

Kubernetes Problem-Solving Workshop - Setup

Version 2.2 – 04/11/22

Brent Laster, Lead Instructor - Tech Skills Transformations LLC

IMPORTANT SETUP INSTRUCTIONS – PLEASE COMPLETE BEFORE THE CLASS.

ALSO, IT IS RECOMMENDED TO PRINT A COPY OF THE CLASS LABS TO HAVE WITH YOU IF YOU PREFER TO WORK FROM A PAPER COPY. THE LABS ARE SIGNIFICANT.

- Labs will be made available closer to the workshop date.

Hello and thank you for registering for this workshop. In order to be able to work through the labs and effectively understand the material, it will be to your advantage to get your system setup following the instructions below.

Note: This workshop assumes you have a working knowledge of Kubernetes. This is a prerequisite for success in this workshop.

This workshop has labs and it is intended that **you use your own computer to do these on, so plan to bring a laptop with you ALREADY SETUP per the directions below.** There will not be time to get everything setup at the workshop.

You may choose to setup your laptop environment either by running the application VirtualBox and a preconfigured virtual machine (VM) with all of the applications already installed and setup on it or by installing and configuring the applications yourself directly on your system. Instructions for the VirtualBox approach are below under the heading **"Option B: VirtualBox Option."** Instructions for the self-configured approach are below under the heading **"Option A: Manual setup"**.

While the VirtualBox Option is the simplest in terms of steps required and is the recommended approach, there are times when students may run into issues with running VirtualBox on their systems. For that case, or if you prefer not to run VirtualBox, you may install the applications separately on your laptop and run them there.

For either option, it is important that you verify you can get to and use a Kubernetes environment on your laptop prior to the workshop. At a minimum, you should be able to run the following commands and see appropriate output.

"kubectl get ns"

"helm version"

After setting up the environment, follow the pre-req steps at the bottom of the document under "Startup - to do before first lab" to be ready for the workshop.

Option A: Manual setup

1. Install a version of [Git](#). (If you are installing on a Windows system, it is recommended to also install the Git Bash Shell for Windows.)
2. Install [Docker](#) (any edition is fine). (Note on Windows or Mac, you may need to install the [Docker desktop](#).)
3. Install and run a Kubernetes cluster using any package/application you want such as [minikube](#), [kind](#), etc. (It is not recommended to rely on a cloud instance of Kubernetes due to possible internet bandwidth limitations.)
4. You only need a single node for your Kubernetes instance as a minimum.

© 2022 Brent Laster & Tech Skills Transformations LLC

5. The workshop was designed for Kubernetes version 1.21, though other versions may work.
6. Install the Kubernetes command line tool, [kubectl](#).
7. Install [Helm 3](#). Helm 2 will not work.
8. If running on Windows, install some version of grep. If you already have the Git Bash Shell installed, it should have grep installed. You can install that as part of installing [Git for Windows](#).
9. Optional - install [jq](#) for your platform if running MacOS or Linux and you don't already have it.
10. Optional - install the Kubernetes Metrics Server

Option B: VirtualBox Option

1. You must have a system that can support virtualization and run Virtualbox without problems. Download and install Virtualbox on your system and verify that it runs correctly. Note that some systems may require special access or BIOS settings to support virtualization. Please ensure that you have sorted out any issues with this prior to the start of the first online session. Virtualbox can be obtained and installed from <http://www.virtualbox.org>

2. The class uses a VirtualBox VM with all of the applications installed and configured that we will need. You can download the virtual image from the location noted on the course's landing page. Note that this file is about 5 gig in size, so it may take some time to download (more on a slow connection). It is not recommended to try to download this while you are using a VPN connection as that will greatly slow down the download.

Checksums to check your file downloaded:

MD5 Checksum: 484957278ddf510c21aab0f271108b03

SHA-1 Checksum: 42e4eddaaee695ee53a355888fe053499bb74e72

SHA-256 Checksum: 3cff8368735d9442ca151d2caa4e732e65df972279b947599b5194faa80bfd07

SHA-512 Checksum:

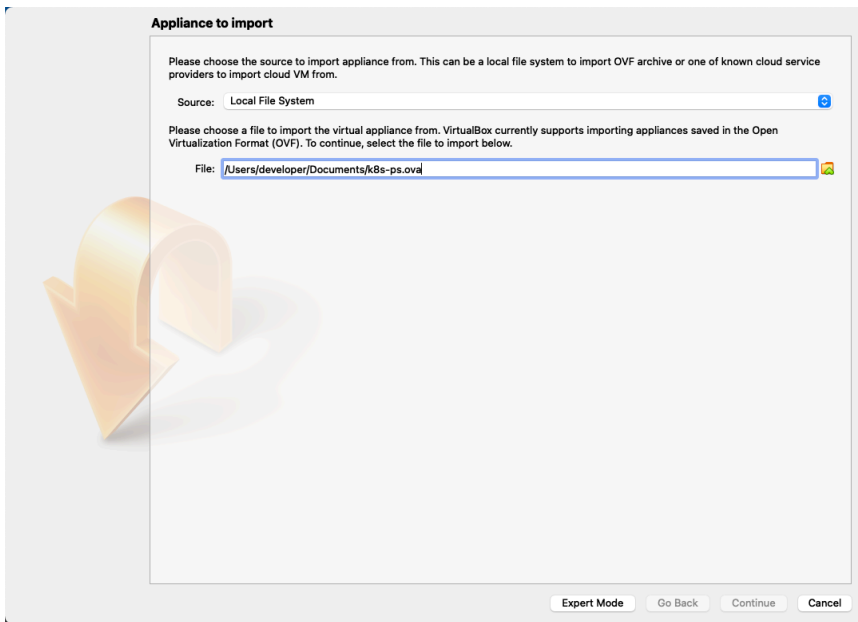
4b7f5503b8aa1213ca665bf0df5bced61039cfd33e4aeb104278f8175ae090fb506d7730155aa41d3
6769c43b90730a04d359b328af8fc5beee16d4e9d04f702

3. Once you have Virtualbox installed and the image downloaded and are ready to proceed, do the steps below to import the appliance.

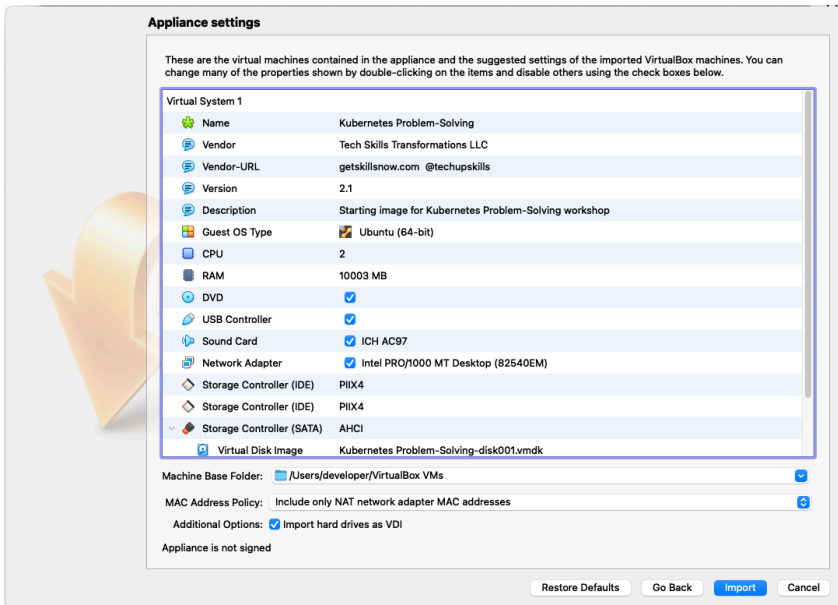
- a. Open **VirtualBox** on your system.
- b. From the **File** menu, select **"Import Appliance..."**.



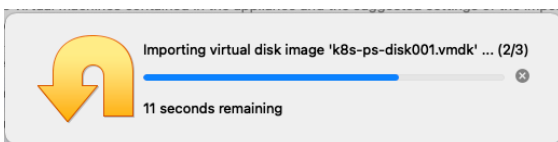
c. From there, you can type in (or browse to, using the folder icon) the path of the **k8s-ps.ova** file. Then click **Next**.



