#KubernetesIn30Days challenge :-

#Day24 :-

Interview Questions on Kubernetes -1
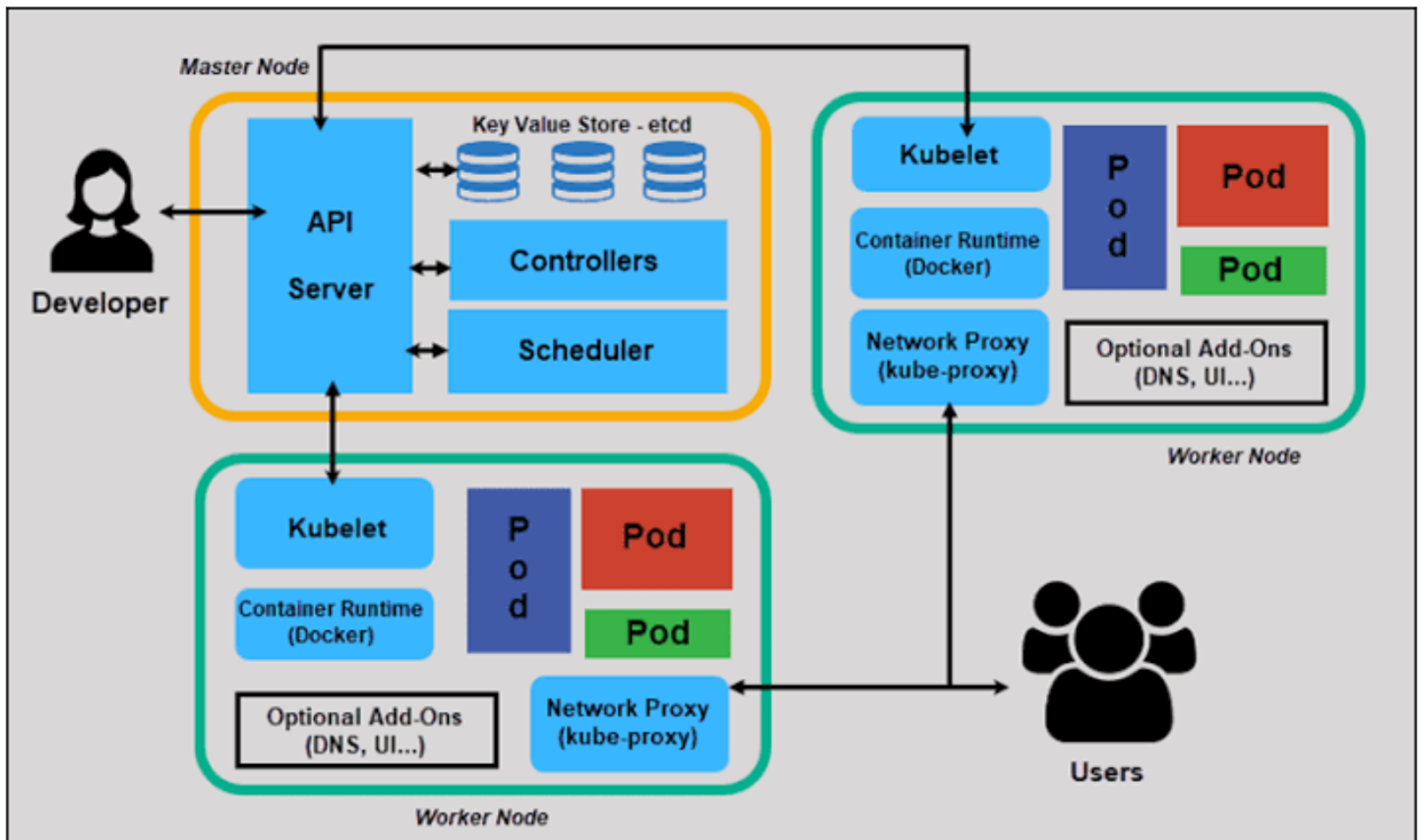
① What is the Architecture of Kubernetes?

Ans:- Basic Architecture of Kubernetes is as

follows:

1) Master Node → API server
→ etcd
→ controller
→ scheduler

2) worker Node
→ Kubelet
→ kube-proxy
→ container Runtime

@Rajasekhar Betha

NOTES

2) Explain the functionality of each component of k8s cluster.

Ans:- **API server**: It process the requests/responses from Kubelet/k8s dashboard. It always listen to the user's requests.

etcd: A Highly available Key-value Database which holds all the transactions of k8s cluster

scheduler: The component which will schedule the pod on worker Node

controller:- It'll control lot of things such as Replicas, Endpoints and Deployment controllers. These will be used for respective functions.

Kubelet:- It'll pull the image and also ensure that POD on Node is running.

Kube-proxy:- It'll be used for managing rules such as iptables at OS-level by taking ingress rules and services.

Container Runtime: The Runtime which is needed for a container to run such as Docker, containerd or any of container runtime will be used.

3) What is manifest file and the components of it.

Ans:- Manifest File:- It is the definition file of kubernetes object such as POD, ~~Deploymet~~, service, Ingress .... . It is called manifest file. It is a YAML file.

There are four basic fields which are

    (1) apiVersion - Version of resource.

    (2) kind - type of resource

    (3) metadata - metadata of resource

    (4) specs - specifications of resource

@Rajasekhar Betha

4) What is Pod affinity, Node Affinity, taint toleration.

Ans:- Node Affinity :- Influences where a pod can be scheduled based on characteristics of a Node. (like labels)

Example :- You want your pod to run only on Nodes labelled as 'SSD' for better storage performance

Pod Affinity :- Influences where a POD can be scheduled based on the presence of other pods.

Example :- You want your pod to be close to

other pods belonging to same application for better communication.

Taints and Tolerations:-

Taints (on Nodes):- Repel certain Pods from a node based on characteristics (key, value, effect)

Tolerations (on Pods):- Allow pods to tolerate (ignore) certain Node characteristics (taints) and still be scheduled there.

Example: You have nodes with limited resources and you only want specific pods to run on them. you use taints on these nodes and pods needing those resources use tolerations to run on them.

Thats all for Today!

Follow me on LinkedIn : Rajasekhar Betha.