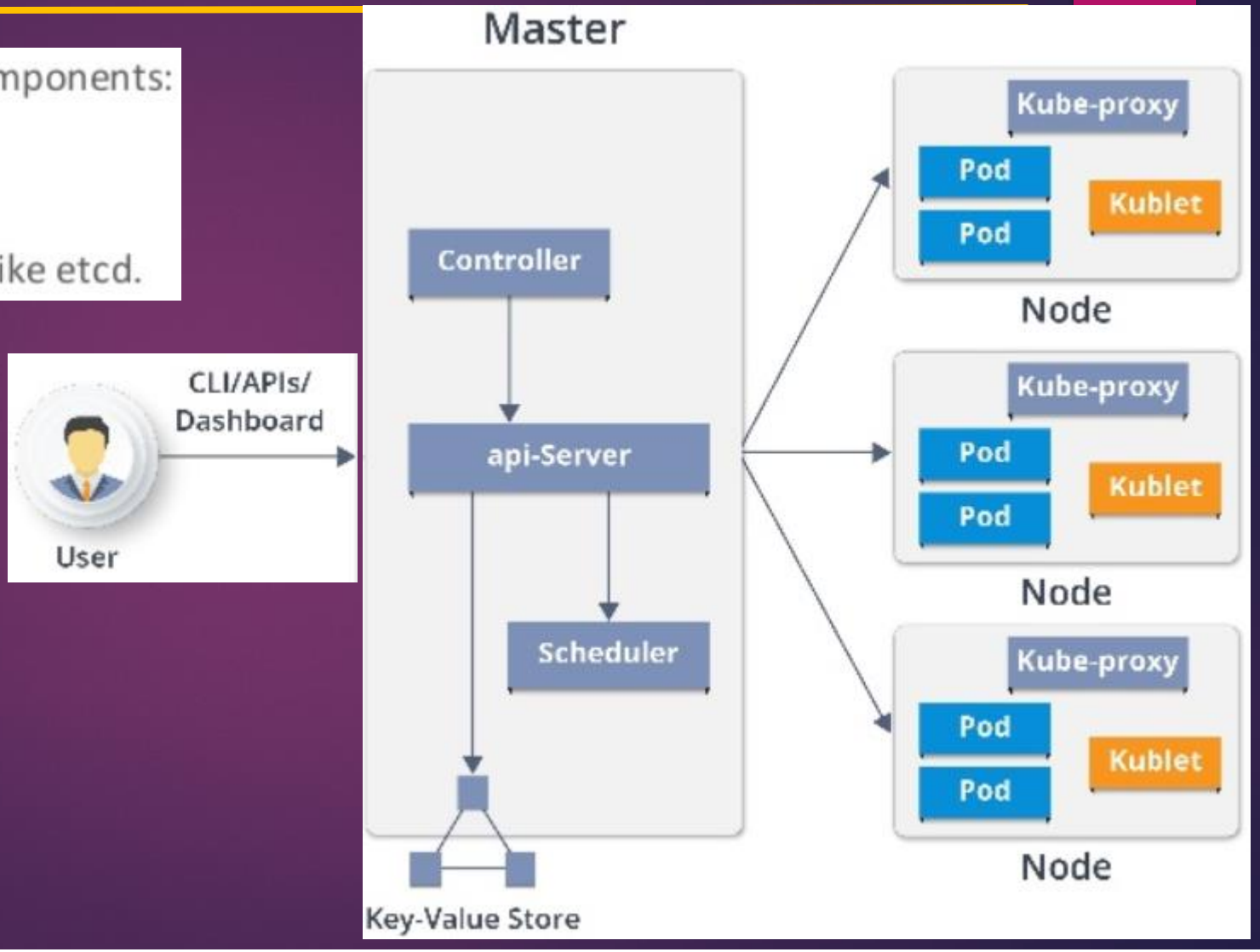
Master-Slave Components:



