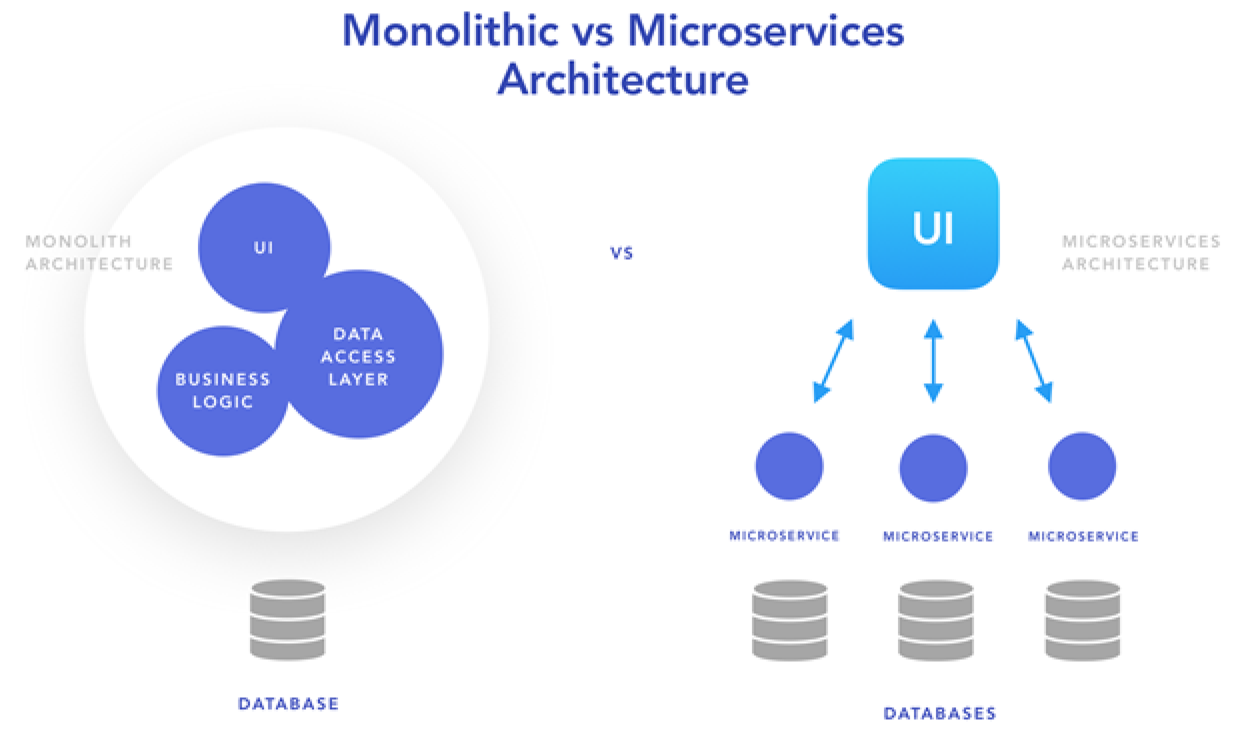
Kubernetes Essential

Agenda

- Trend from Monolithic to Microservices
- What is Kubernetes?
- Kubernetes architecture

Trend from Monolithic to Microservices



Kubernetes Overview

Kubernetes or K8s is an opensource platform for managing containerized workloads and services with rapidly growing ecosystem as well as wide usage of Kubernetes services and tools.

Kubernetes and Microservices

Kubernetes provides a powerful platform for deploying and managing microservices:

- Load balancing
- Scaling
- Resilience
- Resource management
- Deployment management

These features make it easier to build, deploy, and manage microservices at scale, allowing you to deliver more reliable and efficient applications.

Cloud Orchestration Solutions

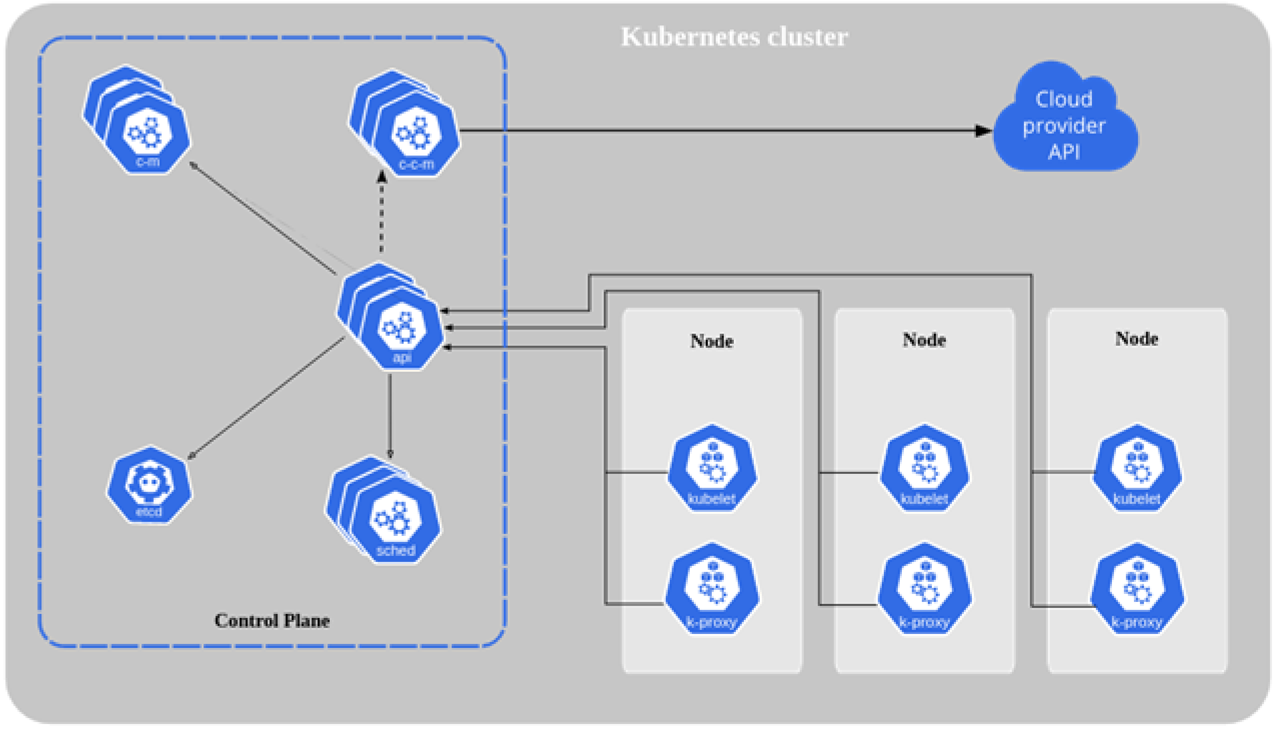
Cloud providers such as Amazon Web Services, Microsoft Azure, and Google Cloud the platform also offers built-in container orchestration solutions, including cloud-native Kubernetes implementations!