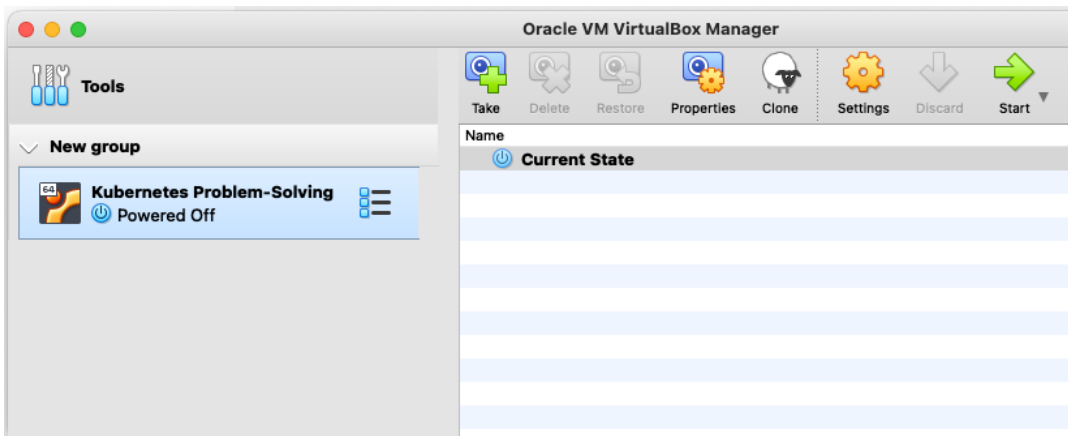
4. On the next screen, if given the option, click the box to reinitialize the MAC addresses. You can just accept the rest of the **Appliance Settings** and then click the **Import** button.



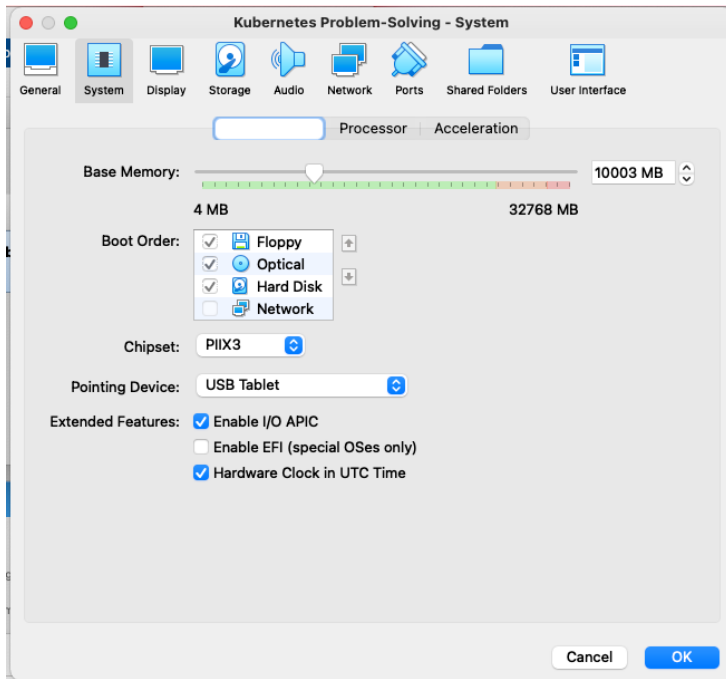
5. You will get a pop-up box for the “license” info. Just click the **Agree** button. Your system will then start processing the import. This may take a while.



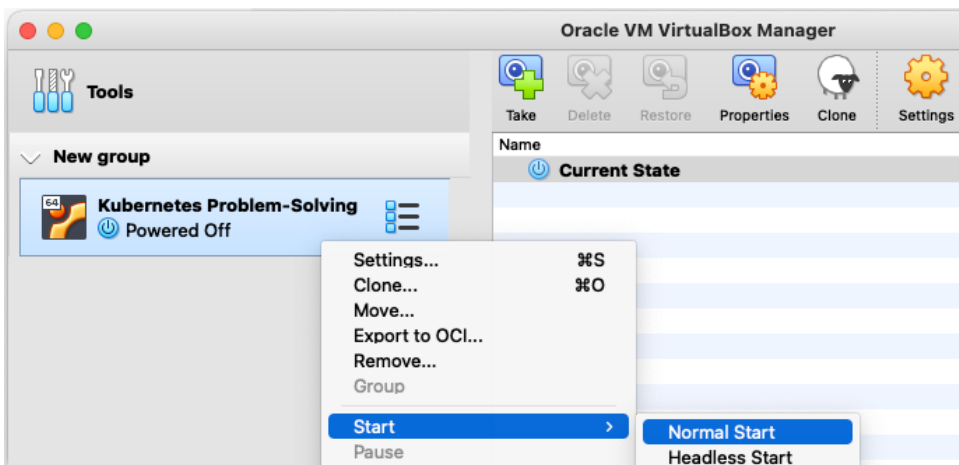
6. After the import is finished, you should have a VM listed in VirtualBox named **Kubernetes Problem-Solving**.