Kubernetes Architecture

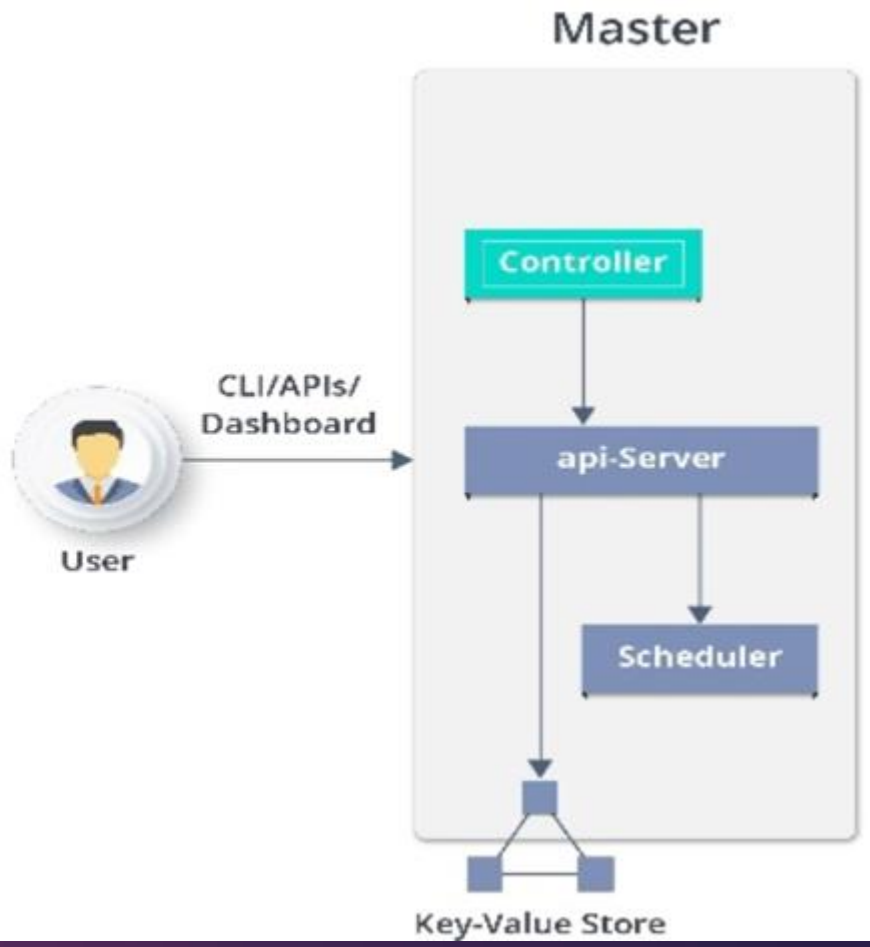
Kubernetes has the following main components:

- Master nodes
- Worker nodes
- Distributed key-value store, like etcd.



