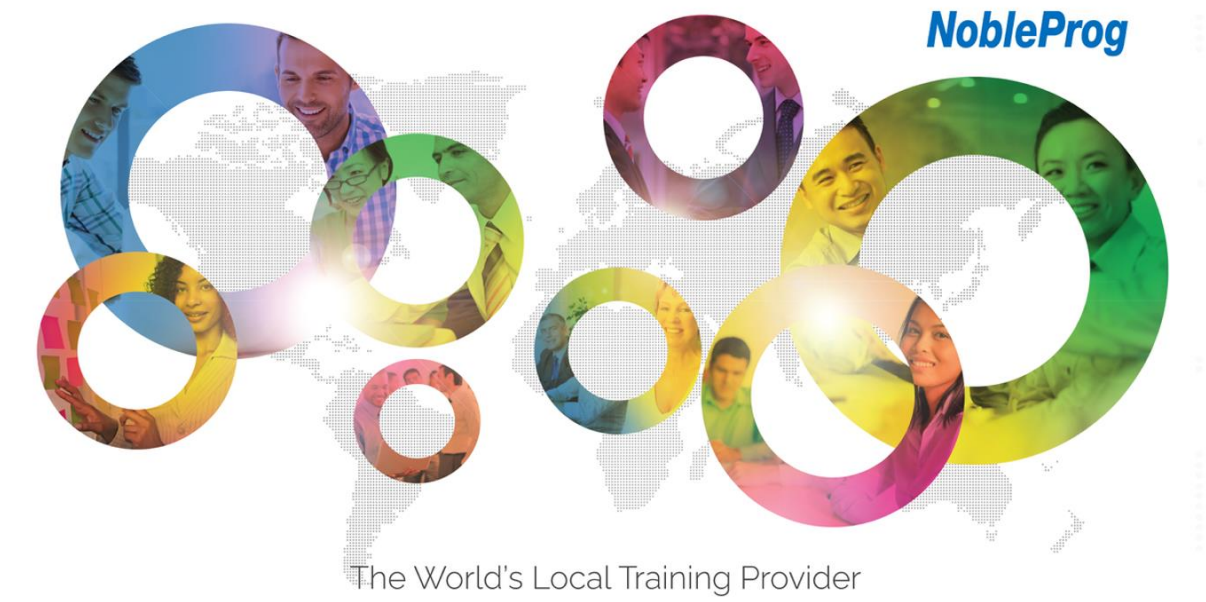


Docker and Kubernetes Training

Requirements

Table of Contents

Mandatory

- Delegates should have administrative access to their laptop or system to setup Docker and Kubernetes with minimum of 8GB RAM.
- VT-x/AMD-v virtualization must be enabled in BIOS in the laptop or system
- A credit card is required to deploy projects to Google Kubernetes Engine (GKE) 12 Month free trial. You must provide credit card details to set up a billing account to verify your identity, but you won't be charged during the free trial.
- High speed internet connection

Installing Docker

If you don't already have docker installed, Docker already has some great guides on how to do it. This guide is about how to setup Docker on your specific OS, but if you already know which OS you want to install on, here's the short-list for downloading it.

Installing on Windows 10 (Pro or Enterprise)

This is the best experience on Windows, but due to OS feature requirements, it only works on the Pro and Enterprise editions of Windows 10 (with latest update rollups). You need to install "Docker for Windows" from the Docker Store.

With this Edition, it's recommended using PowerShell for the best CLI experience.

Installing on Windows 7, 8, or 10 Home Edition

Unfortunately, Microsoft's OS features for Docker and Hyper-V don't work in these older versions, and "Windows 10 Home" edition doesn't have Hyper-V, so you'll need to install the Docker Toolbox, which is a slightly different approach to using Docker with a VirtualBox VM. This means Docker will be running in a Virtual Machine that sits behind the IP of your OS, and uses NAT to access the internet.

NOTE: For all examples that use http://localhost, you'll need to replace with http://192.168.99.100

Installing on Mac

You'll want to install Docker for Mac, which is great. If you're on an older Mac with less than OSX Yosemite 10.10.3, you'll need to install the Docker Toolbox instead.

Installing on Linux

Do *not* use your built in default packages like **`apt/yum install docker.io`** because those packages are old and not the Official Docker-Built packages.

It's preferred to use the Docker's automated script to add their repository and install all dependencies: `curl -sSL https://get.docker.com/ | sh` but you can also install in a more manual method by following specific instructions on the DockerStore for your distribution, like this one for Ubuntu.

What if None of These Options Work

Maybe you don't have local admin, or maybe your machine doesn't have enough resources. Well the best free option here is to use play-with-docker.com, which will run one or more Docker instances inside your browser, and give you a terminal to use it with. You can actually create multiple machines on it, and even use the URL to share the session with others in a sort of collaborative experience. It's highly recommend you check it out. Most of the examples in this course can be used with "PWD", but its only limitation really is it's time bombed to 4 hours, at which time it'll delete your servers.