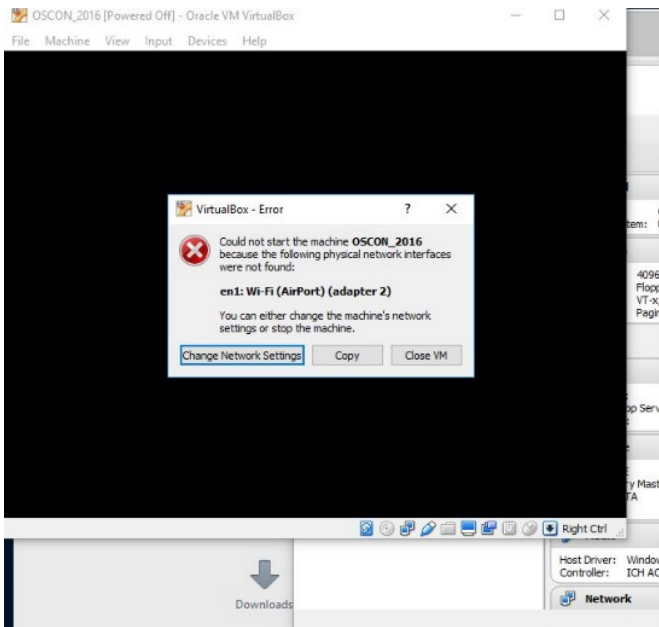
7. **(OPTIONAL)** At this point, depending on the settings of your physical system, you can adjust the amount of memory for the image if you need/want. You can do this by clicking on the **Settings** icon in the menu bar, then **System** in the pop-up box for the settings. Then you can adjust the amount of memory for the virtual machine with the slider.



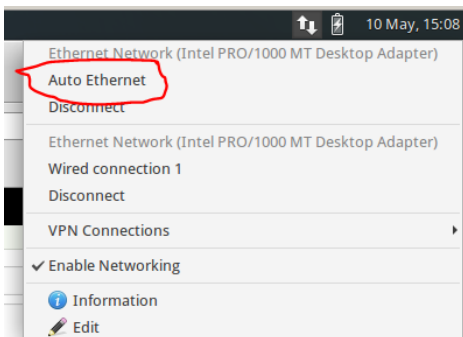
8. At this point, you can start up the virtual image by right-clicking on the image name and then selecting **Normal Start**.



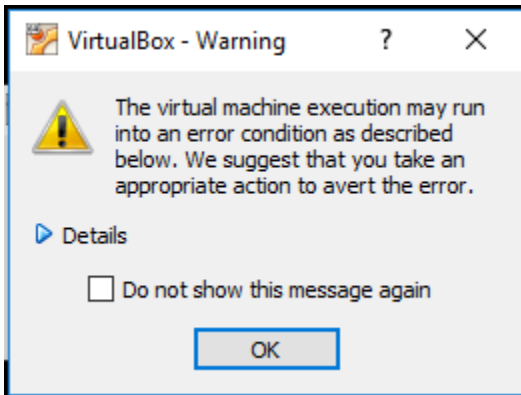
9. You may receive an error about network adapters here similar to the one below.



If so, just click on the option given to “Change Network Settings”. Then click on “Ok” in the network settings. (After the system starts, you can also enable enable “**Auto Ethernet**” in the networking menu (click on double arrows in upper-right corner) but that is probably not necessary).

