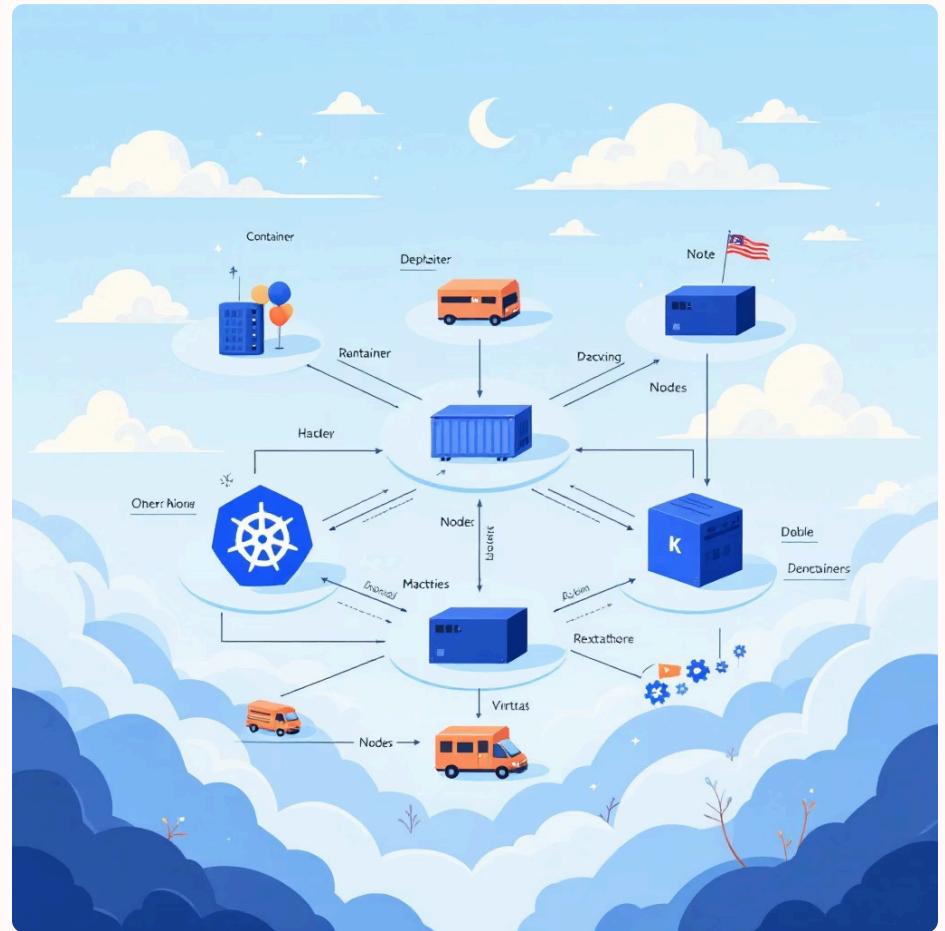


Bugün Lab'da Ne Yapacağız? (VM Üzerinde Minikube)

Benim hedefim şu: Herkesin VM'inde çalışan bir Kubernetes ortamı olsun ve en az 1 uygulamayı deploy edelim.



Plan



Docker kur

Minikube'un kullanacağı konteyner çalışma zamanı olan Docker'ı kuruyoruz.



kubectl + minikube kur

Kubernetes komut satırı aracı olan kubectl'i ve yerel Kubernetes ortamı Minikube'u yükliyoruz.



örnek deploy

Kümenin çalışır durumda olduğunu doğrulamak için örnek bir Nginx uygulamasını dağıtıyoruz.

erişim

Uygulamamızı dışarıya açarak bir web tarayıcısı üzerinden erişim sağlıyoruz.

toparla

Gerçekleştirdiğimiz adımları gözden geçiriyor ve lab ortamını sonlandırıyoruz.

Doğrula

Tüm bileşenlerin doğru bir şekilde kurulduğundan ve çalıştığından emin olmak için kontroller yapıyoruz.