Kubernetes Architecture

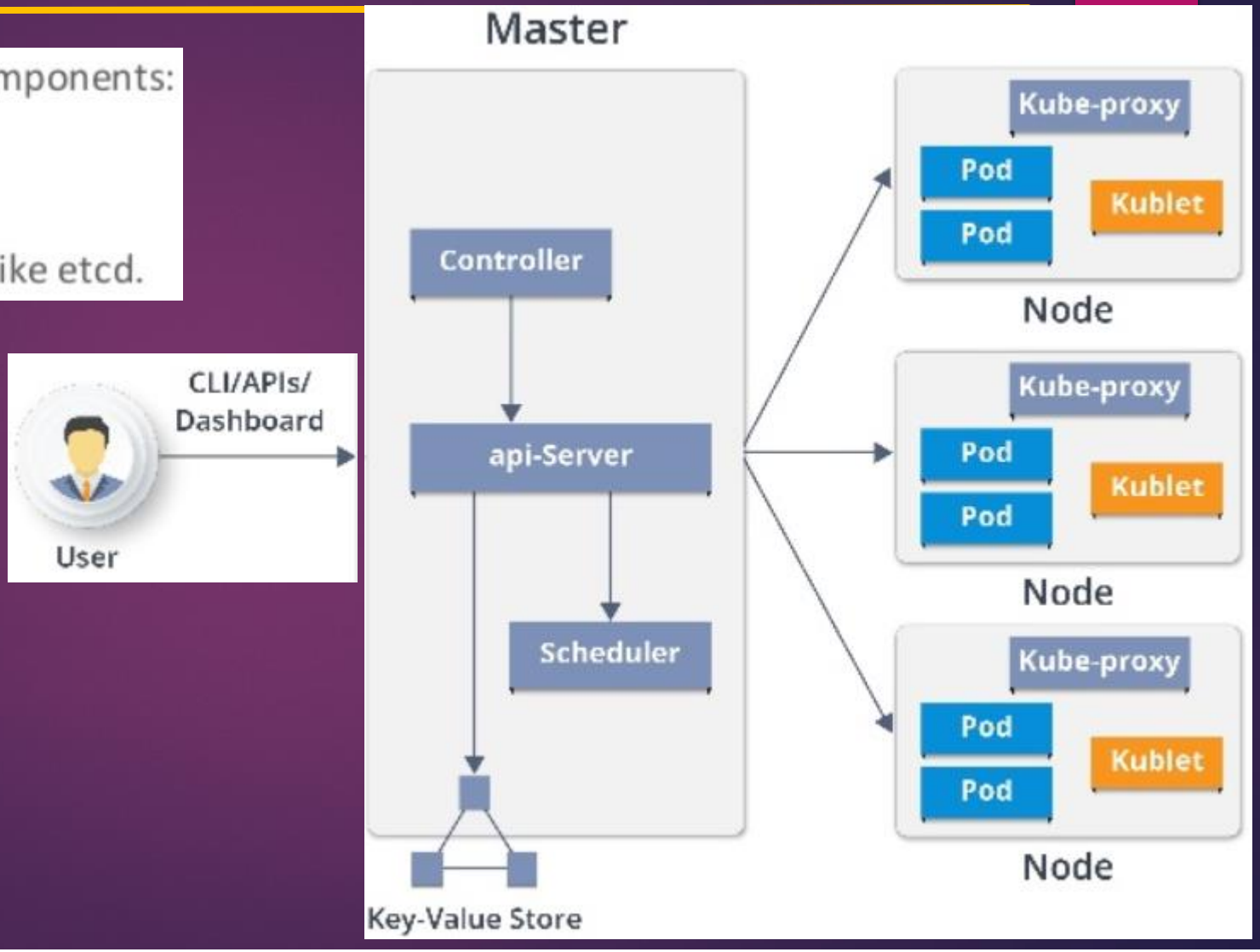
- Responsible for managing the Kubernetes cluster
- It is the entry point for all administrative tasks
- Can be more than one master node in the cluster
- Only one of them will be the leader
- Kubernetes uses etcd



Kubernetes Architecture

Kubernetes has the following main components:

- Master nodes
- Worker nodes
- Distributed key-value store, like etcd.

