|  |
| --- |
| **LAPORAN PRAKTIKUM VIRTUALISASI KOMPUTER**  **INSTALASI MINIKUBE DAN SETUP KUBERNETES CLUSTER** |
|  |
| **Agus Pranata Marpaung**  **13323033**  **DIII TEKNOLOGI KOMPUTER** |
| **INSTITUT TEKNOLOGI DEL**  **FAKULTAS VOKASI** |

**Judul Praktikum**

|  |  |  |
| --- | --- | --- |
| **Minggu/Sesi** | : | XII/3 |
| **Kode Mata Kuliah** | : | 4332103 |
| **Nama Mata Kuliah** | : | VIRTUALISASI KOMPUTER |
| **Setoran** | : | Jawaban dalam bentuk *softcopy* |
| **Batas Waktu Setoran** | : | *Senin, 25 November jam 21:30* |
| **Tujuan** | : | 1. Mahasiswa mampu melakukan instalasi dan pembuatan cluster Kubernetes menggunakan Minikube pada sistem operasi Ubuntu. |

**Petunjuk**

# Teori

1. **Apa itu Kubernetes?**



(Sumber: [kubernetes.io)](https://kubernetes.io/)

**Kubernetes** adalah *system orchestration container*. Dengan kubernetes, Anda bisa membuat container di server yang berbeda, baik fisik maupun virtual, dan semua itu dilakukan secara otomatis. Kubernetes menangani distribusi beban di beberapa server yang memungkinkan Anda untuk menggunakan *resource* secara efisien dan menghindari penggunaan yang kurang atau berlebihan *resource*. Selain itu, Kubernetes juga menangani pemantauan dan pemeriksaan container. Dan jika ada container yang gagal karena alasan tertentu, Kubernetes dapat secara otomatis mengganti container yang gagal.

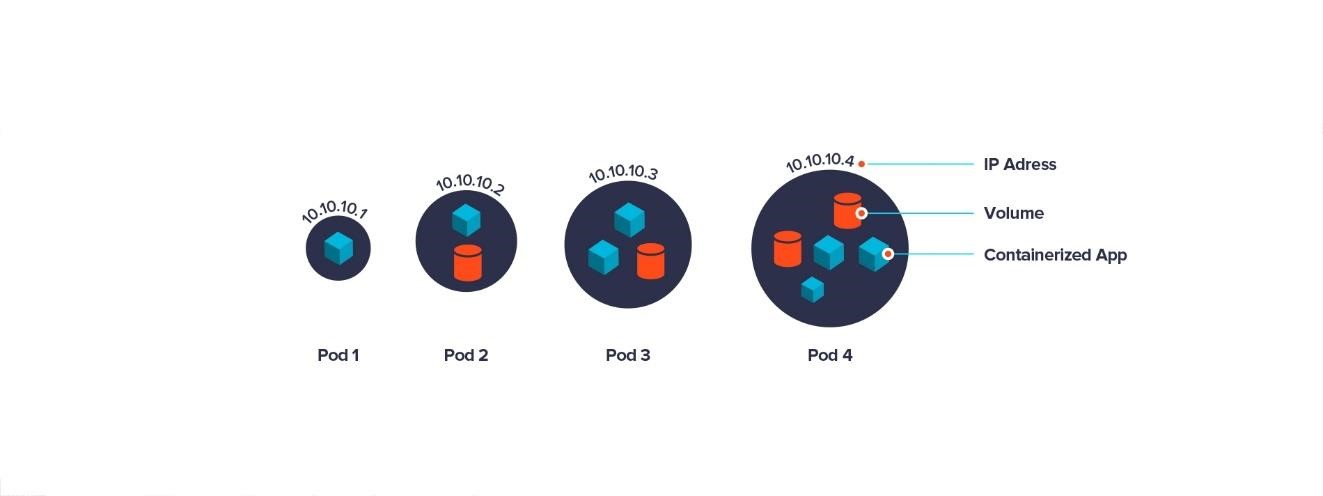
1. **Apa itu Minikube?**



(Sumber: [alxibra.medium.com)](https://alxibra.medium.com/menggunakan-local-image-di-minikube-fb5c027a76c1)

**Minikube** adalah sebuah *tools* yang bersifat open source yang memungkinkan Anda menjalankan Kubernetes cluster secara local pada computer/laptop Anda. Minikube dirancang untuk memfasilitasi pengembangan dan pengujian di lingkungan local.

