

K8s Bible - <https://kubernetes.io/docs/tasks/tools/>

Following YT video covers from basic to advanced topics - playlist has 36 videos - tutor explains every component in a structured manner and is easy to understand. (**Topics not covered** : Taints/tolerants, node affinity, Kustomize, crictl, coreDNS)

<https://www.youtube.com/watch?v=jgmdY73RF6w&list=PLMPZQTftRCS8Pp4wiiUruly5ODScvAwcQ>

PPT links - if you are barely new to k8s, its highly recommended to start from here :

https://drive.google.com/file/d/17ueftU41jWNSsUJ4zshzUp9bZ8Z4hN-B/view?usp=drive_link

https://drive.google.com/file/d/16QNglIArP9RO3bv9sUHa_CpUPH2PcrOR/view?usp=drive_link

Course from Udemy - <https://www.udemy.com/course/kubernetes-made-easy/>

Use Coupon code for offers : UDEMYOCT19

Tutor's email for queries or donation -

srinathchalla@outlook.com

TOOLS

For learning commands and getting your hands dirty with k8s - <https://labs.play-with-k8s.com/>
(Has 4 hrs timeout for each session, can create up to 3nodes, wonderful SAAS site and FREE.)

For Exam Practice <https://killercoda.com/> - has Exam scenarios created, you can directly start practicing without setting the environment , not recommended until you learn the concepts clearly otherwise you get intimidated with the questions and scenarios.

I practiced all my k8s commands and setting up cluster thru the killercoda site, setting up cluster takes about 5minutes from scratch. Then you can start practicing directly on the website.

If you like to install your own local environment, then you must install minikube tool, follow

-<https://k21academy.com/docker-kubernetes/how-to-install-minikube/>

(Remember you also need a VM setup (Either Virtualbox or any desktop virtualization tool)

Exam simulator question and answer - <https://www.youtube.com/watch?v=Zm5sy6otLGc>

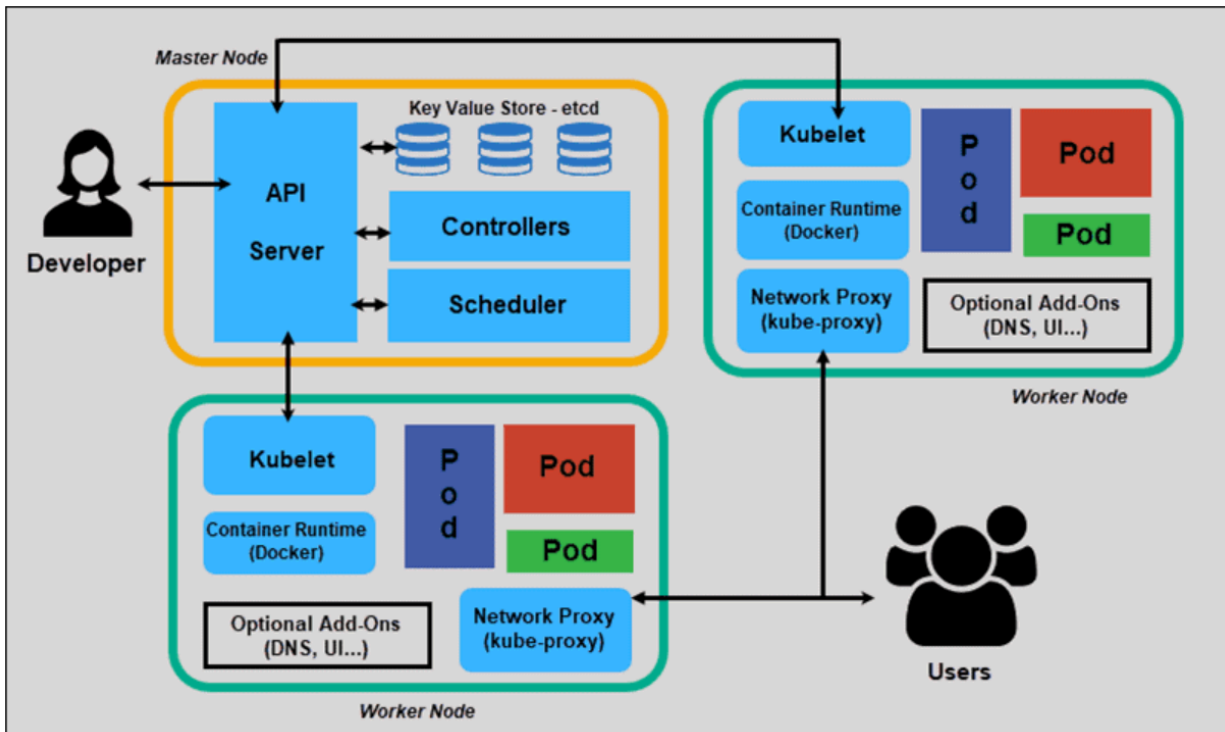
Certification portal From Linux Foundation CNCF-