Code or Text Editor for Docker and Compose files

You'll need a text or code editor. Preferably one that can do Syntax Highlighting for Dockerfile and YAML files. Good free ones are GitHub's Atom, Microsoft's Visual Studio Code, or even VIM if you're into shell-based editors.

It's recommended using Visual Studio Code for the best experience.

Terminal Shell and Tab Completion for Docker CLI

Windows Terminals

On Windows you'll likely want to use PowerShell terminal rather than Command Prompt, but if you don't like the default GUI window that Microsoft provides, you might want to check out ConEmu, or cmder, which is built on ConEmu and easier to get started. You'll want to change both of those options to always use PowerShell though.

Windows PowerShell Tab Completion

Tab completion lets you quickly fill the rest of a command or argument as you type by hitting Tab. To get "tab completion" in Windows, it's easiest to use PowerShell and install the posh-docker Module by following this guide from the Docker Docs.

macOS Terminals

On Mac you can use the built in Terminal program which uses the default Bash shell. It’s preferred to use iTerm as GUI terminal interface to gain more features than the built in one.

macOSBash Tab Completion

Tab completion lets you quickly fill the rest of a command or argument as you type by hitting Tab. Assuming you're running Docker for Mac, you'll want to connect up the completion scripts that come with Docker for Mac to your shell. Docker Captain Alex Ellis has a nice walk through of how to get this working. If you're familiar with brew, you can find his method more up to date then using the brew packages for Docker completion scripts.

Linux

In a desktop Linux distro like Ubuntu, Mint, etc. you'll likely need to do nothing. They already have good built-in terminals and Bash tab completion works out of the box.

Installing Chocolatey

This package manager for Windows will make installing the programs you need easier. You can install it following the instruction here or by using the Windows command line prompt using the following command.

Open a command line window as administrator.

```
@"%SystemRoot%\System32\WindowsPowerShell\v1.0\powershell.exe" -NoProfile -InputFormat None -ExecutionPolicy Bypass -Command "iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1')) " && SET "PATH=%PATH%;%ALLUSERSPROFILE%\chocolatey\bin"
```

Installing Minikube

a) Open a command line window with administrator privileges.

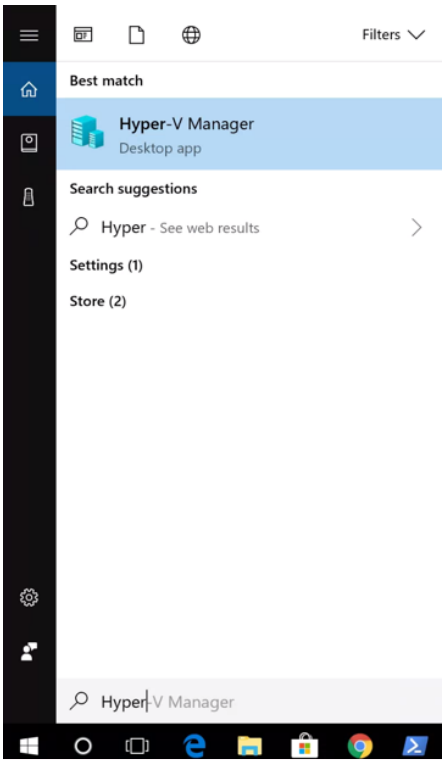
b) Use the chocolatey package manager to install minikube.

```
choco install minikube
```

c) Install the kubernetes command line program with the chocolatey package manager