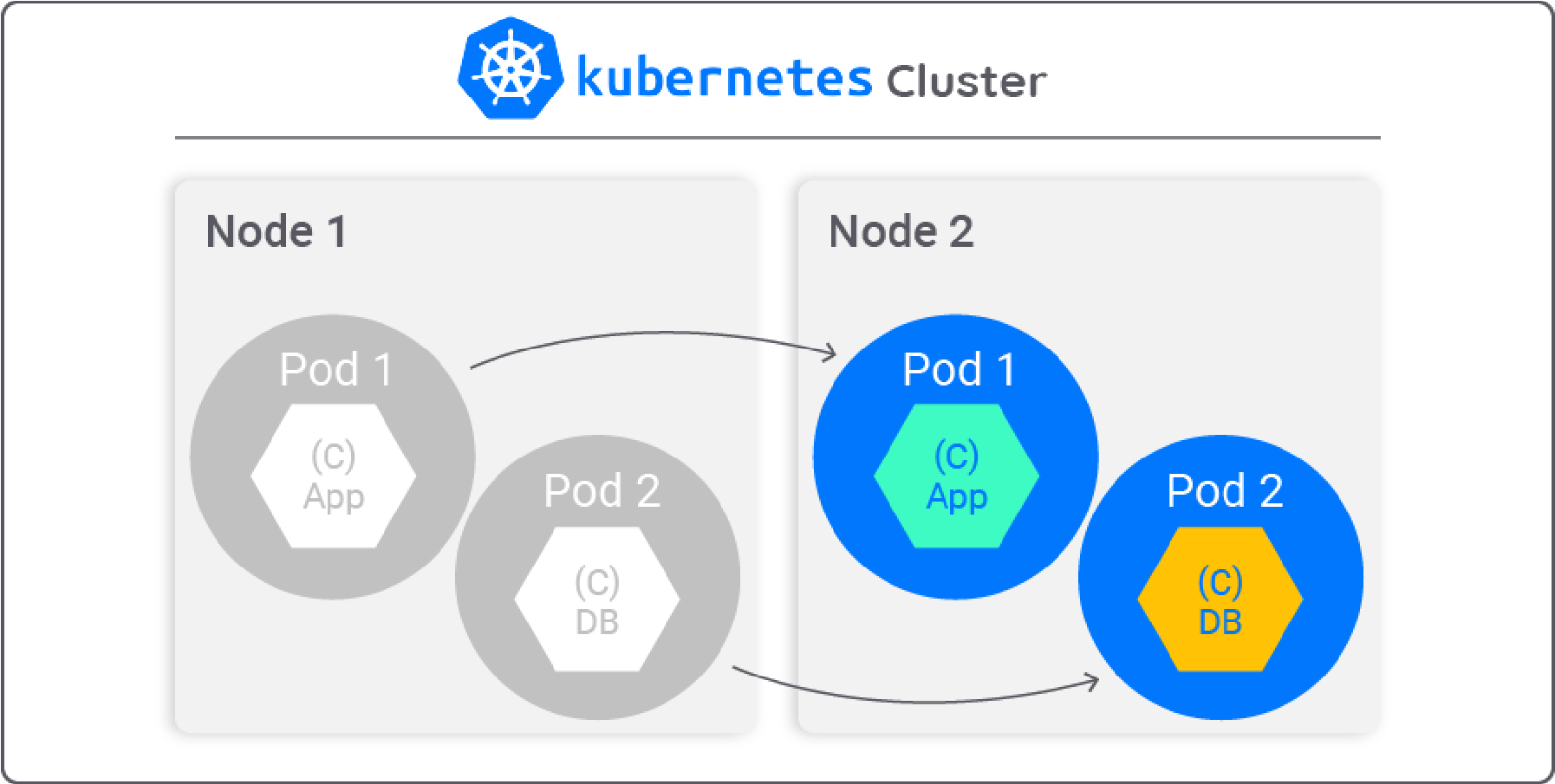
1. **Apa itu Pods?**



(Sumber[: www.weave.works)](https://www.weave.works/blog/kubernetes-node-everything-you-need-to-know)

Dalam dunia Docker, unit terkecilnya adalah container. Sama halnya dengan dunia Kubernetes, pods adalah unit terkecil di dunia Kubernetes. Dan container dibuat didalam pods, baik satu atau bahkan beberapa container. Semua container yang berada di dalam pods yang sama akan saling berbagi volume dan IP Address.