<https://training.linuxfoundation.org/certification/certified-kubernetes-administrator-cka/>

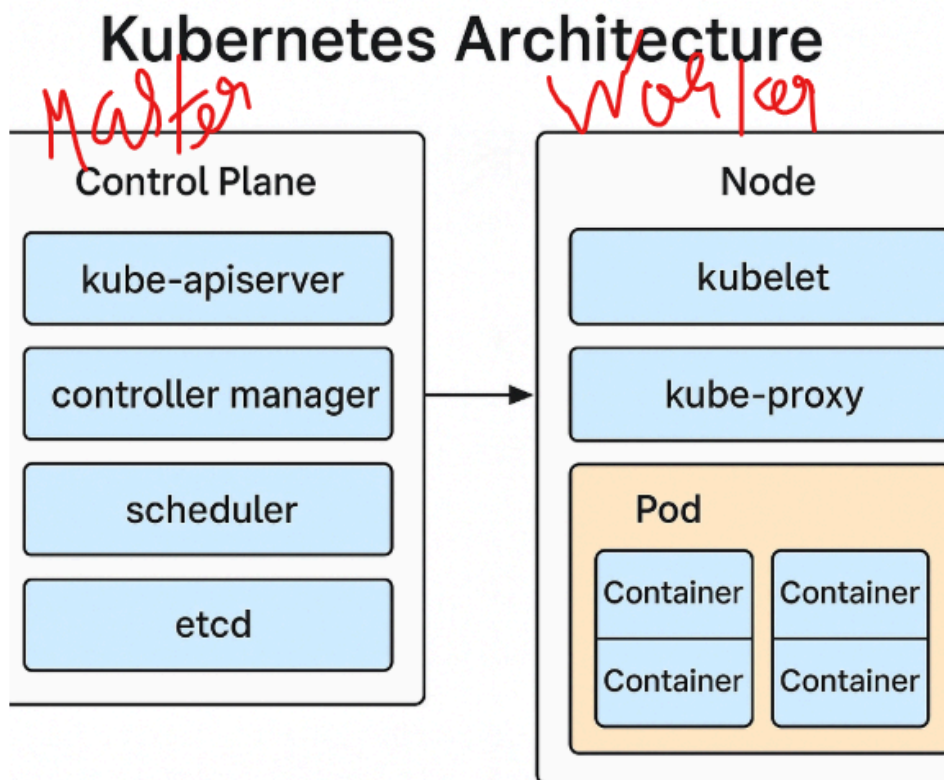
Use Coupon : DCUBE30 - 20% offer (July 2025 - when i was documenting this, for more latest ones, look here - <https://github.com/techiescamp/linux-foundation-coupon>)

My study notes :

Architecture diagram :



Master node / worker node key components:



DETAILED : Worker node components:

Kubelet

The kubelet runs on every node in the cluster. It is the principal Kubernetes agent. By installing kubelet, the node's CPU, RAM, and storage become part of the broader cluster. It watches for tasks sent from the API Server, executes the task, and reports back to the Master. It also monitors pods and reports back to the control panel if a pod is not fully functional. Based on that information, the Master can then decide how to allocate tasks and resources to reach the desired state.

Container Runtime

The container runtime pulls images from a container image registry and starts and stops containers. A 3rd party software or plugin, such as Docker, usually performs this function.

Kube-proxy

The kube-proxy makes sure that each node gets its IP address, implements local *iptables* and rules to handle routing and traffic load-balancing.

Pod

A pod is the smallest element of scheduling in Kubernetes. Without it, a container cannot be part of a cluster. If you need to scale your app, you can only do so by adding or removing pods.

The pod serves as a 'wrapper' for a single container with the application code. Based on the availability of resources, the Master schedules the pod on a specific node and coordinates with the container runtime to launch the container.

DETAILED MAster node components :

API Server

The API Server is the front-end of the control plane and the only component in the control plane that we interact with directly. Internal system components, as well as external user components, all communicate via the same API.

Key-Value Store (etcd)

The Key-Value Store, also called etcd, is a database Kubernetes uses to back-up all cluster data. It stores the entire configuration and state of the cluster. The Master node queries etcd to retrieve parameters for the state of the nodes, pods, and containers.

