LAB 2

Install Minikube for MacOS

Lab Objectives

Install Minikube Kubernetes cluster for MacOS

Lab Structure - Overview

1. Download and Install Minikube for MacOS
2. Configure kubectl bash completion for bash terminals
3. Start and configure Minikube
4. Enable a Minikube add-on

Lab Overview

Conventions

Lab Guide Conventions

`reboot`	Any text a student needs to enter is printed like this.
`<your.ip>`	Any time a student needs to insert their own value, the text has brackets.
File	User Interface (UI) buttons and objects are bold.
Special Font	Unusual or important words or phrases are marked with italics.

Code Blocks

Blocks of sample code are set apart from the body and marked accordingly. It is recommended that students do not copy/paste text from the lab into their files. Extra formatting is often transferred in this process and can result in failed operations.

```
# ls -l /var/www/html/index.html
-rw-rw-r--  1 root root 1872 Jun 21 09:33 /var/www/html/index.html
# date
Wed Jun 21 09:33:42 EDT 200
```

1. Download and Install Minikube for MacOS

Step by Step Guide

This process will take approximately 10 minutes.

To get Kubernetes installed, Minikube will be used. For this course, the single node cluster that Minikube creates is fine. Additional community hosted Kubernetes clusters can be used for further development outside this course (like Amazon Web Services, Google Cloud Platform).

Step	Action
1.	Open a terminal console (iTerm, Terminal, PowerShell, Ubuntu Bash, Git Bash, etc).

2.	On MacOS, run the following command and it'll automatically pull down the minikube binary, update the mode, and move the binary to /usr/local/bin. Validate the location and version of the binary.
	
	`$ curl -Lo minikube https://storage.googleapis.com/minikube/releases/v0.25.0/minikube-darwin-amd64 && chmod +x minikube && sudo mv minikube /usr/local/bin/`
	
	`$ which minikube`
	`minikube is /usr/local/bin/minikube`
	
	`$ minikube version`
	`minikube version: v0.25.0`

3.	On MacOS, run the following command and it'll automatically pull down the kubectl binary, update the mode, and move the binary to /usr/local/bin. Validate the location and version of the binary.
	
	`$ curl -LO https://storage.googleapis.com/kubernetes-release/release/$(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/darwin/amd64/kubectl`
	
	`$ chmod +x kubectl && mv kubectl /usr/local/bin/`
	
	`$ which kubectl`
	`kubectl is /usr/local/bin/kubectl`
	
	`$ kubectl version`
	`Client Version: version.Info{Major:"1", Minor:"6", GitVersion:"v1.6.1", GitCommit:"b0b7a323cc5a4a2019b2e9520c21c7830b7f708e", GitTreeState:"clean", BuildDate:"2017-04-03T23:37:53Z", GoVersion:"go1.8", Compiler:"gc", Platform:"darwin/amd64"`

Step by Step Guide

This process will take approximately 5 minutes.

Step	Action
1.	Open a terminal console (iTerm, Terminal, PowerShell, Ubuntu Bash, Git Bash, etc)
2.	See the available options for minikube. `$ minikube --help`

3.	On MacOS, run the following command to start minikube in xhyve. Use the --memory and --cpus options to change the default values (memory = 2, cpus = 2).
	`$ minikube start --vm-driver xhyve --memory 4096 --cpus 2`
	
	If there is an error flagged on the availability of the xhyve binary, you may need to install an xhyve driver for your system, as follows:
	
	`/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`
	`$ brew install docker-machine-driver-xhyve`
	`# docker-machine-driver-xhyve need root owner and uid`
	`$ sudo chown root:wheel $(brew --prefix)/opt/docker-machine-driver-xhyve/bin/docker-machine-driver-xhyve`
	`$ sudo chmod u+s $(brew --prefix)/opt/docker-machine-driver-xhyve/bin/docker-machine-driver-xhyve`

4.	On MacOS, run the following command to start minikube in VirualBox. `$ minikube start`

5.	Validate minikube's configured Kubernetes cluster. Type **`kubectl cluster-info`** and press enter. This will take a few minutes to complete configuration. When completed, the Kubernetes cluster information will be displayed.
	`$ kubectl cluster-info`
	`The connection to the server 192.168.64.10:8443 was refused - did you specify the right host or port?`
	
	`$ kubectl cluster-info`
	`Kubernetes master is running at https://192.168.64.10:8443`

6.	Run **`kubectl get nodes`** to the attached nodes. For this cluster, we'll see a single node called minikube.

```
$ kubectl get nodes
NAME       STATUS    AGE       VERSION
minikube   Ready     8m        v1.6.0
```

3. Enable the Heapster Minikube addon

Step by Step Guide

This process will take approximately 5 minutes.

Step	Action
1.	Open a terminal console (iTerm, Terminal, PowerShell, Ubuntu Bash, Git Bash, etc)

2.	Run the following command to get a list of addons for the lab cluster
	`minikube addons list`
	`$ minikube addons list`
	`- kube-dns: enabled`
	`- heapster: disabled`
	`- ingress: disabled`
	`- registry-creds: disabled`
	`- addon-manager: enabled`
	`- dashboard: enabled`
	`- default-storageclass: enabled`

3.	Enable the heapster minikube addon: **`minikube addons enable heapster`**
	`$ minikube addons enable heapster`
	`heapster was successfully enabled`

4.	Run the following command to get a list of addons for the lab cluster
	`minikube addons list`
	`$ minikube addons list`
	`- addon-manager: enabled`
	`- dashboard: enabled`
	`- default-storageclass: enabled`
	`- kube-dns: enabled`
	`- heapster: enabled`
	`- ingress: disabled`
	`- registry-creds: disabled`

5.	Validate the Kubernetes cluster, run **`kubectl cluster-info`**

```
$ kubectl cluster-info
Kubernetes master is running at https://192.168.64....
heapster is running at https://192.168.64.13:8443/a...
KubeDNS is running at https://192.168.64.13:8443/ap...
```

Lab Complete!