1. **Apa itu Kubernetes Cluster dan Nodes?**



(Sumber: [cloudify.co)](https://cloudify.co/blog/migrating-pods-containerized-applications-nodes-kubernetes-cluster-using-cloudify/)

Kubernetes cluster terdiri dari node. Node merupakan server baik itu server bare metal dan server virtual. Node tidak akan secara otomatis membentuk cluster tanpa Anda konfigurasi terlebih dahulu. Setelah Anda konfigurasi, semuanya akan serba otomatis.

1. **Apa itu kubectl?**

**Kubectl** adalah sebuah *tools* *command* *line* yang memungkinkan Anda untuk terhubung ke Kubernetes cluster dan mengelolanya dari jarak jauh. Dengan menggunakan kubectl, Anda bisa terhubung ke layanan API pada sebuah node master melalui https.

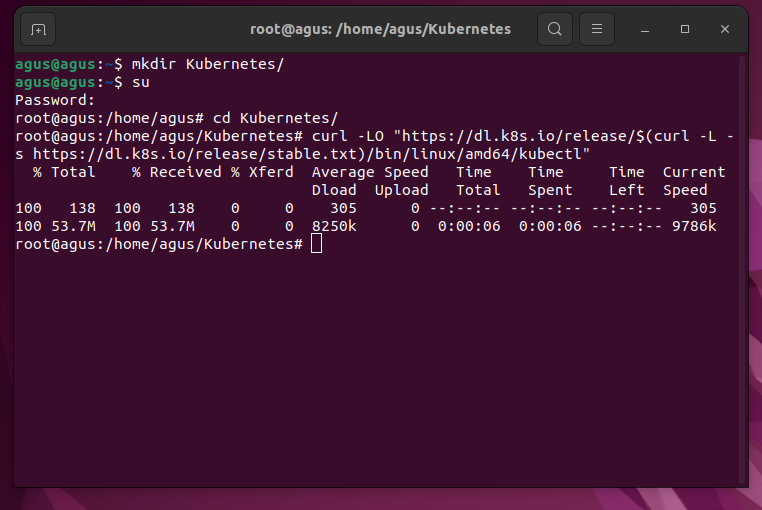
# Praktikum

1. Sebelum dilakukan instalasi Minikube, Anda pastikan bahwa system Anda memenuhi persyaratan berikut:
   1. **CPU : 2 atau lebih**
   2. **Memory : 2GB atau lebih**
   3. **Harddisk : 40GB atau lebih**
   4. **Koneksi internet**

1. Kemudian pastikan bahwa pada host OS Anda sudah terinstal Docker. Jika tidak ada, Anda bisa melihat instalasi Docker pada praktikum sebelumnya.

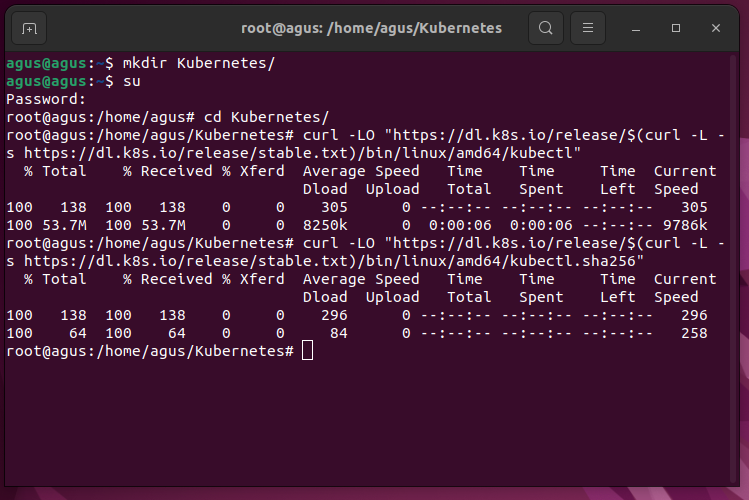
1. Setelah Anda instal package yang missing secara manual, Anda downloadlah kubectl dengan menjalankan *command* berikut.