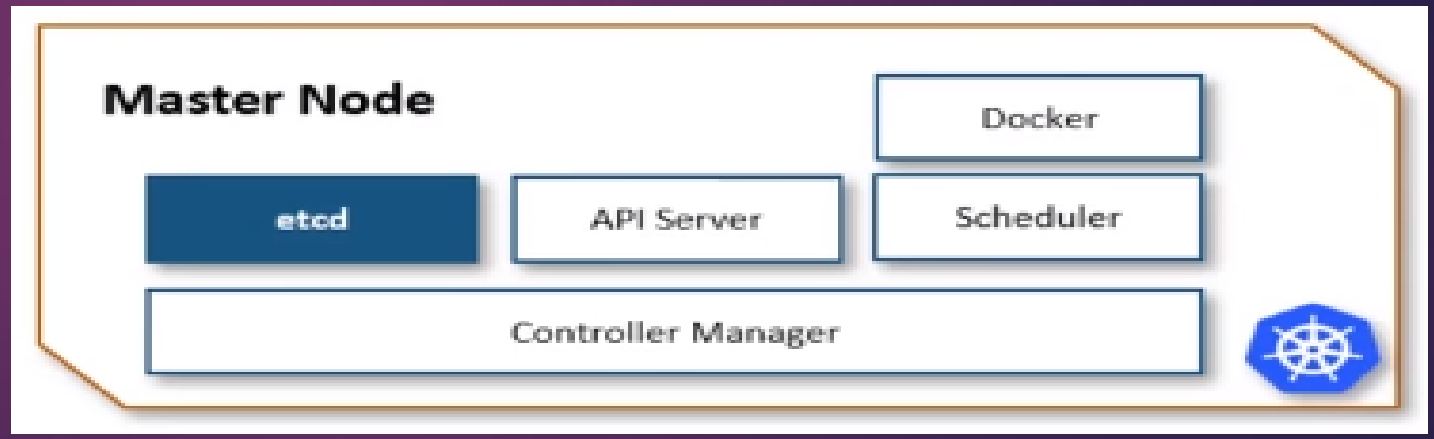


Kubernetes Architecture: Master Components

- etcd
- API Server
- Scheduler
- Controller Manager

This is highly available distributed key value store, which is used to store cluster wide secrets.

It is only accessible by Kubernetes API Server, as it has sensitive information.

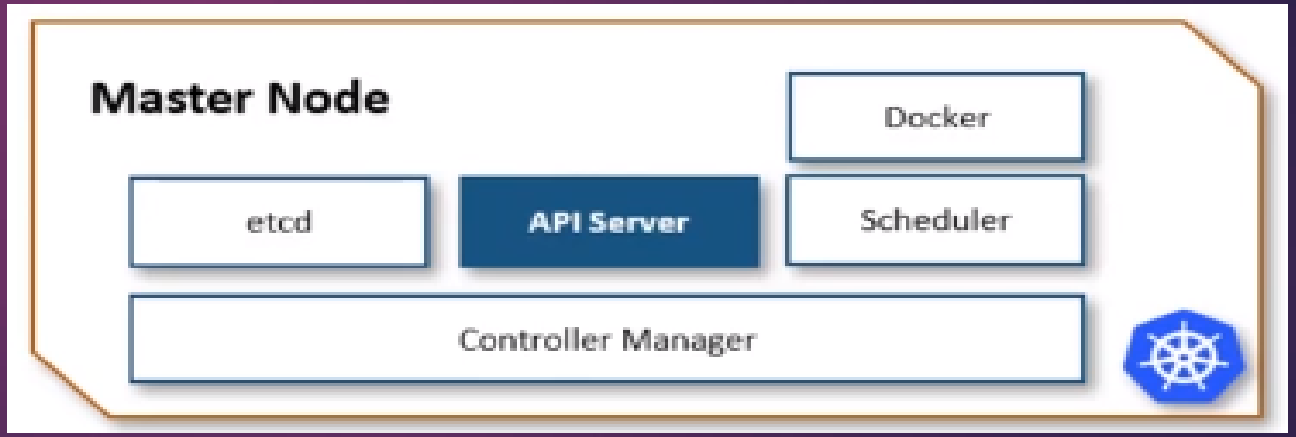


Kubernetes Architecture: Master Components

- etcd
- API Server
- Scheduler
- Controller Manager

It exposes the Kubernetes API. The kubernetes API is the front-end for Kubernetes Control Plane.

It is used to deploy and execute all operations in Kubernetes.