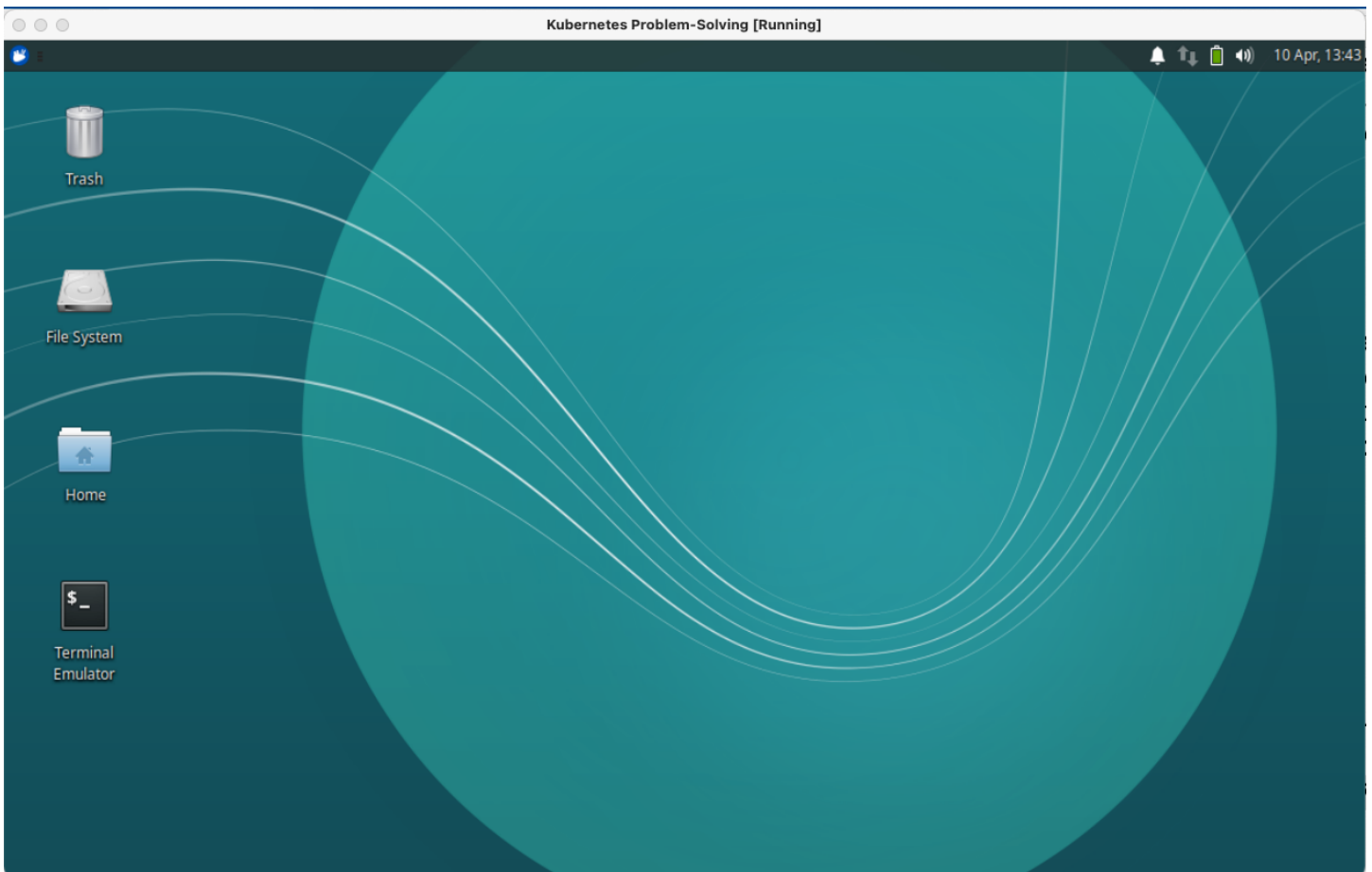


10. If you get a warning dialog that pops up like the one below, it is due to a missing shared directory. (You can see that if you click the Details – not required.) This is not important for using the VM and you can just click OK to proceed and ignore the warning.



11. It may take several minutes for the desktop to appear. If, after several minutes, you don't see the desktop, try switching to full screen mode (Host key + F) or access the menu item for it through the View menu. On most systems, the Host key here will be the right Ctrl key. (Note: Windows 10 seems to have issues if you try to switch to scaled mode. If on a Windows 10 system, you may want to avoid that setting. If you do go into scaled mode and the screen seems to disappear, try using the Host key + F to switch out. Or the menu may still be accessible, although hidden, at the very top of the screen.)