Ön Koşullar ve Hızlı Kontrol

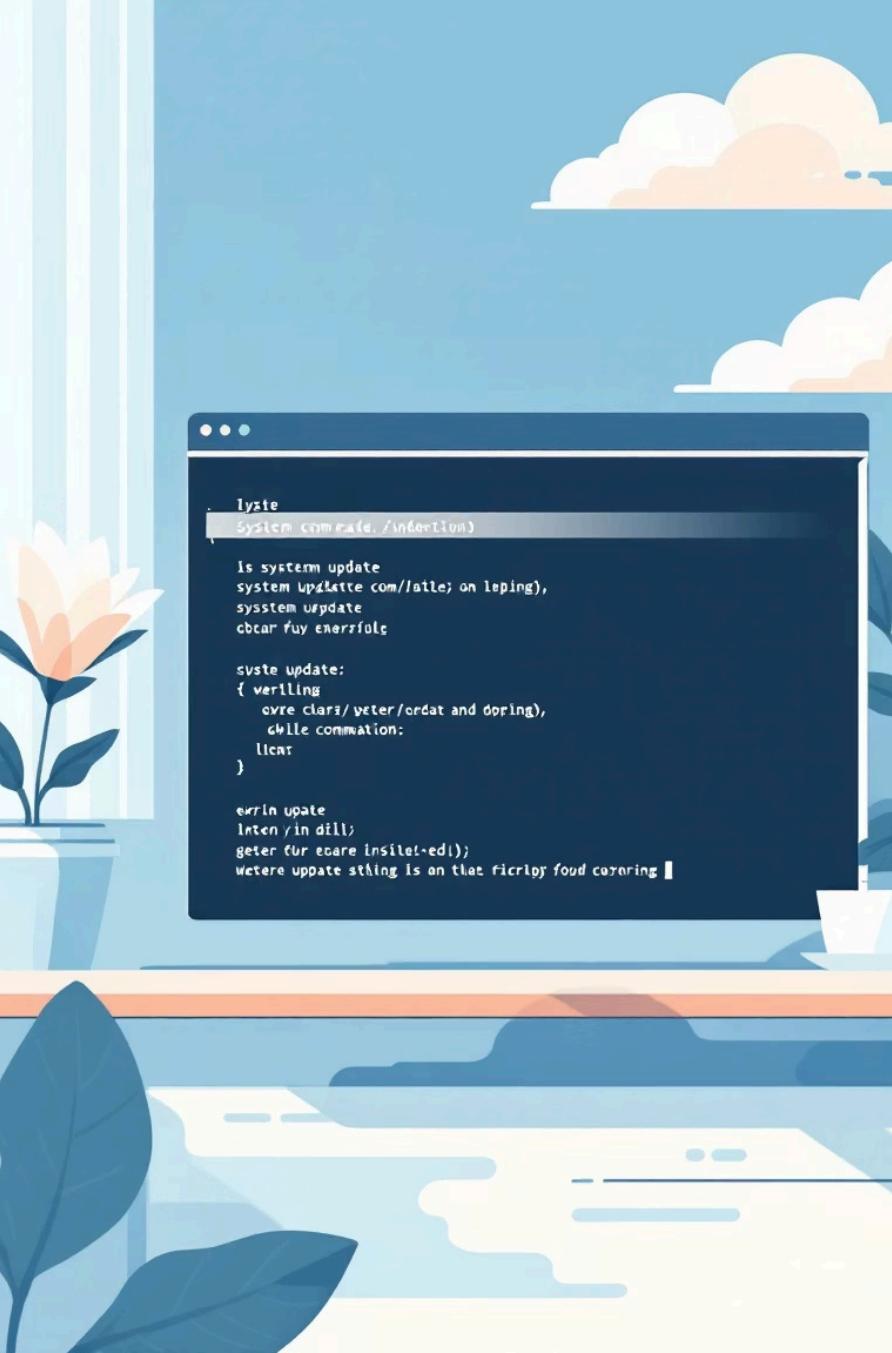
VM önerisi (minimum):

- 2 vCPU (tercihen 4)
- 4 GB RAM (tercihen 8)
- 20 GB disk
- Ubuntu 22.04/24.04

Hızlı kontrol (VM içinde):

```
lsb_release -a  
uname -a  
free -h  
df -h
```

❑ **Not:** VM içindeyiz; en sorunsuz yol minikube + docker driver.



```
lzyte
System commands: /sbin/lzytione

ls system update
system update com//alte; on leping),
sysstem update
cbcari fuy enersfule

syste update:
{ veriling
  ovre clara/yeter/ordat and öpring),
  chile commination:
  licar
}

errin update
Inten yin dill;
geter fur esare insitel-ed);
Wetere update stting is on the ficerly food carning |
```

Sistem Güncelle + Temel Paketler

```
sudo apt update && sudo apt -y upgrade
```

```
sudo apt -y install curl wget ca-certificates gnupg lsb-release apt-transport-https
```

Docker Kurulumu (Ubuntu)

Docker repo anahtarı

```
sudo install -m 0755 -d /etc/apt/keyrings  
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg  
sudo chmod a+r /etc/apt/keyrings/docker.gpg
```

Docker repo ekle

```
echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu  
$(. /etc/os-release && echo ${VERSION_CODENAME}) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null  
  
sudo apt update  
sudo apt -y install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

Docker'ı aktif et

```
sudo systemctl enable --now docker
```

Kullanıcıyı docker grubuna ekle (sonra tekrar login gereklidir)

```
sudo usermod -aG docker $USER  
newgrp docker  
docker version  
docker ps
```



kubectl Kurulumu

Ben burada hızlı ve stabil olsun diye Ubuntu'nun paketini kullanacağım:

```
sudo apt update  
sudo apt -y install kubectl  
kubectl version --client --output=yaml
```



Minikube Kurulumu

```
curl -LO  
https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64  
sudo install minikube-linux-amd64 /usr/local/bin/minikube  
minikube version
```

Minikube Start (Docker Driver ile)

Burada işi garantiye almak için driver'ı açık açık docker seçiyorum.

```
minikube start --driver=docker --cpus=2 --memory=4096  
minikube status  
kubectl get nodes -o wide  
kubectl get pods -A
```

İsteğe bağlı (kullanışlı addon'lar):

```
minikube addons enable metrics-server  
minikube addons enable ingress  
minikube addons list
```



İlk Deploy: "Çalışıyor mu?" Testi

```
kubectl create deployment web --image=nginx  
kubectl get deploy,rs,pods -o wide
```

```
kubectl expose deployment web --type=NodePort --port=80  
kubectl get svc web -o wide
```

```
minikube service web --url
```

Çıkan URL'yi tarayıcıdan açıyoruz. Nginx ekranını
gördüysek tamamdır.