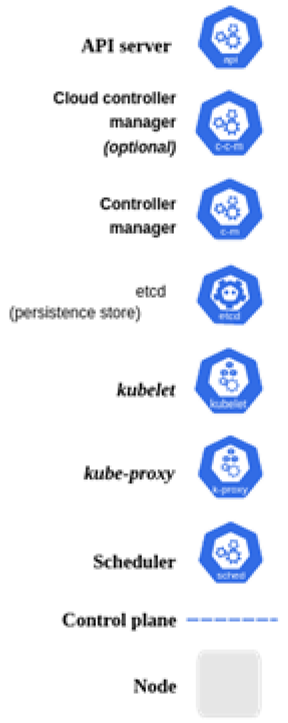
Amazon EKS

Azure Kubernetes Service (AKS)

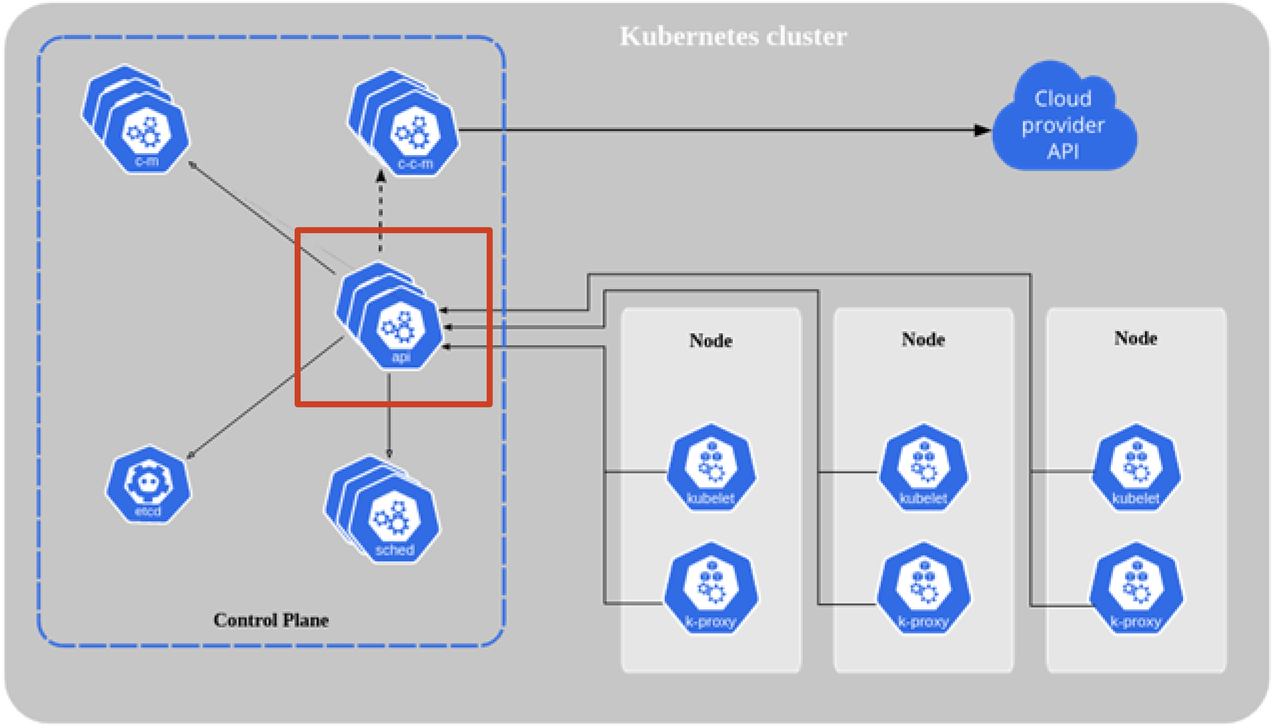
Google Kubernetes Engine

Cluster Architecture





Control Plane Components

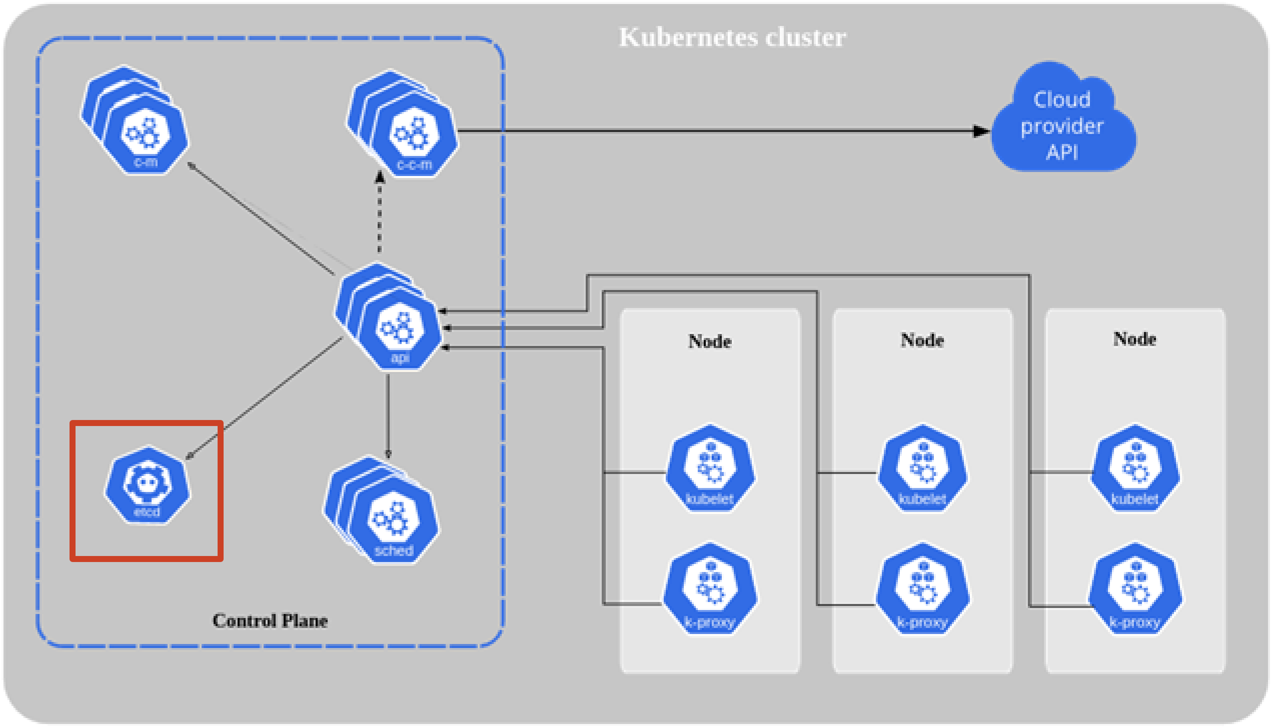


kube-api-server:

- Exposes the Kubernetes API to external users, authenticate and validate the requests from outside.

- Acts as the frontend to the cluster's shared state through which all other components interact.

ETCD

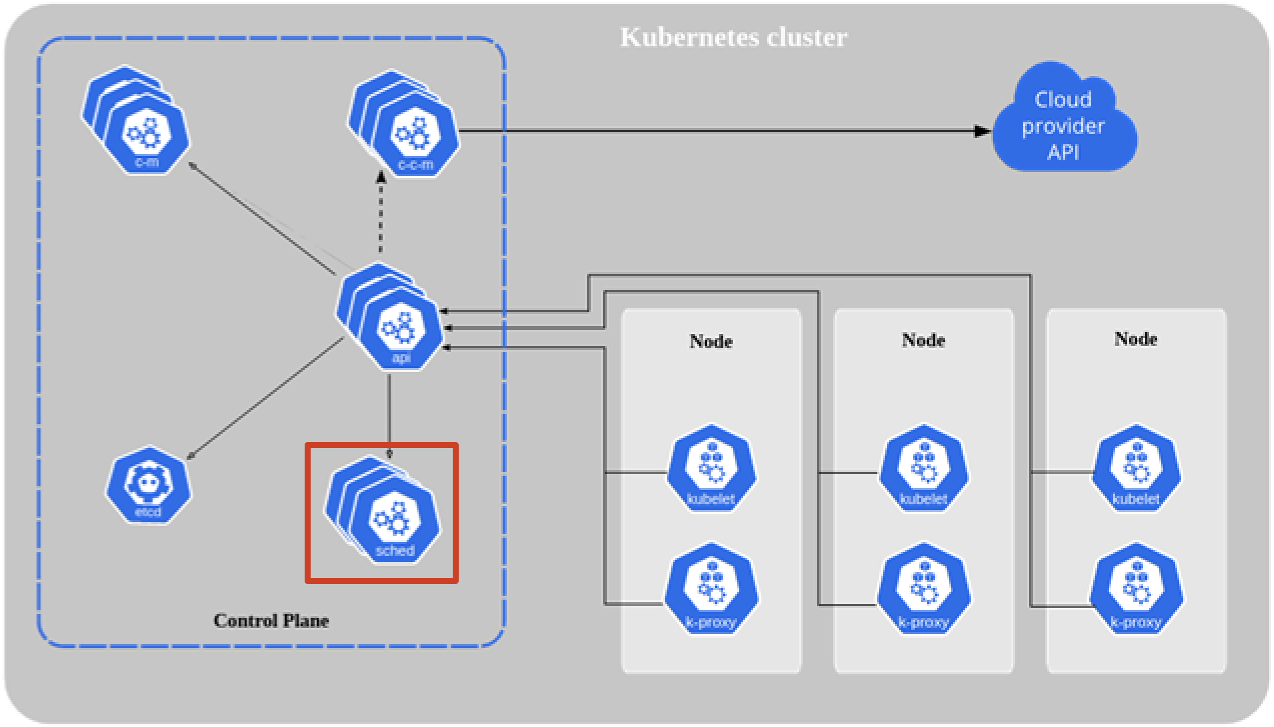


etcd-server:

- Key-value store used as Kubernetes' backing store for all cluster data.

