This is an export from this page:
https://github.com/jbellver99/minikube-podman/wiki/Installing-Podman-Minikube-CRI-O-as-Docker-Desktop-alternative

# Installing Podman Minikube CRI O as Docker Desktop alternative

You are a few steps far from moving away from Docker Desktop to a much powerful free alternative, which will enable you to introduce Kubernetes in your development environment if desired, or continue working as normal with plain Docker-like containers.

# Installation

The given commands are expected to be run in a PowerShell console. Both parts must be executed.

## 1. As administrator

```
Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V -All
```

Reboot if asked to do so.

```
([adsi]"WinNT://./Hyper-V Administrators,group").Add("WinNT://$env:UserDomain/$env:Username,user")

logoff;exit
```

Yo will be logged off. This is needed to make Windows aware of the needed group membership. Do not skip this step.

Install: https://storage.googleapis.com/minikube/releases/latest/minikube-installer.exe

## 2. As standard user

Then execute the script “installation_podman.ps1” For that, you need to open a new powershell console, type: `cd <path_to_the_extracted_archive>`

`.\installation_podman.ps1`

There is also the script `.\installation_podman_boosted.ps1` that exists, this script creates a VM with 40GB instead of 20 and uses 3500 MB of memory

Now you are able to use `podman <command>` (or its alias `docker <command>`) as well as `minikube` and `kubectl` freely.

# How to use Podman and what are the most used commands

The goal of this section is to give you the basics of minikube and podman, some links are given at the end if you desire to go further with podman.

## 3. Minikube

*"Minikube is a tools which enables you to run Kubernetes locally. It runs a single-node Kubernetes cluster on your computer".*

Podman is not normally compatible with Windows, that is why we will be using *Minikube*, we will have a Linux Virtual Machine with podman installed, and a podman remote client installed in our local windows machine. Every time that podman is called in our windows machine it will connect to the VM and execute the podman command.

⚠️ Note: Be careful when you try to access and application in a container, as with Docker Desktop you used to connect to your localhost, but with podman you will need to connect to the virtual machines IP. In order to know the virtual machines IP, execute the following command: `minikube ip`.

Other commands that you will also need are the following:

- `minikube start` Starts a local kubernetes cluster on your machine.
- `minikube stop` Used to stop minikube correctly.

- `minikube pause` Leaves kubernetes in pause, but the VM will continue running. (If you just need podman, we advice you to use this command in order to get a better performance.
- `minikube delete` This command will remove permanently your minikube node.
- `minikube minikube start --container-runtime=cri-o --cpus 4 --memory $(((Get-CimInstance Win32_PhysicalMemory | Measure-Object -Property capacity -Sum).sum - 6gb) /1mb)` Used to create a new minikube node, it only works if there is no other node running.

You can also check `minikube --help` for more information on commands.

We also created custom commands to save all images or load them at the start, because if you use minikube stop, all your local images will be deleted:

- `minikube_save_images` Save all images as tar files in a tmp directory, if the `-s` is detected, will shut down the VM (as a `minikube stop` would do)
- `minikube_load_images` Load all the tar files generated by `minikube_save_images` to recreate the images on the VM, then for every tar file loading succesfuly it deletes it. If the `-s` flag is detected, will start the VM (as a `minikube start` would do), Note: the loading of images cananot work if your VM is shutdown

ℹ️ Every time you stop/start your VM, nothing will be saved. You should not modify anything directly on the VM.

⚠️ Sometimes, if you shut down your VM with existing containers inside, when you start again your VM, you may have problems with the container names. It may tell you that the name of container is already in use but that container name will not exists if you try to erase it.

# 4. How to use Podman

With the installation of podman, an alias for docker was created, this means that whenever you write `docker <command>` in your command line, you will be calling `podman <command>`.

The vast majority of commands that you were using with Docker will work exactly the same with podman. Podman will read Dockerfiles the same way as Docker Desktop was doing.

ℹ️ For those who have not used Docker, here is a quick memo of basic commands: https://www.pluralsight.com/courses/getting-started-docker.

In order to build an image and run a container with this image, follow these steps:

1. Create the Dockerfile with the instructions to build your app.
2. Once you got your Dockerfile, open a Powershell prompt where your file is located.
3. Execute `podman build -t <tag>`, if you want to push it to an online repository, then the tag should look similar to this: `<user>/<repository>:<tag>`.
4. Execute `podman push "image tag" --creds <user>:<password>` (Optional). Execute this step only if you want to push your image into an online repository. It uses Docker hub as predetermined.
5. Run the container with the command `podman run --name <name_of_your_container> <tag> -d -p <container_port:local_port>`.
6. You are able to check that your container is running with `podman ps`.
7. Finally, you will need to stop, start and delete your containers sometimes, this is done with the following commands:

- `podman stop <name_of_your_container>`
- `podman start <name_of_your_container>`
- `podman rm <name_of_your_container>`

In some cases you might get a problem with pulling images from repository because the certificate is not recognized. In this case you can modify the registries.conf file (inside the folder `C:\Users\<user>\Downloads\podman-2.2.1)` this file looks initially like that:

```
[registries.search]
registries = ['docker.io']

[registries.insecure]
registries = ['']
```

if you want to authorize connection to the repository you have to change your file so it looks like this (in this example we authorized two registry) :

```
[registries.search]
registries = ['docker.io']

[registries.insecure]
registries = ['registry.example']
insecure=false
registries = ['registry2.example2']
insecure=false
```

## 5. Podman-compose

Podman does not include a function to launch interdependent containers. For this, a python script has been created, and has been installed with the installation script. This script can read docker-compose.yml files, and will create a pod and launch your different containers inside this pod.

## What is a pod?

*"Pods are a way of grouping containers together inside their own namespace, network and security context".*

Fore more information about pods: https://developers.redhat.com/blog/2019/01/15/podman-managing-containers-pods#:~:text=The%20Pod%20concept%20was%20introduced,other%20containers%20to%20the%20pod..

For the moment, it is not possible to use podman-compose in windows, and it is not included in the minikube yet, so a script has been created in order to execute podman-compose.

ℹ️ This script creates a unix container with podman-compose installed on it and will use this container to generate all the commands that podman-compose will execute and then execute them inside the minikube VM.

To call this script just:

1. Place yourself inside the directory where your docker-compose.yml file is.
2. Execute `podman-compose` which is an alias for `C:\Users\<user>\Downloads\podman-2.2.1\podman_compose_Windows_part.ps1`.

There are two flags that can be used, `-m` and `-u`:

- It needs to have a shared folder between the windows host and the minikube VM, in order to create it, add the `-m` flag. This will open a new Powershell prompt in the background, once the sharing is done the script will continue. (It is recommended to leave the prompt running).
- It is possible that inside the shared folder that we have just created in the VM becomes corrupted. In this case, close the Powershell prompt which is running the process and use the `-u` flag. This flag will unmount the folder inside the VM. (If you activate this flag, the `-m` one will be automatically triggered).

# 6. Differences with Docker

- No HMI.
- In order to use compose, you can only play the "xxxxx compose up".
- No equivalent for docker-swarm.
- It is possible to manage pods with podman.

# 7. Using podman with the VPN

It is not allowed for security reason to have a VM on the network when connectiong to VPN.

# 8 More documentation

Minikube documentation: https://minikube.sigs.k8s.io/docs/start/

Oficial documentation of Podman https://docs.podman.io/en/latest/

News feed about podman: https://podman.io/

Github for podman and podman-compose: https://github.com/containers/podman
https://github.com/containers/podman-compose