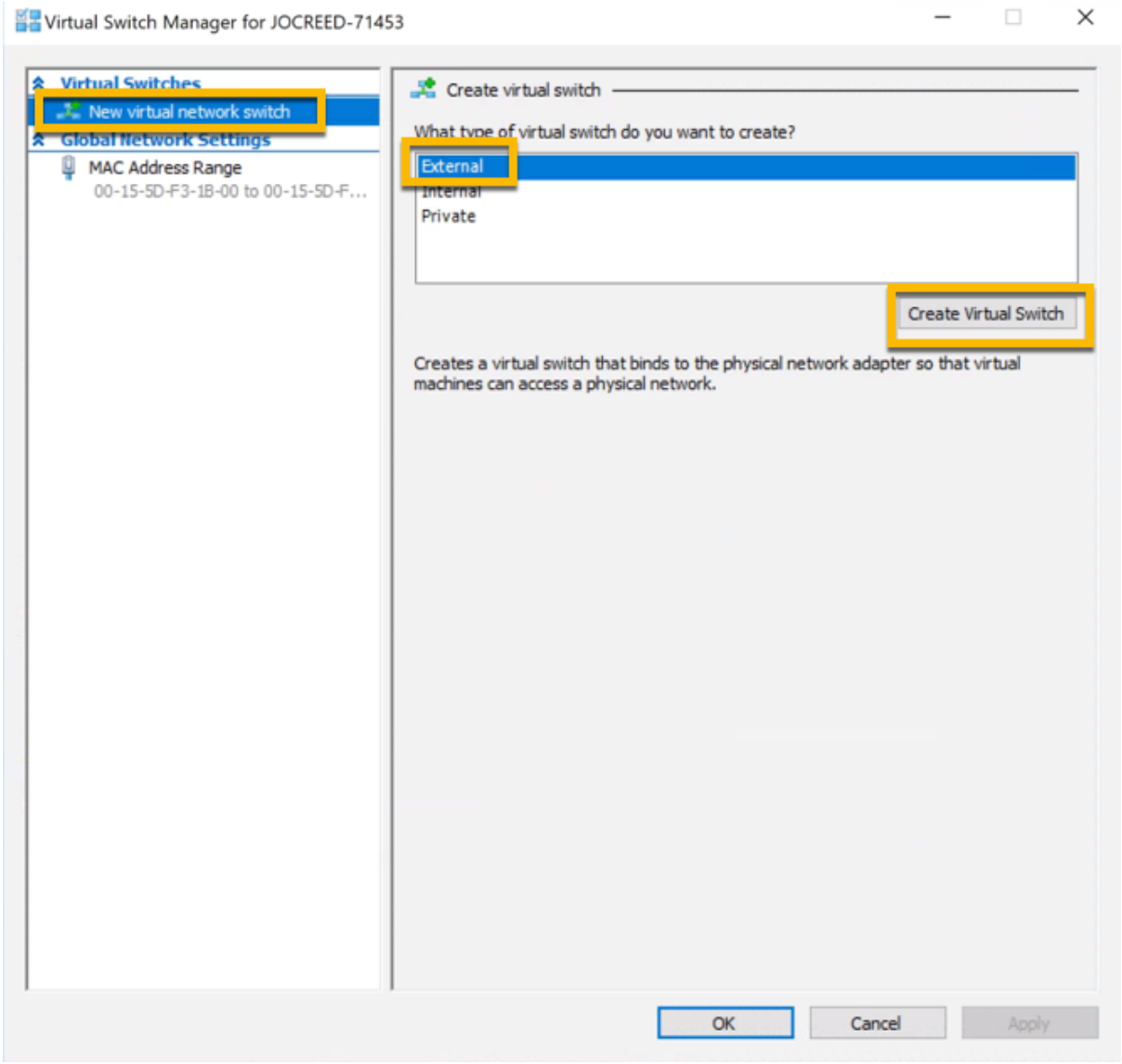
```
choco install kubernetes-cli
```

d) Open the Hyper-V Manager.



e) Once in the Hyper-V Manager, on the right panel, select the Virtual Switch Manager.

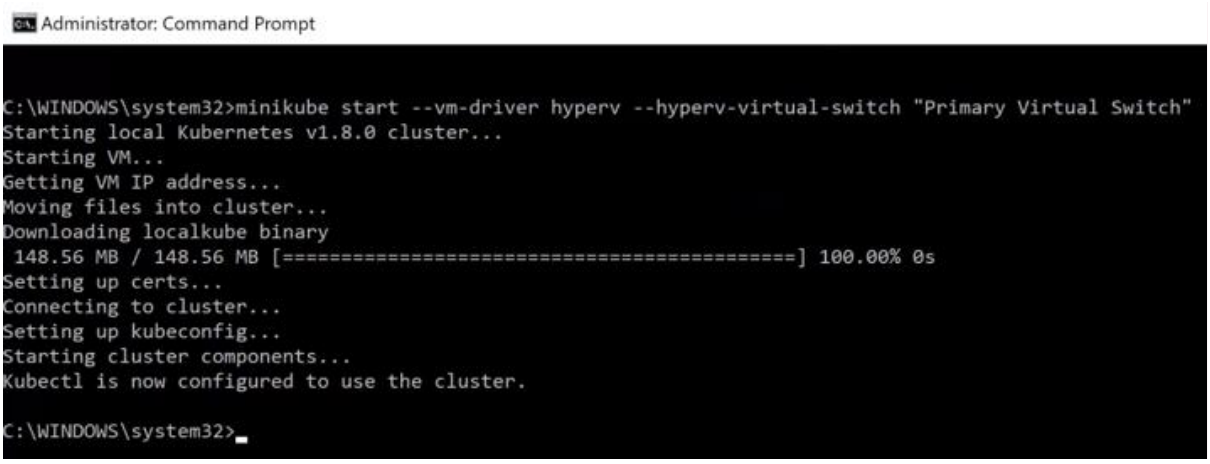
f) Create a virtual switch for minikube. Select **New virtual network switch** on the right-hand side, select **External** for the network type, and then press the **Create Virtual Switch** button.



g) Name the switch **Primary Virtual Switch** and click the apply button.

h) Run the following command to start the minikube VM

```
minikube start --vm-driver hyperv --hyperv-virtual-switch "Primary Virtual Switch"
```



i) Confirm the installation is ready by checking couple of commands in command line window,

```
kubectl get pods -n kube-system
```

```
minikube dashboard
```

Installing Git

Git is a free and open source distributed version control system. Download and install it from here.

Installing Google Cloud SDK

`gcloud` provides the primary command-line interface for GCP, and `kubectl` provides the command-line interface for running commands against Kubernetes clusters.

If you prefer using your local shell, you must install the `gcloud` and `kubectl` command-line tools in your environment.

Follow the instructions provided here to install the Google Cloud SDK and configure default settings for `gcloud`.