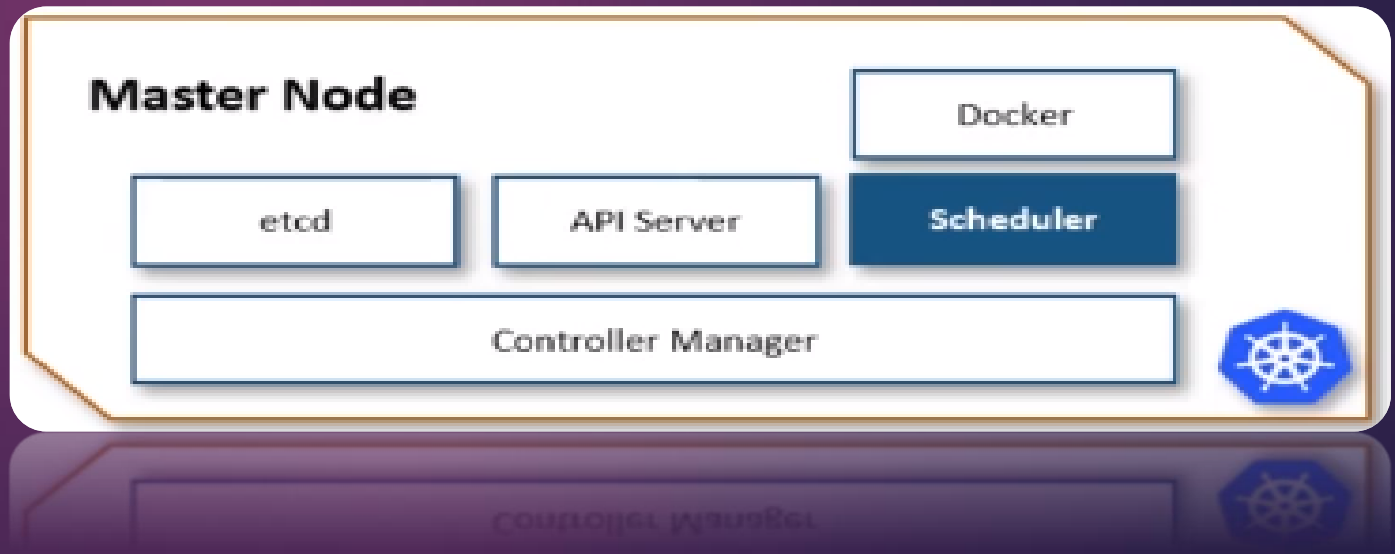


Kubernetes Architecture: Master Components

- etcd
- API Server
- Scheduler
- Controller Manager

The scheduler takes care of scheduling all the processes, Dynamic Resource Management.

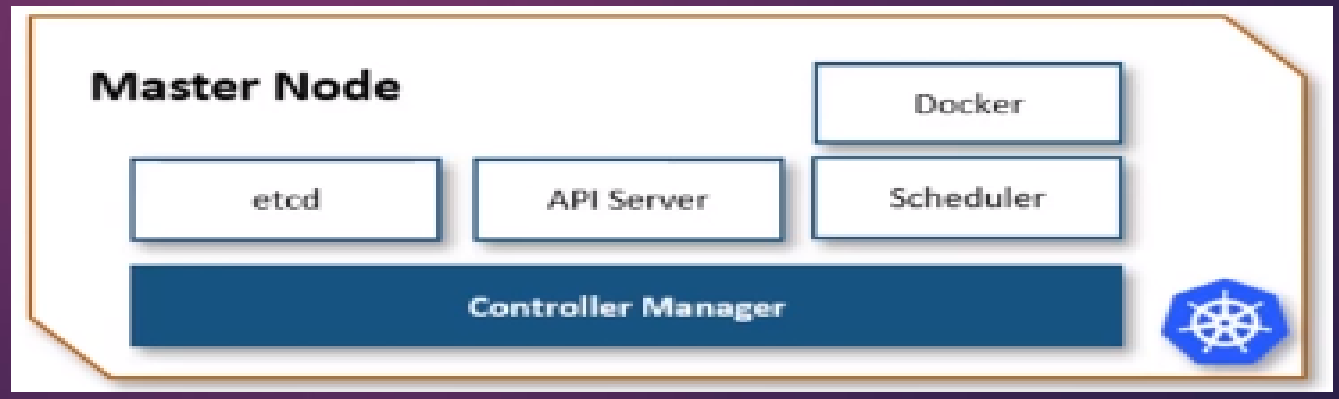
It also helps and manages present and future events on the cluster.



Kubernetes Architecture: Master Components

- etcd
- API Server
- Scheduler
- Controller Manager

The controller manager, runs all the controllers on the Kubernetes Cluster. Although each controller is a separate process, but to reduce complexity, all the controllers are compiled into a single process. Such as Node Controller, Replication Controller, Endpoints Controller, Service Accounts and Token Controllers.



Kubernetes Architecture: Slave Components

Kubelet

Kube-Proxy

Kubelet takes the specification from API Server and ensures that application is running according to the specifications which were given to them.