- Follow a consensus algorithm to ensure cluster's state is fault-tolerance.

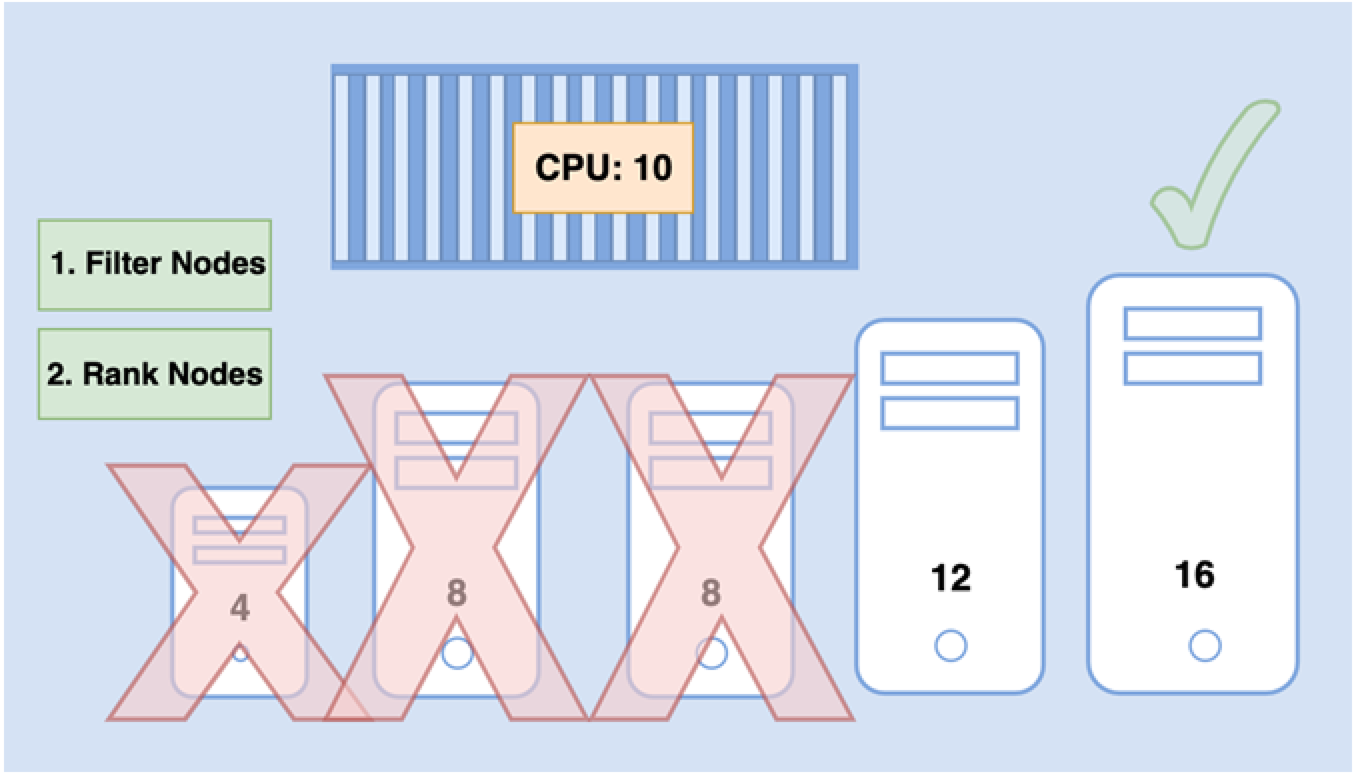
KUBE-SCHEDULER



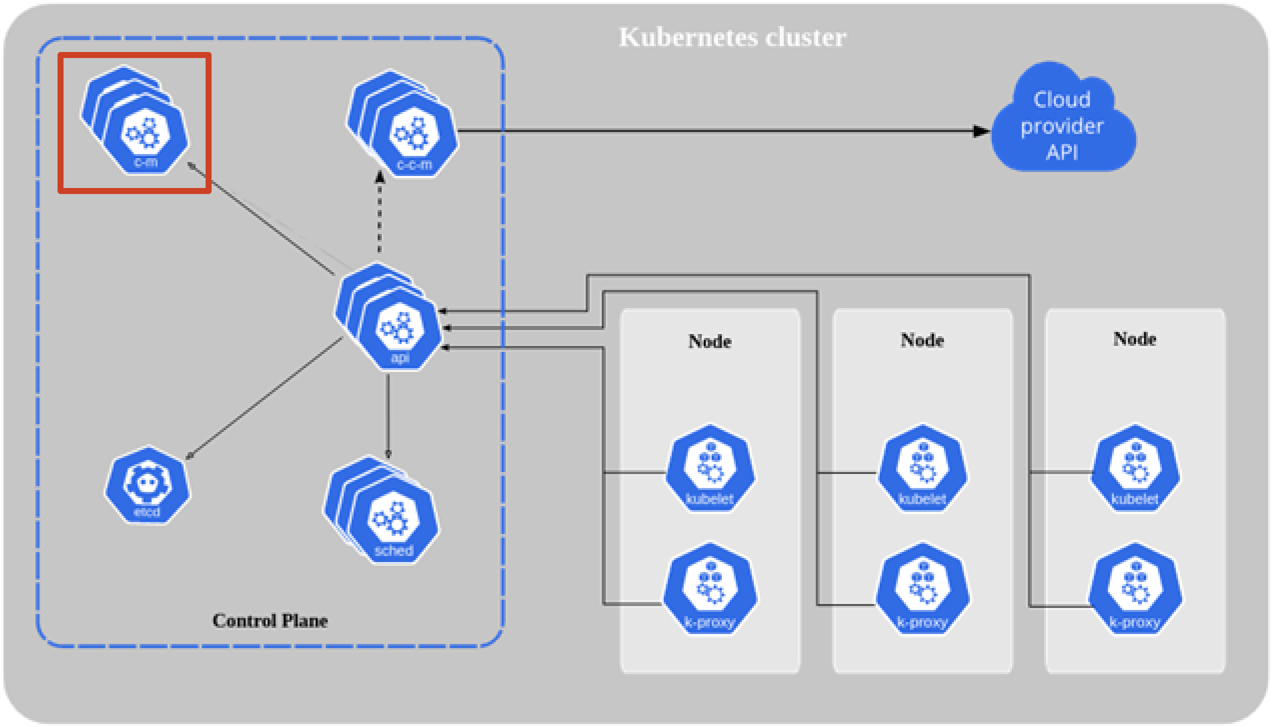
kube-scheduler-server:

- Chooses node for created Pods based on several factors. E.g: Resource constraints, policies, …