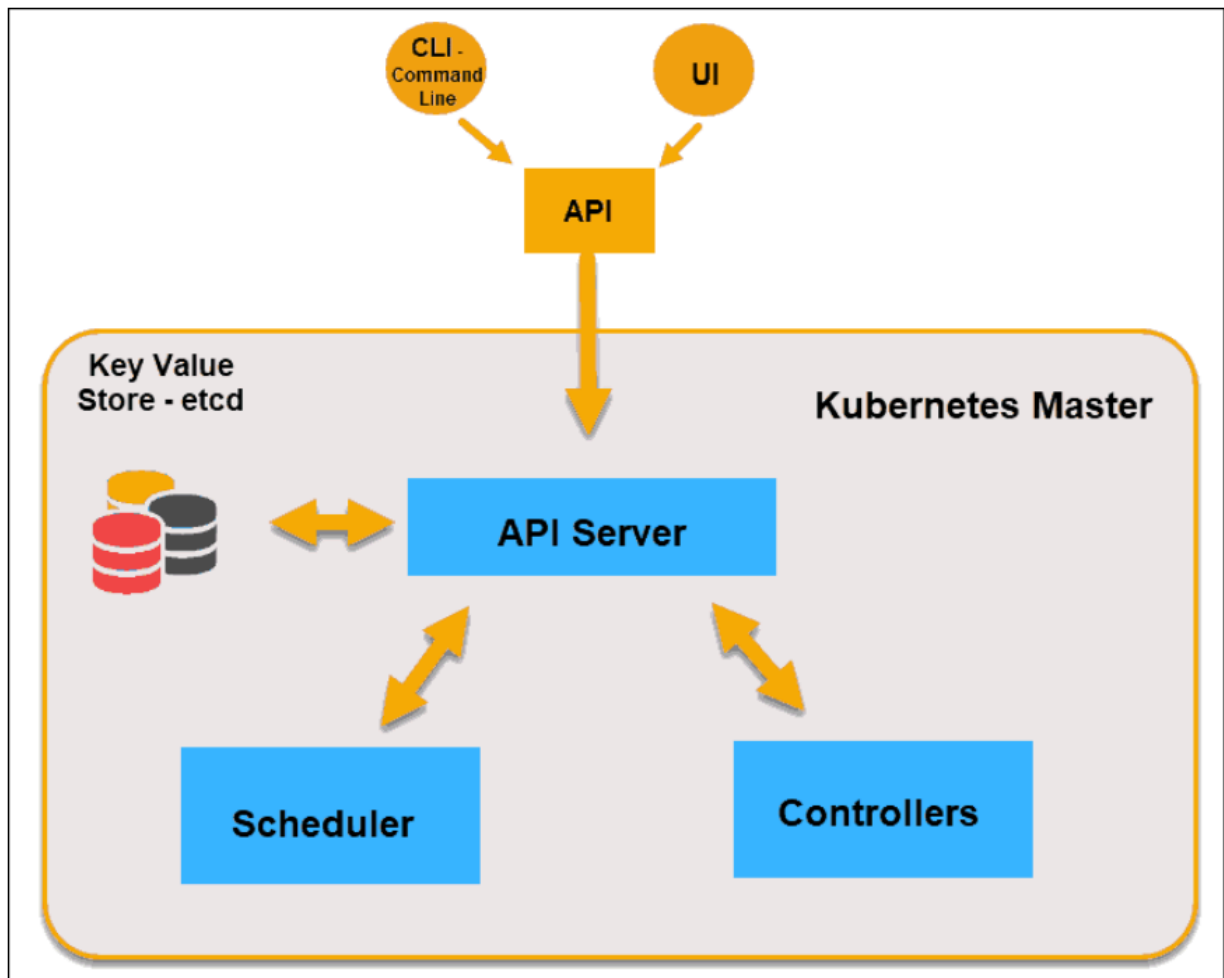
Controller

The role of the Controller is to obtain the desired state from the API Server. It checks the current state of the nodes it is tasked to control, and determines if there are any differences, and resolves them, if any.