|  |  |
| --- | --- |
| curl -LO "https://dl.k8s.io/release/$(curl -L https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl" | -s |



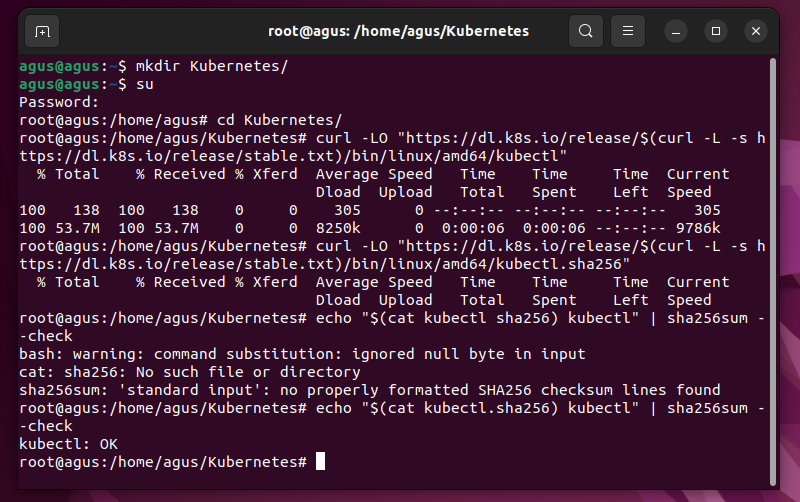
1. Lalu Anda bisa mendownload file checksum kubectl dengan menjalankan *command* berikut.

curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256"



1. Kemudian Anda bisa melakukan validasi kubectl terhadap file checksum dengan menjalankan *command* berikut.

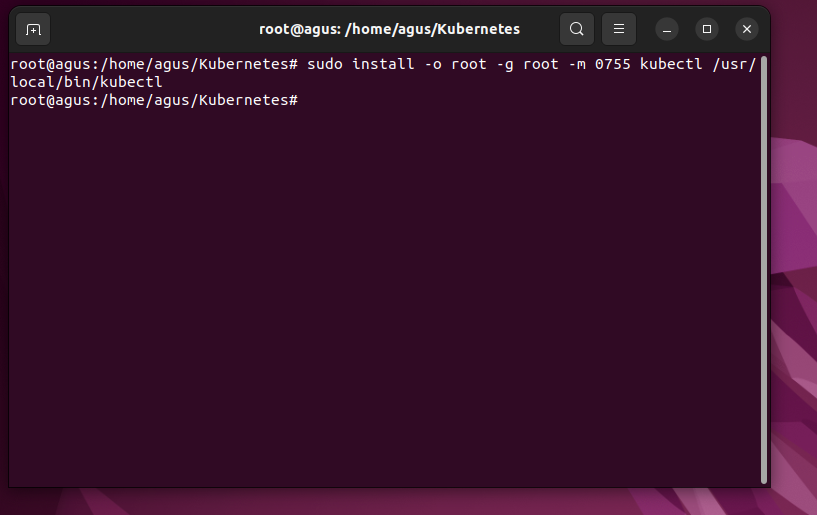
echo "$(cat kubectl.sha256) kubectl" | sha256sum --check



Pada gambar diatas menunjukkan bahwa kubectl sudah valid.

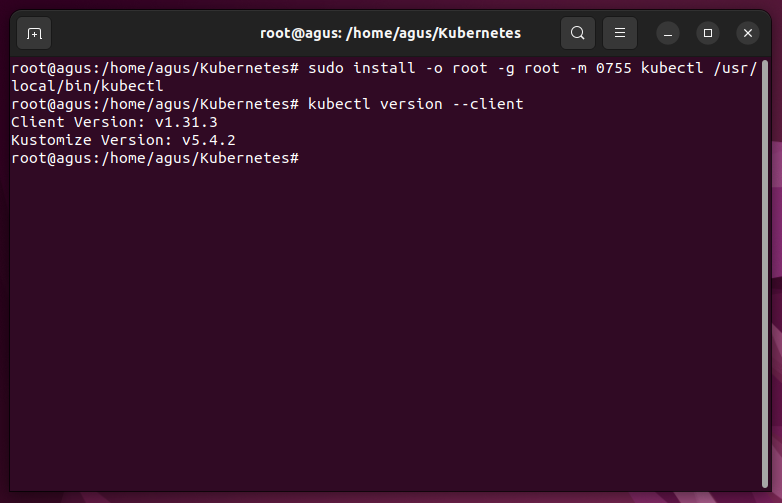
1. Setelah didownload dan sudah valid, Anda bisa lakukan instalasi kubectl dengan menjalankan *command* berikut.