KUBE-SCHEDULER



KUBE-CONTROLLER-MANAGER

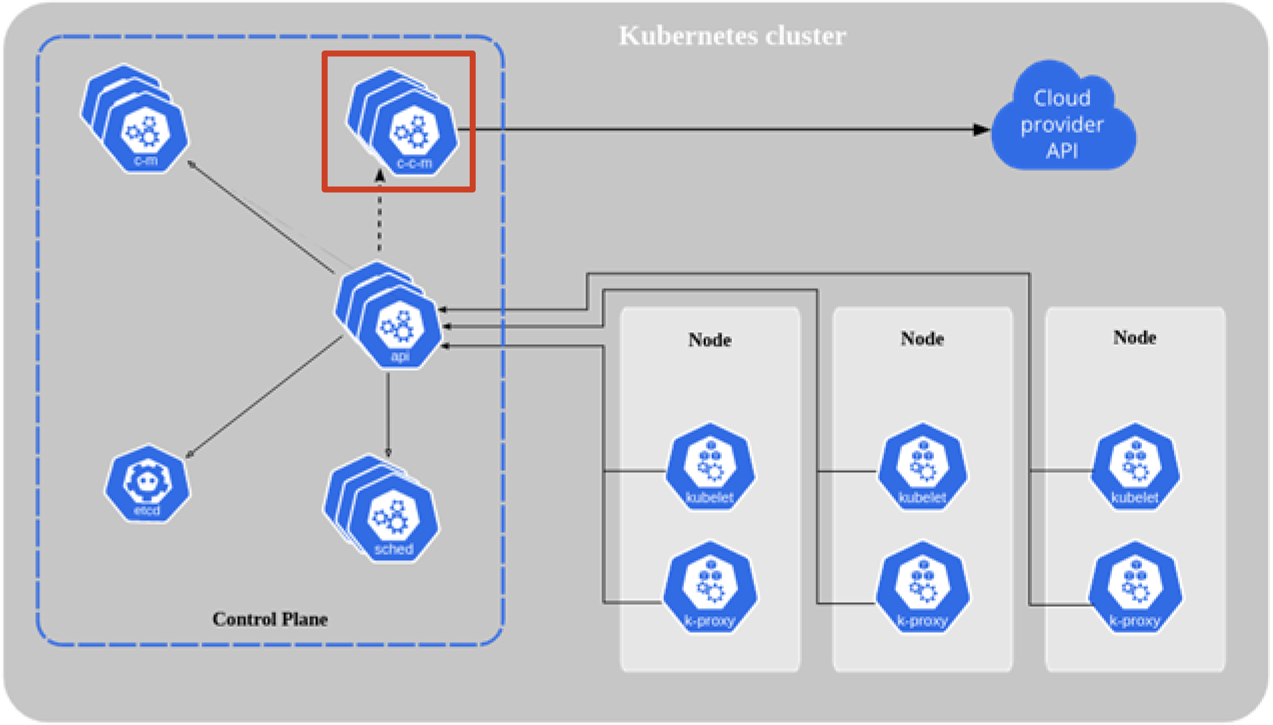


kube-controller-manager:

- Serves as the primary daemon that manages all core component control loops.

- Monitors the cluster state via the **kube-api-server** and steers the cluster towards the desired state.