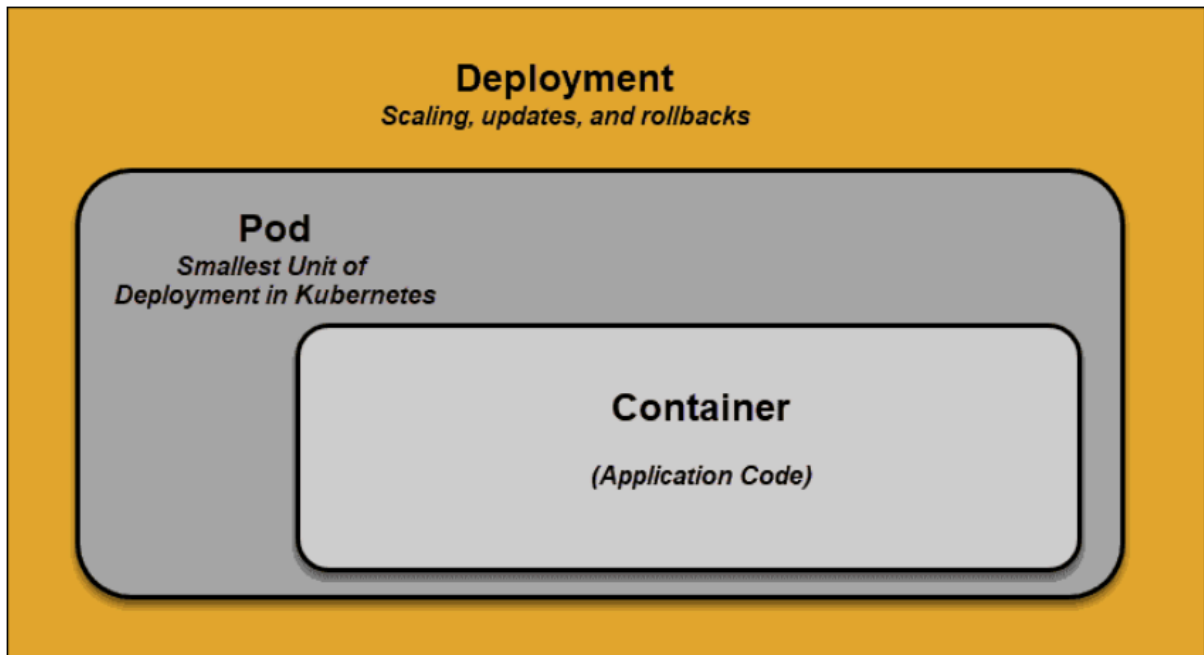
Scheduler

A Scheduler watches for new requests coming from the API Server and assigns them to healthy nodes. It ranks the quality of the nodes and deploys pods to the best-suited node. If there are no suitable nodes, the pods are put in a pending state until such a node appears.

Flow of API requests to Master node:



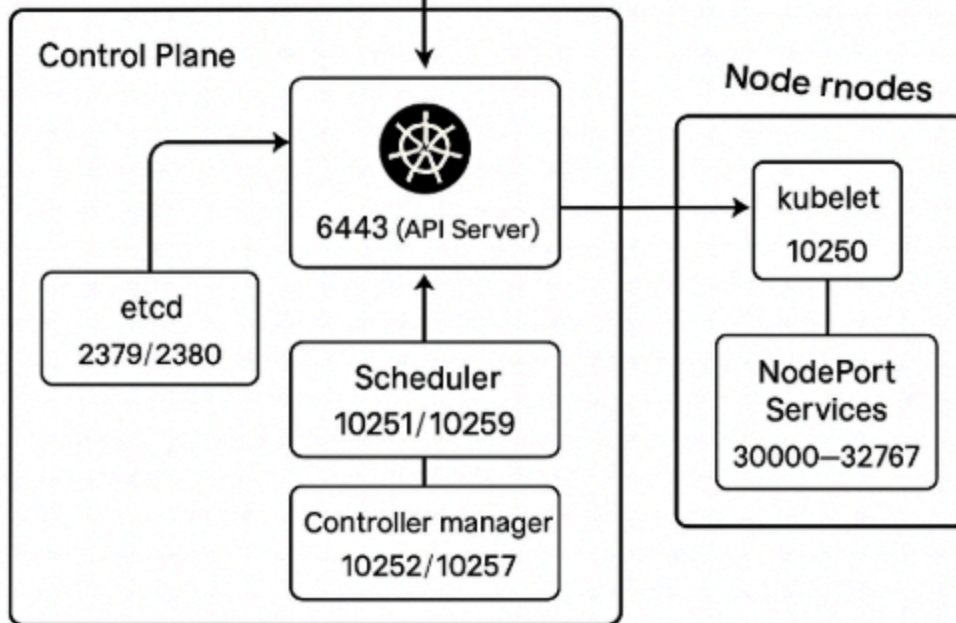
Deployments: abstract on top of PODS.



Network ports:

USER
Admin/Client

kubectl



Add-ons and Optional Components