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl



1. Setelah terinstal Anda bisa memastikan versi kubectl yang Anda instal dengan menjalankan *command* berikut.

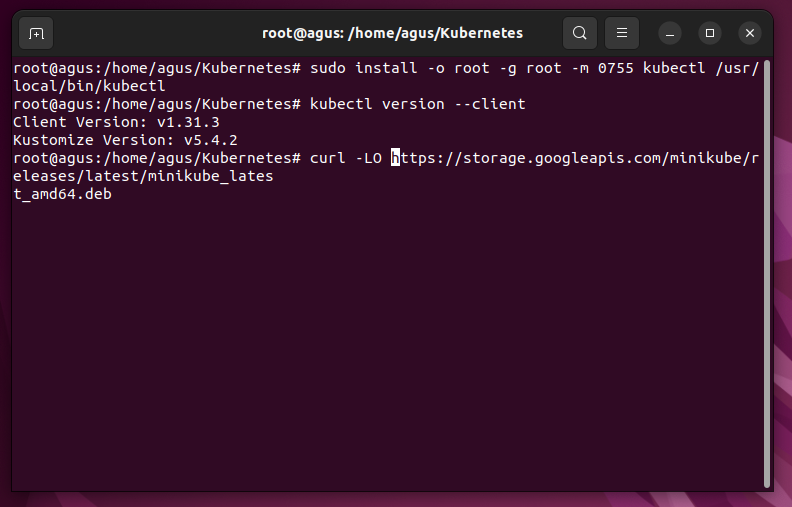
kubectl version --client



1. Setelah itu, Anda sudah bisa melakukan download Minikube dengan menjalankan *command* berikut.

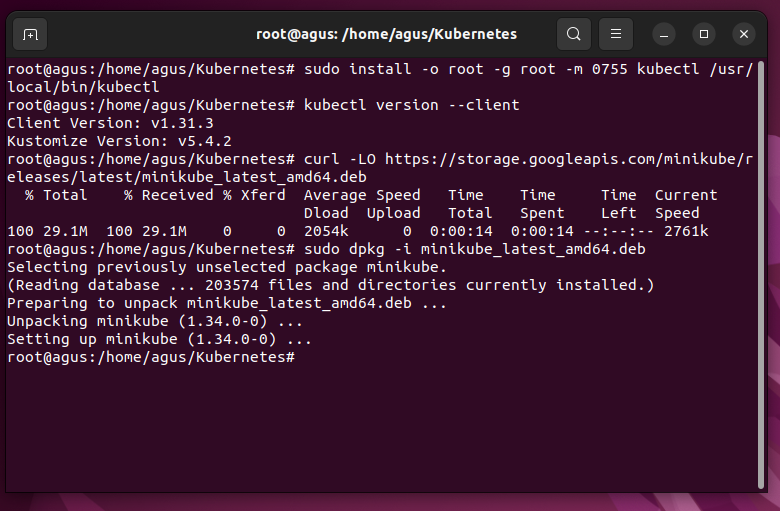
curl -LO

https://storage.googleapis.com/minikube/releases/latest/minikube\_lates t\_amd64.deb