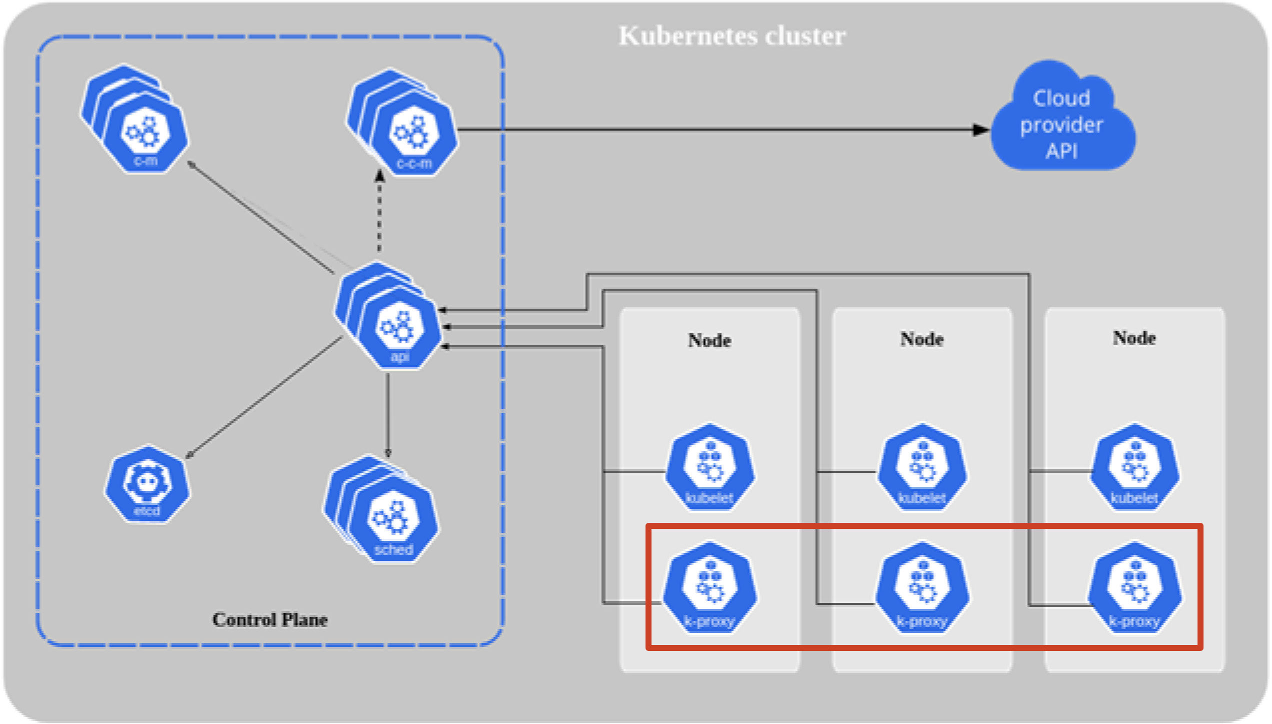
CLOUD-CONTROLLER-MANAGER



cloud-controller-manager:

- Daemon that provides cloud-provider specific knowledge and integration capability into the core control loop of Kubernetes.

KUBE-PROXY

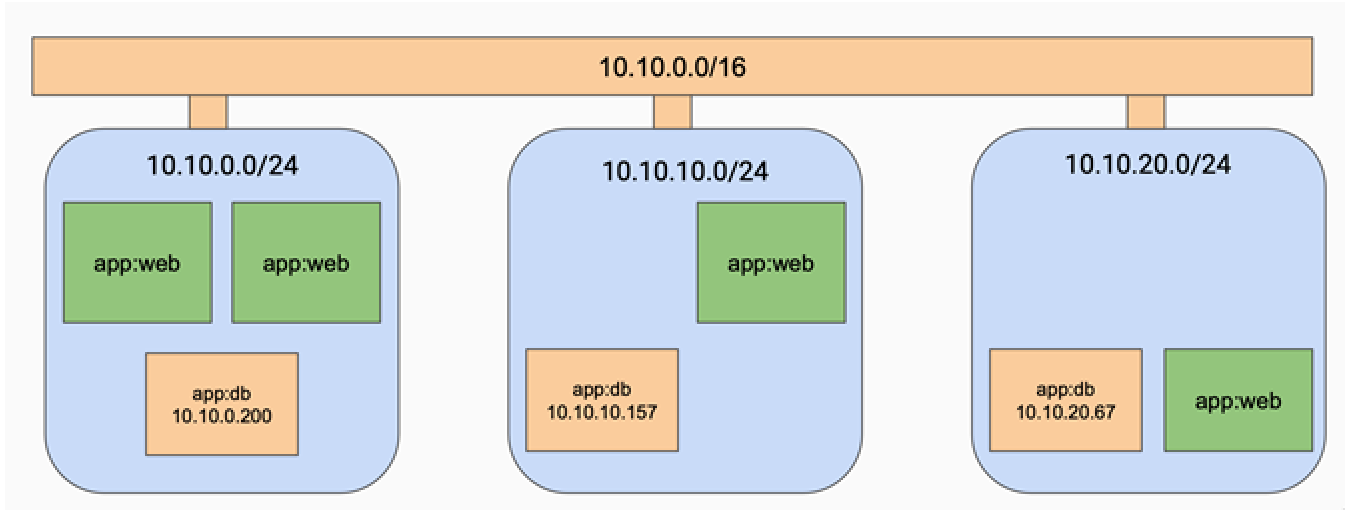


kube-proxy:

- Runs on each node in the cluster, implementing part of the Kubernetes Service concept, allow network communication to your Pods from network sessions inside or outside of your cluster.

