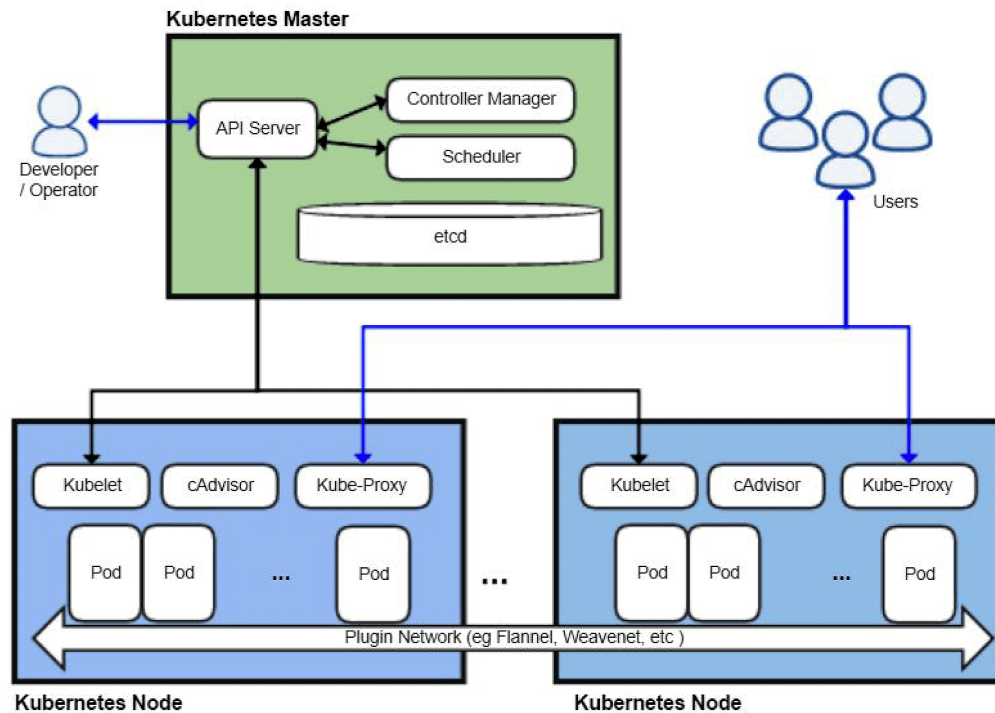


Kubernetes Architecture Diagram



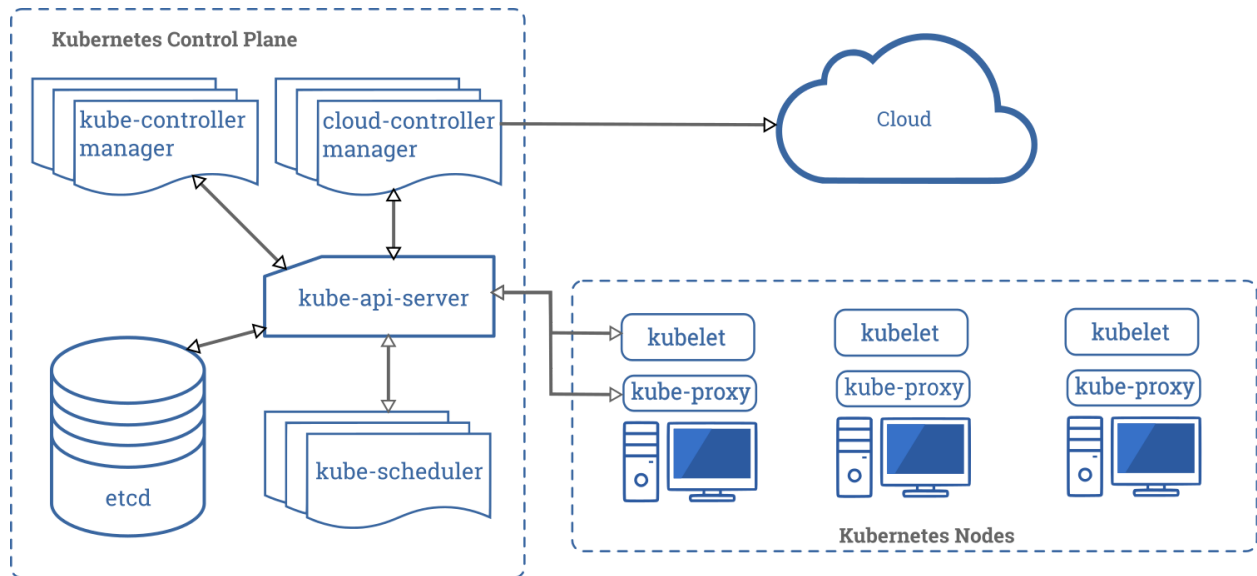
1) In the AWS Kubernetes architecture diagram above you can see, there is one or more master and multiple nodes. One or masters used to provide high-availability.

2) The Master node communicates with Worker nodes using Kube API-server to kubelet communication.

3) In the Worker node, there can be one or more pods and pods can contain one or more containers.

4) Containers can be deployed using the image also can be deployed externally by the user.

Kubernetes Architecture Components



Kubernetes Master Node

Master Node is a collection of components like Storage, Controller, Scheduler, API-server that makes up the control plan of the Kubernetes. When you interact with Kubernetes by using CLI you are communicating with the Kubernetes cluster's master node. All the processes run on a single node in the cluster, and this node is also referred to as the master.

Master Node Components:

- 1) Kube API-server** performs all the administrative tasks on the master node. A user sends the rest commands as YAML/JSON format to the API server, then it processes and executes them. The Kube API-server is the front end of the Kubernetes control plane.
- 2) etcd** is a distributed key-value store that is used to store the cluster state. Kubernetes stores the file in a database called the **etcd**. Besides storing the cluster state, etcd is also used to store the configuration details such as the subnets and the config maps.
- 3) Kube-scheduler** is used to schedule the work to different worker nodes. It also manages the new requests coming from the API Server and assigns them to healthy nodes.
- 4) Kube Controller Manager** task is to obtain the desired state from the API Server. If the desired state does not meet the current state of the object, then the corrective steps are taken by the control loop to bring the current state the same as the desired state.

There are different types of control manager in Kubernetes architecture:

- **Node Manager**, it manages the nodes. It creates new nodes if any node is unavailable or destroyed.
- **Replication Controller**, it manages if the desired number of containers is running in the replication group.
- **Endpoints controller**, it populates the endpoints object that is, joins Services & Pods.

Kubernetes Worker Node

The worker nodes in a cluster are the machines or physical servers that run your applications. The Kubernetes master controls each node. there are multiple nodes connected to the master node. On the node, there are multiple pods running and there are multiple containers running in pods.

Worker Node Components

1) Kubelet is an agent that runs on each worker node and communicates with the master node. It also makes sure that the containers which are part of the pods are always healthy. It watches for tasks sent from the API Server, executes the task like deploy or destroy the container, and then reports back to the Master.

2) Kube-proxy is used to communicate between the multiple worker nodes. It maintains network rules on nodes and also make sure there are necessary rules define on the worker node so the container can communicate to each in different nodes.

3) Kubernetes pod is a group of one or more containers that are deployed together on the same host. Pod is deployed with a shared storage/network, and a specification for how to run the containers. Containers can easily communicate with other containers in the same pod as though they were on the same machine.

4) Container Runtime is the software that is responsible for running containers. Kubernetes supports several container runtimes: Docker, containers.