

[Home](#) [Articles](#) [Kubernetes Book](#) [CKAD Certification Course](#)

How to Install Kubernetes on Mac

This is a step-by-step guide to installing and running Kubernetes on your Mac so that you can develop applications locally.

You will be guided through running and accessing a Kubernetes cluster on your local machine using the following tools:

- Homebrew
- Docker for Mac
- Minikube
- virtualbox
- kubectl

Installation Guide

The only pre-requisite for this guide is that you have [Homebrew](#) installed. Homebrew is a package manager for the Mac. You'll also need [Homebrew Cask](#), which you can install after Homebrew by running `brew tap caskroom/cask` in your Terminal.

1. Install [Docker for Mac](#). Docker is used to create, manage, and run our containers. It lets us construct containers that will run in Kubernetes Pods.
2. Install [VirtualBox for Mac](#) using Homebrew. Run `brew cask install virtualbox` in your Terminal. VirtualBox lets you run virtual machines on your Mac (like running Windows inside macOS, except for a Kubernetes cluster.)

Skip to step three if everything has worked to this point.

In my case, I already had the non-Homebrew VirtualBox app installed which caused issues when trying to start minikube.

If you already have VirtualBox installed, start the installation as before with `brew cask install virtualbox`. You will get a warning that confirms this saying `Warning: Cask 'virtualbox' is already installed..` Once this is confirmed, you can reinstall VirtualBox with Homebrew by running `brew cask reinstall virtualbox`.

If you happen to have VirtualBox already running when you do this, you could see an error saying `Failed to unload org.virtualbox.kext.VBoxDrv - (libkern/kext) kext is in use or retained (cannot unload)`.

This is because the kernel extensions that VirtualBox uses were in use when the uninstall occurred. If you scroll up in the output of that command, beneath `Warning! Found the following active VirtualBox processes:` you'll see a list of the processes that need to be killed.

Kill each of these in turn by running `kill first_column_number` (`first_column_number` is the process identifier for that process).

Now re-run `brew cask reinstall virtualbox` and it should succeed.

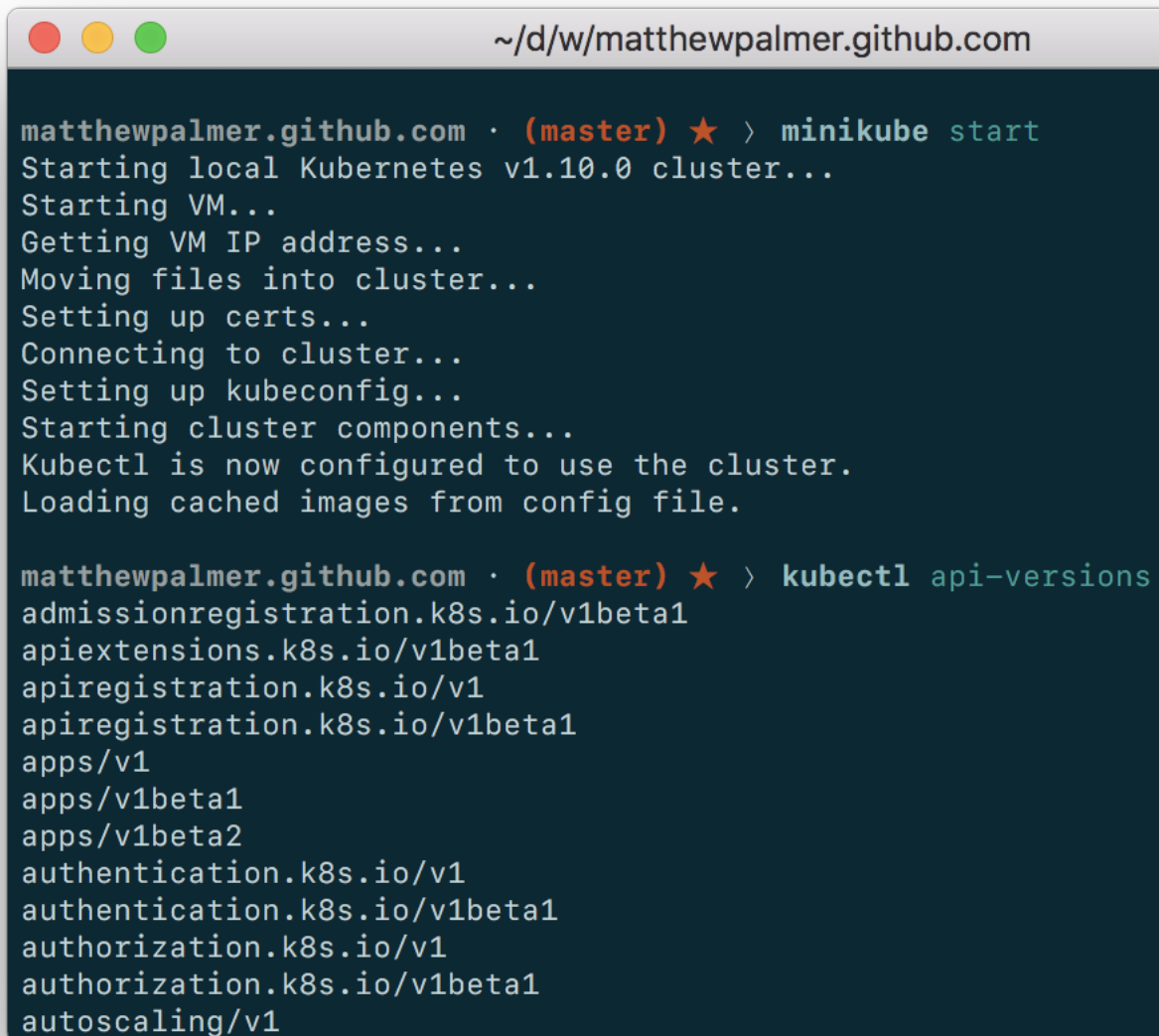
3. Install [kubectl](#) for Mac. This is the command-line interface that lets you interact with Kubernetes. Run `brew install kubectl` in your Terminal.
4. Install [Minikube](#) via the [Installation > OSX instructions from the latest release](#). At the time of writing, this meant running the following command in Terminal...

```
curl -Lo minikube https://storage.googleapis.com/minikube/releases/v0.27.0/minikube-darwin-amd64 &&\
  chmod +x minikube &&\
  sudo mv minikube /usr/local/bin/
```