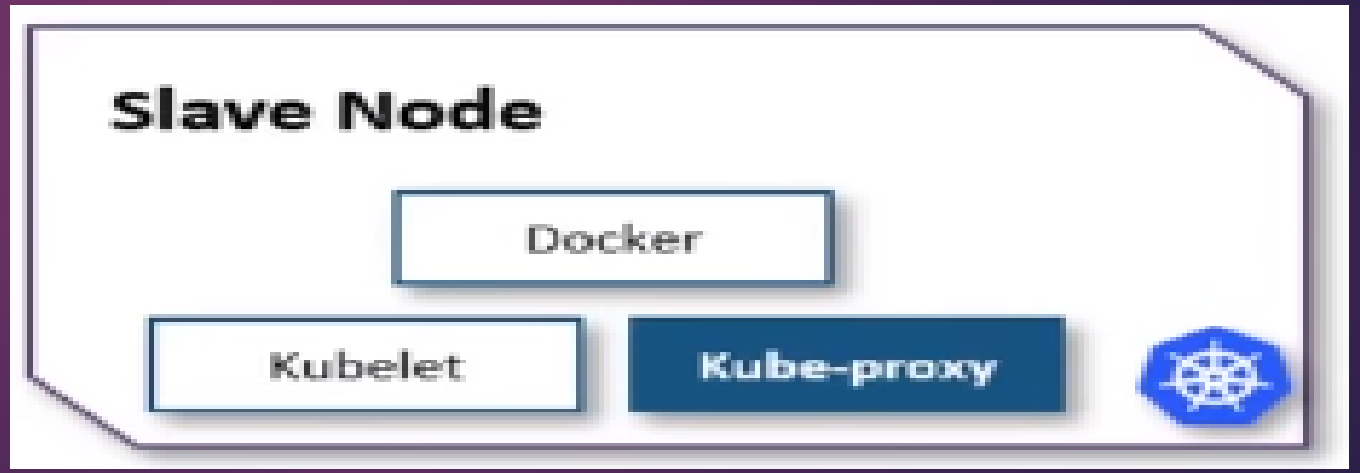
Each node has it's own kubelet service.

Kubernetes Architecture: Slave Components

Kubelet

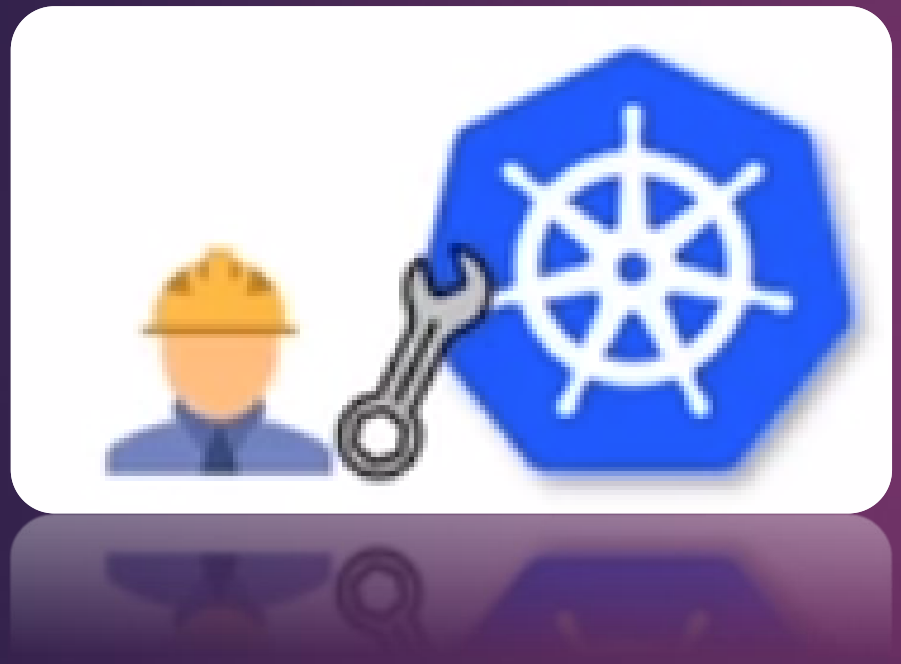
Kube-Proxy

Kube-Proxy has a service which runs on each node and helps in making services available to the external host. It helps in connection forwarding to the correct resources, it is also capable of doing primitive load balancing