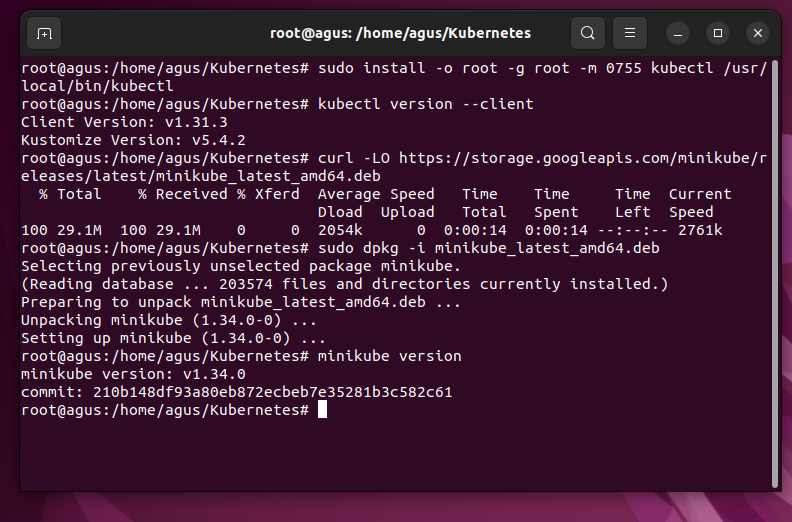
1. Setelah didownload, Anda bisa lakukan instalasi minikube menggunakan package Debian dengan menjalankan *command* berikut.

sudo dpkg -i minikube\_latest\_amd64.deb



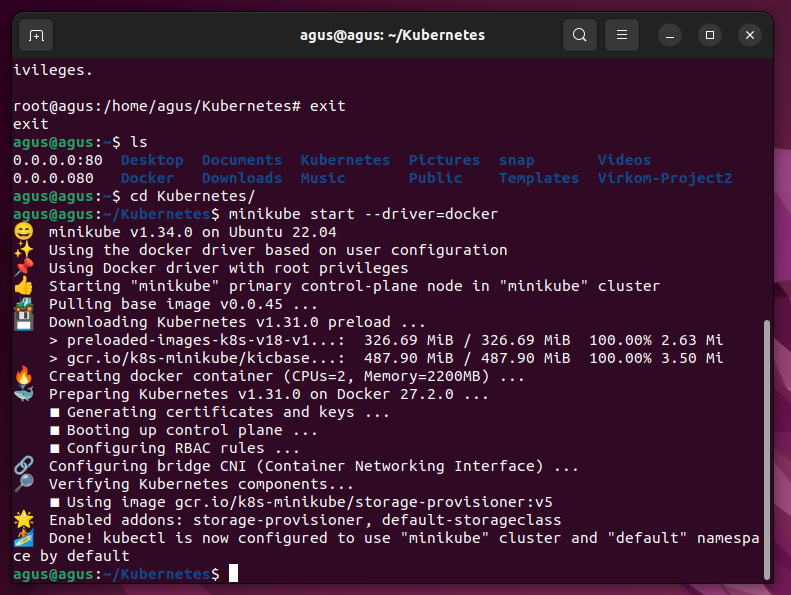
1. Kemudian Anda bisa memastikan versinya dengan menjalankan *command* berikut.

minikube version



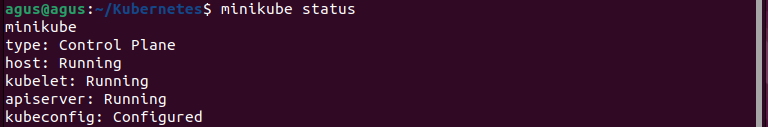
1. Setelah itu Anda akan mencoba membuat sebuah cluster Kubernetes dengan mengetikkan *command* berikut.

minikube start --driver=docker



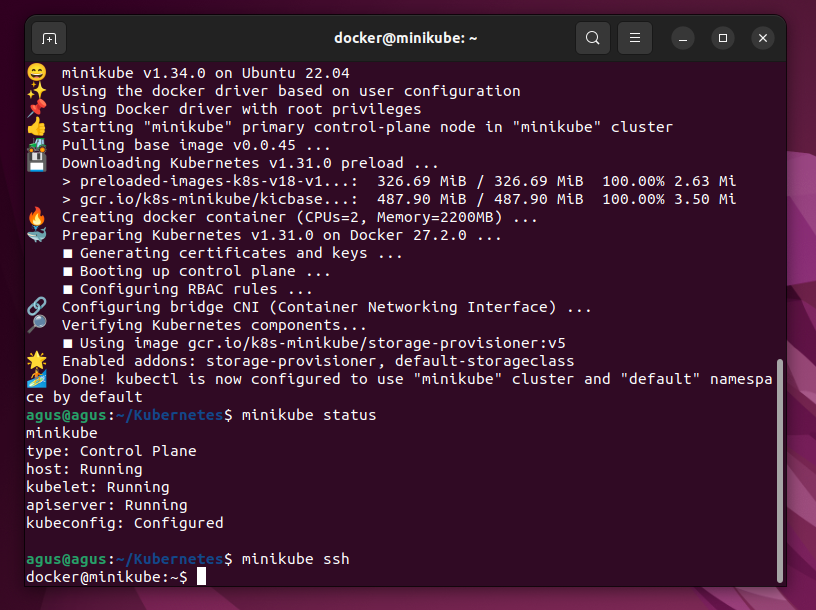
1. Lalu Anda bisa melihat status dari minikube itu sendiri dengan menjalankan *command* berikut.