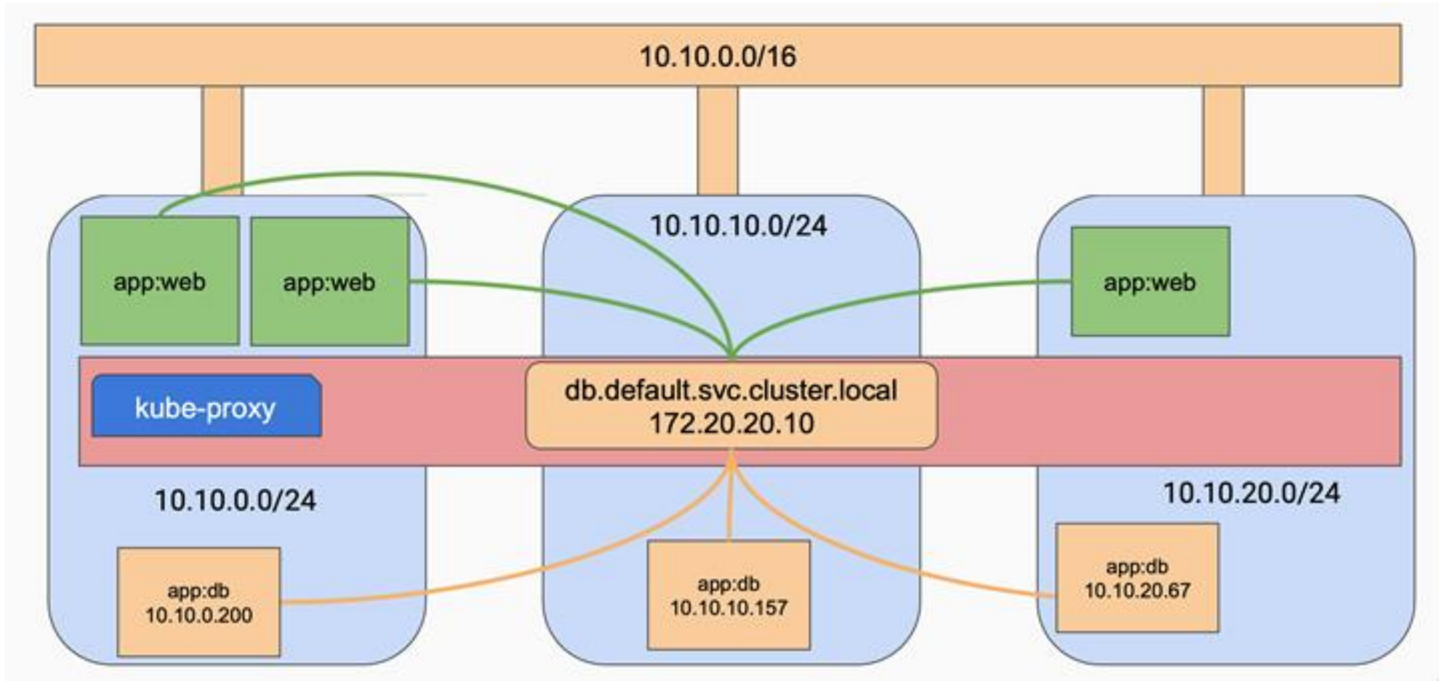
KUBE-PROXY

- How **web service** connect to **database service**?
- How to keep track of **database service** ip addresses in case of ip changing?
- How to do load-balance between many service instances?