Installation of Kubernetes

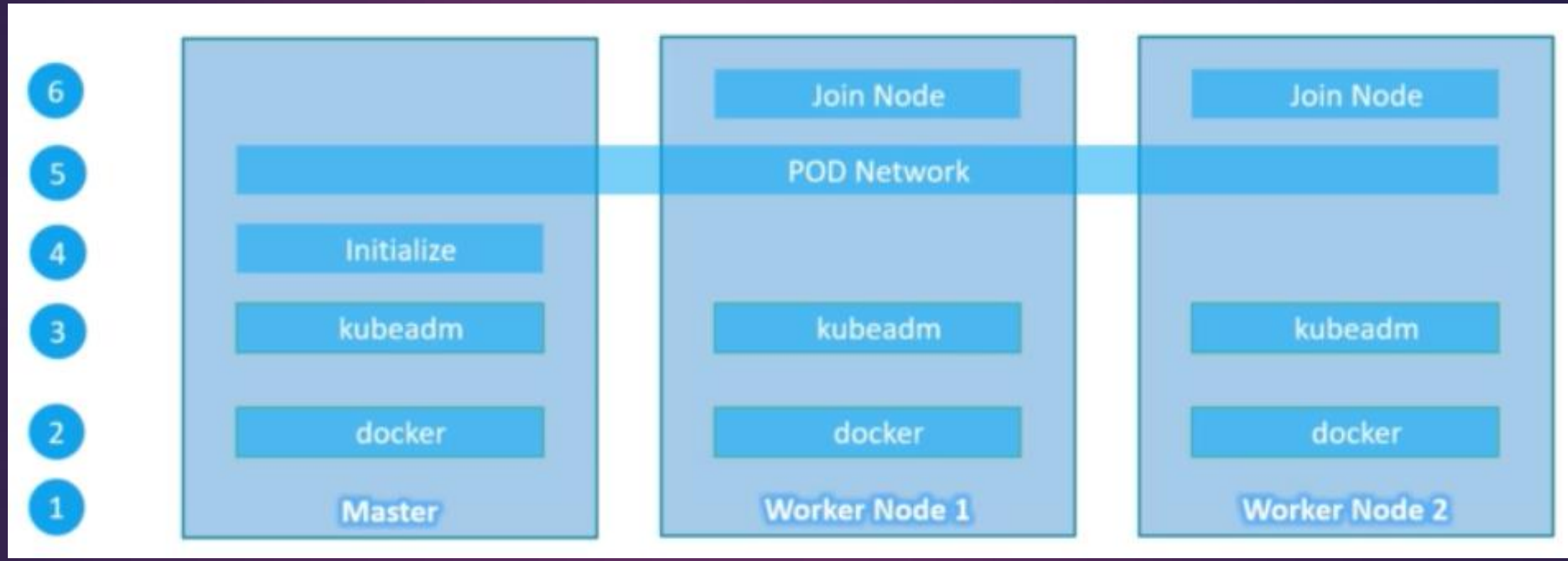
Kubernetes Installation



There are numerous ways to install Kubernetes, following are some of the popular ways:

- **Kubeadm** – Bare Metal Installation
- **Minikube** – Virtualized Environment for Kubernetes
- **Kops** – Kubernetes on AWS
- **Kubernetes on GCP** – Kubernetes running on Google Cloud Platform

Kubernetes Installation : Steps



Kubernetes Installation : Steps

Always Refer official links:

https://kubernetes.io/docs/setup/production-environment/

https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/create-cluster-kubeadm/

https://kubernetes.io/docs/setup/best-practices/cluster-large/

https://kubernetes.io/docs/reference/kubectl/overview/

Kubernetes Building large clusters

Kubernetes supports clusters with up to 5000 nodes

Upto 5000 nodes per cluster

Upto 150000 total pods

Upto 300000 total containers

Upto 100 pods per node