Kubernetes Architecture:

A Kubernetes Cluster consists of a set of nodes, which may be physical servers, VMs, or cloud provisioned VMs.

Two types of nodes present in Kubernetes:

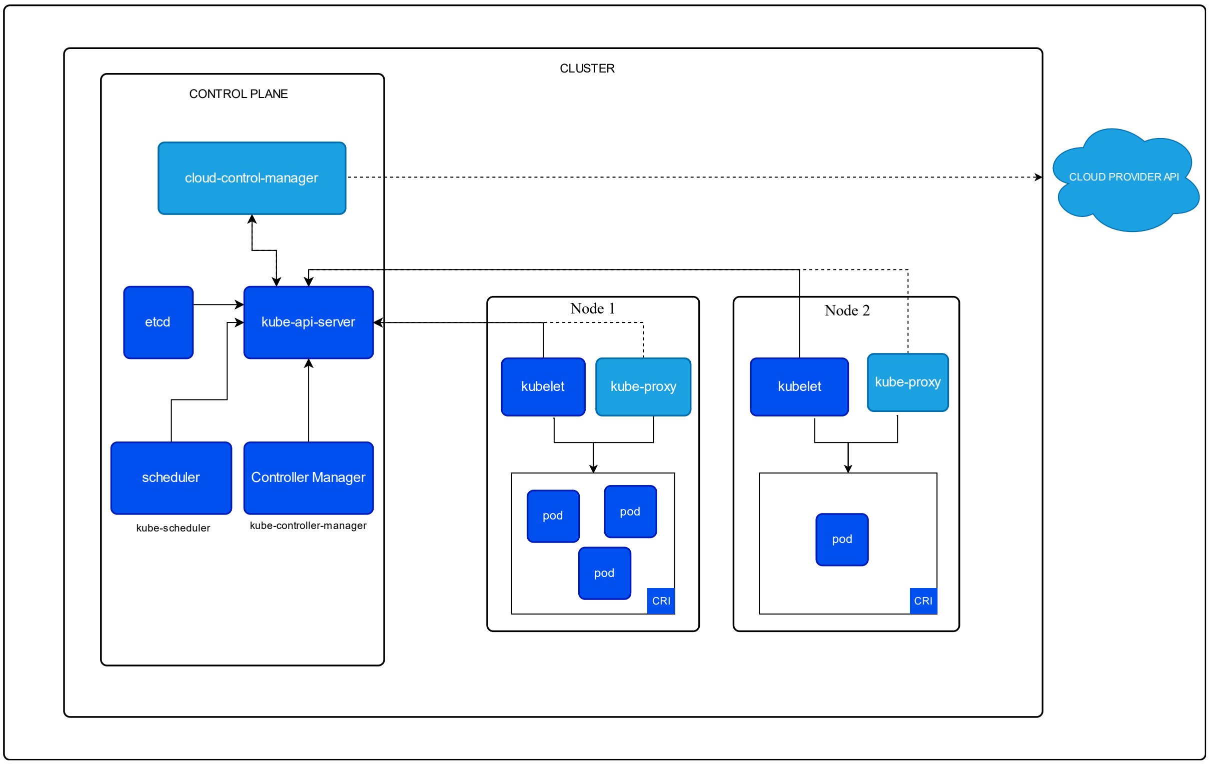
1. Control plane (Master Node): Responsible for managing the cluster, storing information regarding different nodes, planning which pod goes where, monitoring nodes and container workload on them, etc.
2. Worker nodes: Responsible for managing the workloads like running the application containers on them and communicating with master node.

Master Node Make this happen by the helps of Control plane components which are present in it, which are:

1. ETCD Cluster
2. Kube-Scheduler
3. Controller Manager
4. Kube-API Server
5. CRI (Container Runtime Engine) (containerD)

Components of Worker Nodes:

1. Kubelet
2. Kube-proxy
3. CRI



ETCD Cluster: It's a database that stores information about all cluster and worker node containers in a key-value pair.

Kube-Scheduler: It identifies right node to place the container based on container's resource requirements, nodes resource capacity or any declared constraints such as taints and toleration, etc.

Controller Manager: It contains various controller which takes care of specific segment of the cluster i.e. Node-controller will take care of nodes, Resplication-controller will take care of Replica Set, etc.

Kube-API Server: It's the primary management component of Kubernetes, It's responsible for exposing the Kubernetes API to external user to manage the Whole Kubernetes cluster as well as various controller to monitor various changes, and by the worker node to communicate with the server. It also periodically fetches information from Kubelet to continuously monitor the nodes and containers on them.

Kubelet: It's an agent that runs on each node of a cluster, It's responsible for taking instructions from Kube-API server and deploy and destroy containers on nodes as per instructions.

Kube-proxy: It ensures that necessary rules are in place so that it allows the containers running on them to reach each other and establish a communication network.

ETCD:

- ETCD is a distributed reliable key-value store that is simple, secure, and Fast.
- To use ETCD separately we can download the binary and extract it and use it.
- In Kubernetes ETCD Datastore stores information regarding the cluster like Nodes, pods, configs, secrets, accounts, roles, bindings, others.
- Every information we see using "kubectl get" command is from ETCD cluster.
- If we setup our KubeAdm cluster from scratch, then we need to manually download the ETCD as well.
- It runs on every master node, that's why when a High availability Kubernetes cluster is deployed their multiple master nodes present so multiple ETCD DB is also present, in that case we have to setup in ETCD configuration file.

Kube API Server:

- It's responsible for validation of API objects, frontend to the cluster's shared state, rest operations, authentication and authorization of the Cluster.
- When we run kubectl get command, kube-apiserver first authenticate the request, it then retrieves the data from etcd cluster back with the requested information.
- Just like that when we try to create a pod, the request is authenticated first then apiserver first create a pod object without assigning it to any nodes and updates the etcd.
- Then the scheduler get the information from etc, apiserver and schedule a pod on suitable node.
- Then the apiserver passes the information it got from scheduler to the Kubelet of respective node. The Kubelet then creates the pod by instructing the CRI to run the image. Then update the pod state back to apiserver. then api server updates the etcd.
- Kube-apiserver is the only component of a cluster that directly interacts with etcd.

Kube controller manager:

- A Controller is a process inside the master which continuously monitor states of various components. And works towards bringing the whole cluster to desired functioning state.
- Example the node controller in the controller manager checks the status of every nodes every 5secs. By this it monitors states of all nodes.
- Example Replication controller is responsible for maintaining the status of replica sets. So that desired number of pods are available for all times.
- These two are just two controller among many more controller present in controller manager
- Some other controllers are : Deployment controller, Namespace controller, endpoints controller, service controller, jobs controller, pv protection controller, etc.

Kube Scheduler:

- The scheduler is only responsible for deciding which pod goes to which node. It doesn’t place or run the pod on the node, that’s the job of Kubelet.
- It decides it based on certain criteria i.e.
- The scheduler looks at each pod and tries to find the best node for it. i.e. it looks at the resource requirements, limits, taints and tolerations, etc.

Kubelet:

- The Kubelet on each node of a Kubernetes cluster registers the node to the cluster.
- It requests the CRI to pull required image and run an instance.
- Monitors node and pods.

Kube Proxy:

- It’s a process that runs on each node