Minikube will run a Kubernetes cluster with a single node.

5. Everything should work! Start your Minikube cluster with `minikube start`. Then run `kubectl api-versions`. If you see a list of versions, everything's working! `minikube start` might take a few minutes.

At this point, I got an error saying `Error starting host: Error getting state for host: machine does not exist`. because I had previously tried to run Minikube. You can fix this by running `open ~/.minikube/` to open Minikube's data files, and then deleting and deleting the machines directory. Then run `minikube start` again.

A terminal window with a dark blue background and white text. The window title bar shows three colored circles (red, yellow, green) and the path ~/d/w/matthewpalmer.github.com. The terminal output shows the execution of 'minikube start' and 'kubectl api-versions' commands. The 'minikube start' command outputs several status messages: 'Starting local Kubernetes v1.10.0 cluster...', 'Starting VM...', 'Getting VM IP address...', 'Moving files into cluster...', 'Setting up certs...', 'Connecting to cluster...', 'Setting up kubeconfig...', 'Starting cluster components...', 'Kubectl is now configured to use the cluster.', and 'Loading cached images from config file.' The 'kubectl api-versions' command outputs a list of API versions: 'admissionregistration.k8s.io/v1beta1', 'apiextensions.k8s.io/v1beta1', 'apiregistration.k8s.io/v1', 'apiregistration.k8s.io/v1beta1', 'apps/v1', 'apps/v1beta1', 'apps/v1beta2', 'authentication.k8s.io/v1', 'authentication.k8s.io/v1beta1', 'authorization.k8s.io/v1', 'authorization.k8s.io/v1beta1', and 'autoscaling/v1'.

```
matthewpalmer.github.com · (master) ★ > minikube start
Starting local Kubernetes v1.10.0 cluster...
Starting VM...
Getting VM IP address...
Moving files into cluster...
Setting up certs...
Connecting to cluster...
Setting up kubeconfig...
Starting cluster components...
Kubectl is now configured to use the cluster.
Loading cached images from config file.

matthewpalmer.github.com · (master) ★ > kubectl api-versions
admissionregistration.k8s.io/v1beta1
apiextensions.k8s.io/v1beta1
apiregistration.k8s.io/v1
apiregistration.k8s.io/v1beta1
apps/v1
apps/v1beta1
apps/v1beta2
authentication.k8s.io/v1
authentication.k8s.io/v1beta1
authorization.k8s.io/v1
authorization.k8s.io/v1beta1
autoscaling/v1
```

Come Together

You've installed all these tools and everything looks like it's working. A quick explanation of how the components relate is needed.

- VirtualBox is a generic tool for running virtual machines. You can use it to run Ubuntu, Windows, etc. inside your macOS operating system host.
- Minikube is a Kubernetes-specific package that runs a Kubernetes cluster on your machine. That cluster has a single node and has some unique features that make it more suitable for local development. Minikube tells VirtualBox to run. Minikube can use other virtualization tools—not just VirtualBox—however these require extra configuration.
- kubectl is the command line application that lets you interact with your Minikube Kubernetes cluster. It sends request to the Kubernetes API server running on the cluster to manage your Kubernetes environment. kubectl is like any other application that runs on your Mac—it just makes HTTP requests to the Kubernetes API on the cluster.



Learn Kube Faster.
Get the #1 guide.

Get my book on Kubernetes for software developers, used by engineers at Google, Microsoft, and IBM.

[Get Book](#)

Meet the Author



[Matthew Palmer](#) is a software developer and author. He's created popular desktop apps, scaled SaaS web services, and taught Computer Science students at the University of New South Wales.

Contact

Email: hello@matthewpalmer.net

Email List: [Sign Up](#)

Twitter: [@_matthewpalmer](#)

Github: [matthewpalmer](#)