# Kubernetes

**Kubernetes** adalah *open-source platform* untuk mengotomatisasi **deployment**, **scaling**, dan **management** dari aplikasi yang berjalan di dalam **container**. Selain mengelola container secara otomatis, Kubernetes juga bisa melakukan **horizontal scaling** terhadap container (menambah atau mengurangi jumlah container secara otomatis berdasarkan kebutuhan).

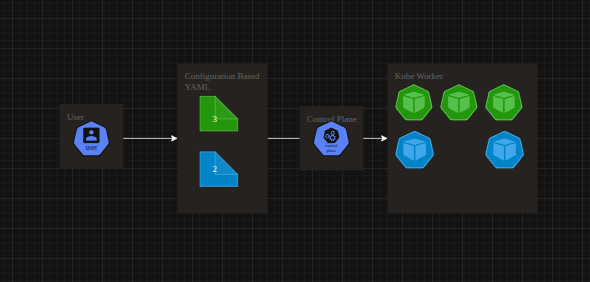
## Sejarah

Kubernetes awalnya dikembangkan oleh **Google**. Sebelum membuat Kubernetes, Google memiliki sistem internal untuk mengelola container bernama **Borg**, dan versi berikutnya bernama **Omega**.

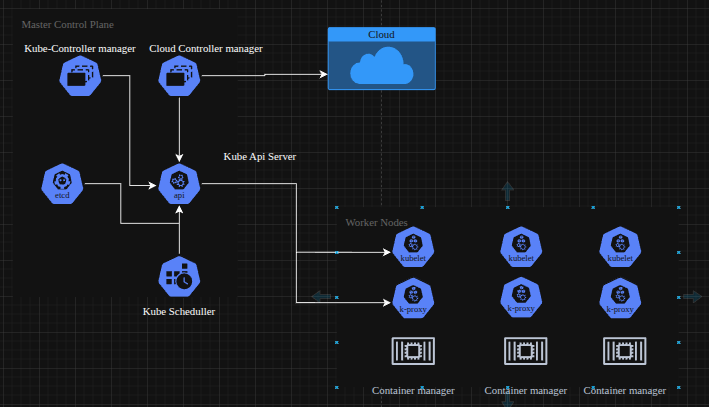
Pengalaman dari mengembangkan dan menjalankan Borg dan Omega selama bertahun-tahun menjadi dasar pembuatan Kubernetes.

Pada tahun **2014**, Google merilis Kubernetes sebagai proyek **open-source**, dan sejak itu proyek ini dikembangkan secara kolaboratif oleh komunitas, termasuk melalui **Cloud Native Computing Foundation (CNCF)**. Kubernetes itu dibuat Menggunakan **Golang**

## Alur Kerja



## Arsitektur Kubernetes



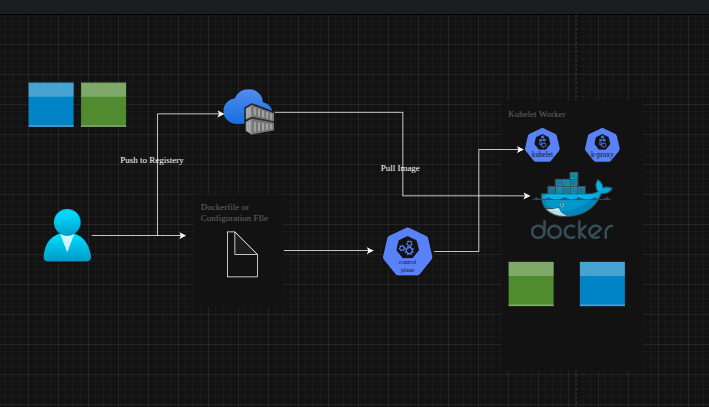
### Kubernetes Master

* Kube APi Server Digunakan Untuk Berinteraksi dengan Cluster Kubernetes
* ETCD digunakan untuk menjadi database dari kubernetes
* Kube-Scheduller digunakan untuk memperhatikan Aplikasi yang kita minta agar bisa di kirim ke node atau dijalankan Node
* Kube-Controller Manager digunakan untuk mengatur Terhadap Kubernetes Cluster
* Cloud Controller Manager digunakan untuk mengatur Terhadap Kubernetes Cluster DI Cloud Provider

### Kubernetes Node Worker

* Kubelet Ada di Setiap Worker Node Biasanya DIgunakan Untuk Memastikan Status Aplikasi Developer Bisa berjalan atau tidak
* Kube Proxxy di Setiap Worker Node ada Digunakan Untuk mengatur Jaringan atau yang bisa akses Aplikkasi Developer Bisa Sebagai Load Balancer
* Container Manager ada di setiap Worker Node biasanya untuk container Manager beberapa Container Manager yang support diantaranya Docker, containerD,cri-o

## Detail Kerja Kubernetes



# Cara Menginstall Kubernetes di Local

* Menggunakan Docker Deskop ( Windows)
* Menggunakan Minikube (mac, Linux)
  + <https://github.com/kubernetes/minikube/releases>
    - Virtualbox https://medium.com/@dsharma182/install-minikube-on-windows-10-home-edition-with-virtualbox-48d1b33909f3
    - Docker

<https://medium.com/@rajeshmamuddu/setup-minikube-with-docker-driver-539c151b1b7e>

## Menginsatall Api Server Kube-api

* Kubectl

<https://kubernetes.io/id/docs/tasks/tools/install-kubectl/>

* kubectx

<https://kubernetes.io/id/docs/tasks/tools/install-kubectl/>

dengan menggunakan kubectx kita bisa berpindah namespace bahkan cluster dengan simpel

# Node

Node adalah worker machine di kubernetes, sebelumnya ada yang menyebutnya minion

Node bisa bisa berbentuk VM atau Server

di dalam Node biasanya ada kube-proxxy, Kubeler, dan Container Manager

command for Node

$kubectl get node

melihat semua node

$kubectl describe node <nama-node>

melihat detail node

# POD

pod adalah kumpulan Container yang ada di kubernetes

Command For Pod

$kubectl get pod

melihat pods

$kubectl describe pod <nama-pod>

melihat detail pods

$kubectl port-forward <nama-pod> <Port-akses>:<port-pod>

untuk melihat akses port nya

# PARAMETER DUMP FILE Configuration

### 

### ImagePullPolicy:

Option

**Always**

Kubernetes akan **selalu menarik (pull) image yang baru dari registry setiap kali Pod dibuat**, **meskipun image dengan tag yang sama sudah ada di node**.

**IfNotPresent**

Kubernetes **hanya akan menarik image dari registry jika image tersebut belum ada di node lokal**.

### progressDeadlineSeconds:

adalah parameter yang digunakan dalam manifest **Deployment** di Kubernetes (bukan pada semua resource), dan berfungsi untuk menentukan **berapa lama (dalam detik)** Kubernetes akan menunggu hingga Deployment menunjukkan **kemajuan (progress)** dalam melakukan **update** sebelum dianggap **gagal** (Deployment dianggap "failed").

Option

600

# COMMAND

**$kubectl config current-context**

untuk melihat cluster yang sedang kita berada sekarang

$kubectl create -f nama-file.yaml

melakukan kube manifest menggunakan file yaml

### Alias Setup

alias k='kubectl'

echo "alias k='kubectl'" >> ~/.bashrc

source ~/.bashrc

echo "alias k='kubectl'" >> ~/.zshrc

source ~/.zshrc