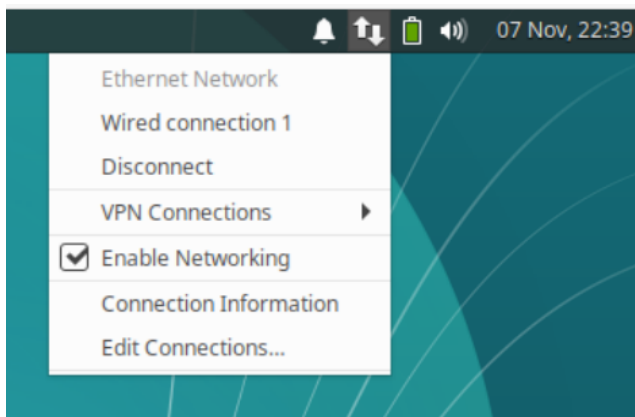
12. After starting up the VM, you should see the desktop of the VM.



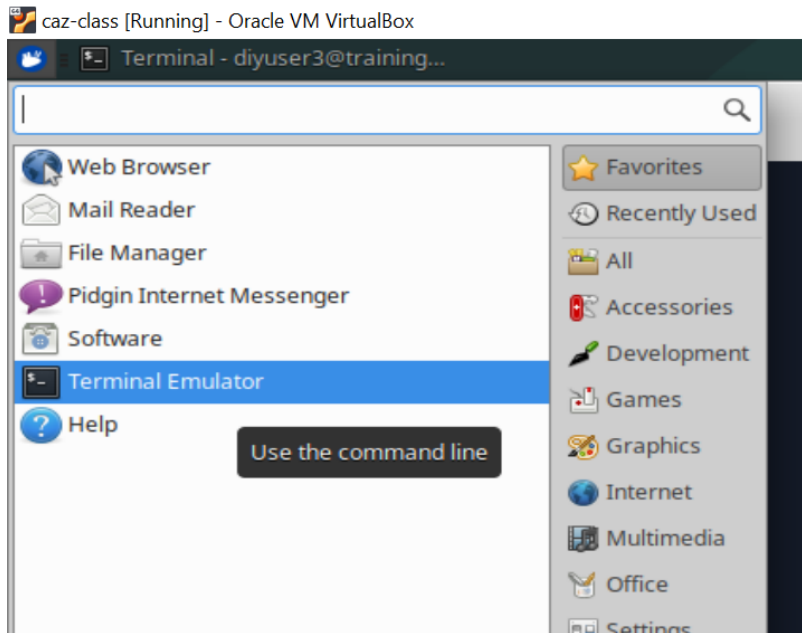
13. If you are on Windows and get a Windows firewall dialog, you can click both boxes and tell Windows to “Allow access”.
14. If you are given an option to upgrade ubuntu, just decline that.
15. If you have messages at the top of the screen about “Auto capture keyboard” and “mouse pointer integration” you can just click the x on the far right of the messages to dismiss those.
16. Verify that you have internet connectivity from the VM. Open up a terminal session from the VM’s desktop and type something like “ping google.com” to make sure you get a response.
17. Login is diyuser3, diyuser3 if you need it.

Startup - to do before first lab

1. **If running in the VM**, enable networking. Enable networking by selecting the up/down arrow icon at top right and selecting the option to "Enable Networking". See screenshot below.



Open a terminal session by using the one on your desktop or clicking on the little mouse icon in the upper left corner and selecting **Terminal Emulator** from the drop-down menu.



2. Get the latest files for the class. For this course, we will be using a main directory *k8s-ps* with subdirectories under it for the various labs.

If running in the VM

In the terminal window, cd into the main directory and update the files.

```
$ cd k8s-ps
$ git pull
```

If NOT running in the VM

```
$ git clone https://github.com/skillrepos/k8s-ps
$ cd k8s-ps
```

3. **Whether running in the VM or not**, pre-pull images we will need for this workshop.

```
$ ./extra/image-prepull.sh
```

4. **If running in the VM**, start up the paused Kubernetes (minikube) instance on this system using a script in the *extras* subdirectory. This will take several minutes to run.

```
$ extra/start-mini.sh
```

5. Enable the Kubernetes metrics-server for the cluster.

If running in the VM

```
$ sudo minikube addons enable metrics-server
```

If NOT running in the VM, consult documentation for your cluster. (Note this is only needed for one step, so is not critical.)

6. Optional - setup alias. In these labs and on the VM, "k" is aliased to "kubectl". If you are not running in the VM, you can usually do this via the following command if you want:

```
$ alias k=kubectl
```