minikube status

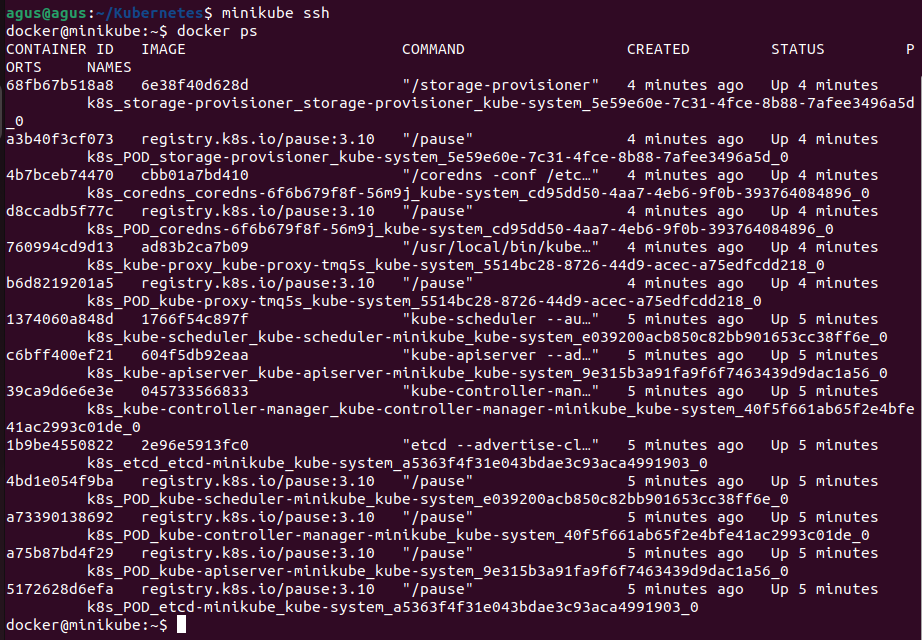


1. Setelah itu Anda bisa melakukan remote menggunakan protocol ssh dengan menjalankan *command* berikut.

minikube ssh

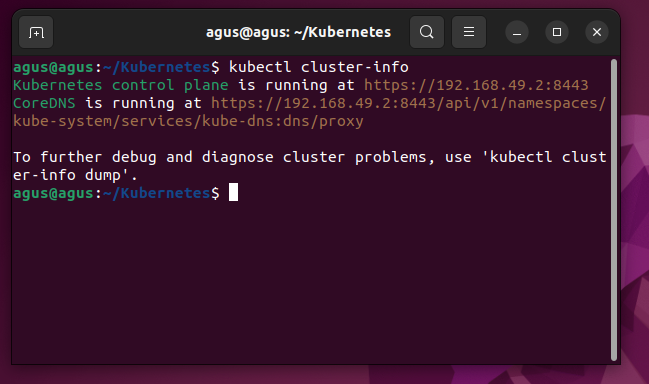


1. Setelah berhasil diremote, sekarang Anda bisa memverifikasi container apa saja yang dibuat didalam node yang baru Anda buat dengan menjalankan *command* berikut.