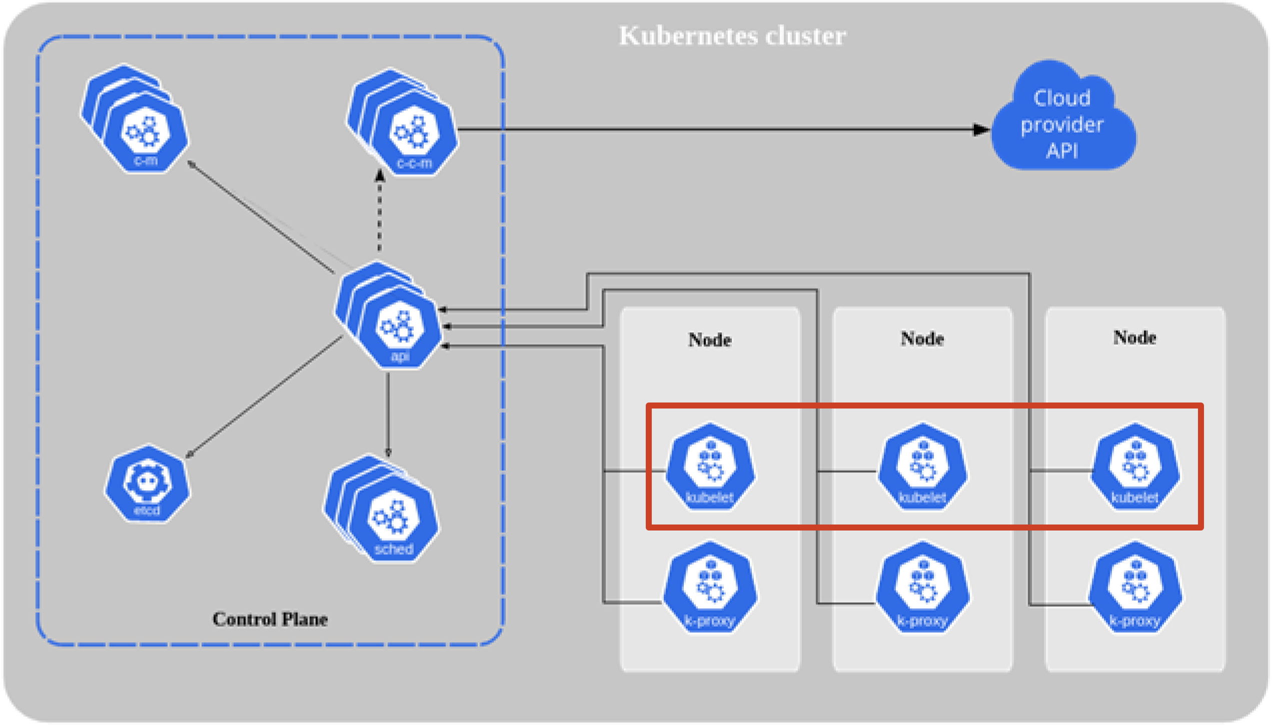
worke

KUBE-PROXY



worke

KUBELET



kubelet:

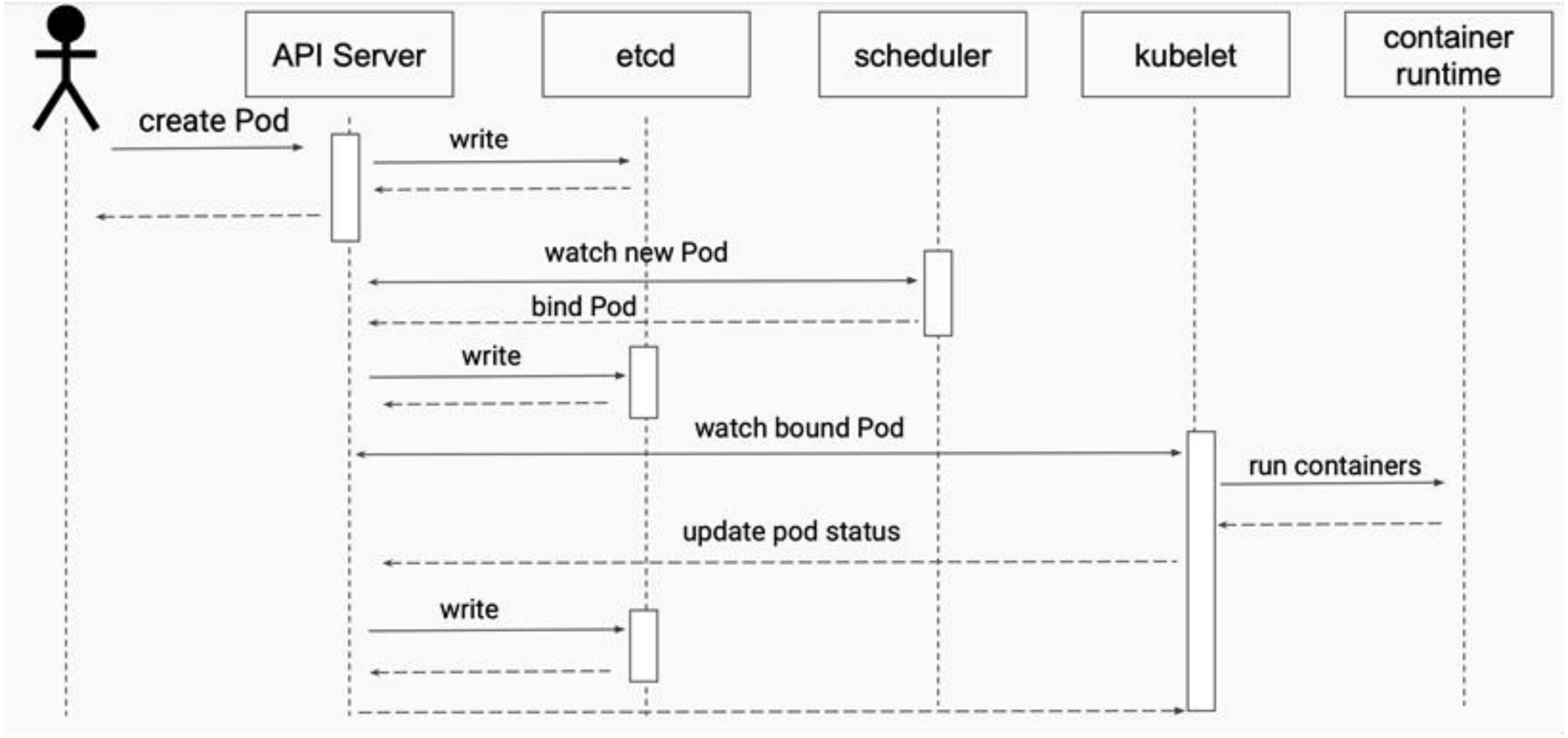
- Runs on each node in the cluster. It makes sure that containers are running in a pod.

- Takes a set of PodSpecs that are provided through various mechanisms and ensures that the containers described in those PodSpecs are running and healthy.

CONTAINER RUNTIME ENGINE

- Docker is not the only option for doing containers!
- rkt – Created by CoreOS, “designed with composability and security in mind.”
- Containerd – Emphasizes “simplicity, robustness, and portability.”
- LXC/LXD

Request Flow



Q&A