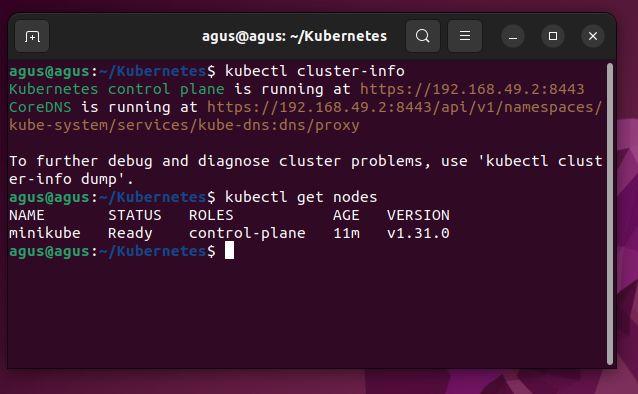
1. Kemudian Anda bisa melihat informasi yang tersedia pada Kubernetes cluster yang baru saja Anda buat dengan menjalankan command berikut

kubectl cluster-info



1. Lalu Anda lihatlah daftar node yang tersedia di Kubernetes cluster Anda dengan menjalankan *command* berikut.

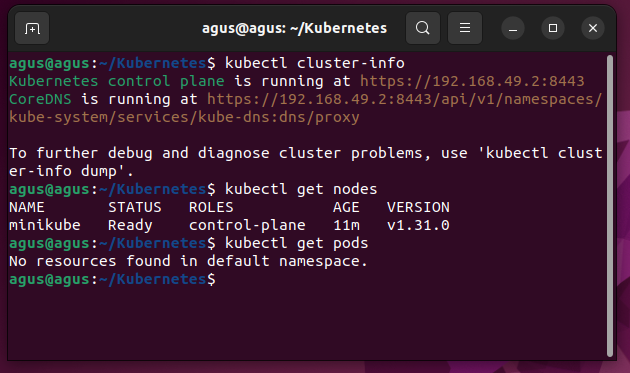
kubectl get nodes



Disini terlihat bahwa hanya satu node dengan nama minikube yang tersedia, karena minicube membuat cluster node tunggal.

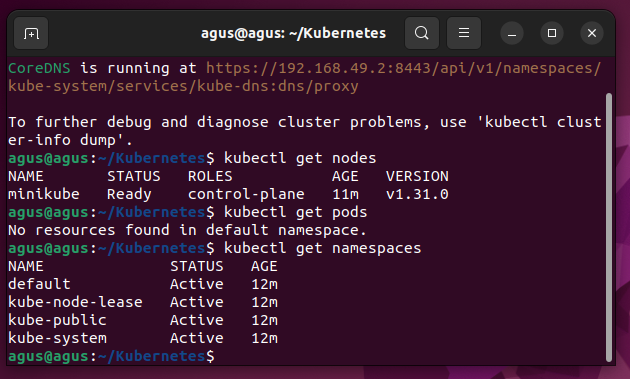
1. Kemudian, Anda bisa melihat pods yang tersedia didalam namespace dengan menjalankan *command* berikut.

kubectl get pods



1. Lalu Anda bisa melihat daftar namespace yang tersedia saat ini pada Kubernetes cluster Anda dengan menjalankan *command* berikut.

kubectl get namespaces

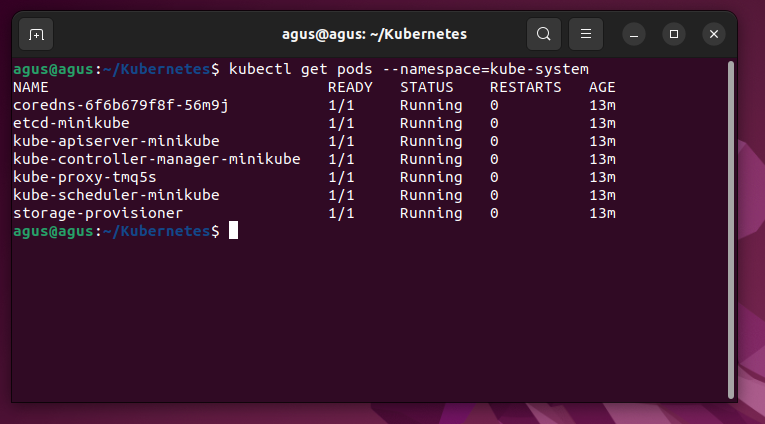


**Note:**

Namespaces digunakan untuk mengelompokkan *resource* yang berbeda dan objek konfigurasi.

1. Jika Anda ingin mengetahui pods mana yang berjalan didalam namespace tertentu, Anda bisa menjalankan *command* berikut.

kubectl get pods –namespace=kube-system



Disini terlihat bahwa beberapa pods yang berjalan pada namespace **kube-system**.

1. Selesai!