**Developers Guide to Velero (Mac OSX version)**

Here's a handy guide to getting started developing with Velero. This guide walks you through required tools and configurations that are needed from a developers standpoint. This guide makes some assumptions on tools (docker, dockerhub, aws, Mac OSX). You are free to swap in the tools of your choosing.

***## Install Prerequisite tools and Configuration***

***# Tool versions used in creating this document***

* Docker 19.03.12
* KIND 0.8.1
* Kubectl 1.18.2
* Mac OSX 10.15.6
* Velero 1.5

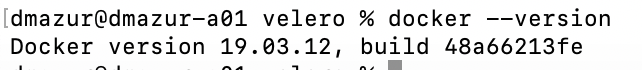
**# Install and/or make sure docker desktop is running on the Mac**

* Install docker desktop if needed
  + <https://www.docker.com/products/docker-desktop>
* Open a terminal session on the mac

A screenshot of a social media post

Description automatically generated

* % Docker --version



* How to check for updates and restart

A screenshot of a cell phone

Description automatically generated

* Go into "Preferences-> Command Line" and enable experimental features
  + This enables "docker buildx" which is experimental in Docker 19.03 but required to build a velero container image later.
  + <https://docs.docker.com/buildx/working-with-buildx/>
  + Select "Apply & Restart" docker

A screenshot of a cell phone

Description automatically generated

* Where to get more information
  + <https://www.docker.com/products/docker-desktop>

**# Install and/or test the Velero client on the Mac**

* In the terminal session
* % Brew install velero
  + Make sure "brew" is installed first
  + <https://docs.brew.sh/Installation>
* % velero version

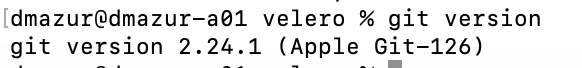
A picture containing knife, table

Description automatically generated

* Brew installs the Velero client at
  + /usr/local/Cellar/velero/1.5.0/bin/velero
  + -and- creates a symlink at
  + /usr/local/bin/velero
  + Why is this useful? If you build a new version of the Velero client and want to replace it later…

**# Setup a Github account**

* <https://docs.github.com/en/github/getting-started-with-github/set-up-git>
  + Follow the steps for installing the client and setting up username, commit email and credentials
* You will be using github from the terminal session
* Install the github client
* % brew install git
* % git version



* *[note] need a command to make sure you can access your github account*

**# Setup a Dockerhub image registry account**

* <https://docs.docker.com/docker-hub/>
* You will use this to push/pull container images from the registry
* In the terminal session
* % docker login –username=yourhubusername –email=youremail
  + Using your dockerhub username/password
* You can also use Quay, Harbor or other image registry

**# Setup an AWS account with an S3 bucket**

* The AWS S3 bucket will be used to store the Velero backup
* [Setup Free AWS account](https://aws.amazon.com/free/?trk=ps_a134p000003yBfsAAE&trkCampaign=acq_paid_search_brand&sc_channel=ps&sc_campaign=acquisition_US&sc_publisher=google&sc_category=core&sc_country=US&sc_geo=NAMER&sc_outcome=acq&sc_detail=%2Baws%20%2Baccount&sc_content=Account_bmm&sc_segment=438195700988&sc_medium=ACQ-P|PS-GO|Brand|Desktop|SU|AWS|Core|US|EN|Text&s_kwcid=AL!4422!3!438195700988!b!!g!!%2Baws%20%2Baccount&ef_id=CjwKCAjwnef6BRAgEiwAgv8mQaByCWXmk6FPCfx6KPwAxEFPGbltgNevNb4f5liGoTSxcEzS4rjkmhoCsesQAvD_BwE:G:s&s_kwcid=AL!4422!3!438195700988!b!!g!!%2Baws%20%2Baccount&all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc)
* Under Services->Storage->S3

A screenshot of a cell phone

Description automatically generated

* Create a bucket which will store the velero backup

A screenshot of a cell phone

Description automatically generated

* In a terminal session on the mac, install the aws cmd line tool
* % brew install awscli
* % aws help
* Next, setup aws credentials
  + This is used to make programmatic calls to aws from the aws cli

A screenshot of a cell phone

Description automatically generated

* Select "My Security Credentials"
* Under "Access Keys" select "Create New Access Key"

A screenshot of a cell phone

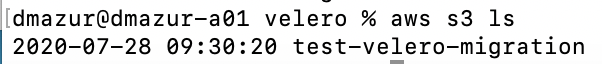
Description automatically generated

* Select "Download Key File" and save in a secure location on your Mac as <name>.csv
* Next use the access key to connect the aws cli on your mac to your aws account
  + % aws configure

A screenshot of a cell phone

Description automatically generated

* Enter your access key id and secret access key from the .csv key file
* Select your aws region
* Enter "text" as the default output format
* Test the aws connection
  + % aws s3 ls
  + You should see the S3 bucket you created

* 

* [Note] You can use another approved cloud provider besides AWS (i.e. Google Cloud, Microsoft Azure, etc)]
* In your terminal session, go to your home dir and check for the .aws credentials directory
  + % cd ~
  + % ls -al

A screenshot of a cell phone

Description automatically generated

**# Install JSON helper utility "jq" on the Mac**

* % brew install jq
* <https://formulae.brew.sh/formula/jq>
* This will be helpful for viewing Velero JSON files

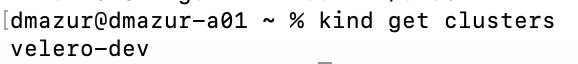
**# Install KIND ( a kinda Kubernetes version) on the Mac**

* Feel free to use your own Kubernetes cluster. This suggests KIND due to simplicity.
* You will create a Kubernetes cluster using KIND and back it up using Velero (later)
* <https://kind.sigs.k8s.io/docs/user/quick-start/>
* % brew install kind
* % kind version

A close up of a clock

Description automatically generated

* Create a cluster for testing the Velero backup
  + % kind create cluster --name=velero-dev
* Check to make sure the cluster exists
  + % kind get clusters



***## Alright! You've made it this far! Let's get started with Velero!!!***

**# Install the Velero backup controller in the cluster**

* Setup environment variables for AWS
* Run the velero install command
* *[Note] What does this really do?????*

A screenshot of text

Description automatically generated

**% Backup your cluster to AWS using Velero**

* In a terminal session on the mac:
* % velero backup create <clusterlevel-v1-09092020>
* This command submits a backup request to the Velero backup controller that you installed in the cluster earlier
* The controller listens for requests and starts the backup when it sees a request
* Check to see if the backup completed

A picture containing bird

Description automatically generated

* You can also go into your AWS S3 bucket and see the backup

A screenshot of a cell phone

Description automatically generated

**# Delete your cluster (Yes, really delete your Kind cluster!)**

* % kind delete cluster --name=velero-dev
* % kind get clusters
  + Velero-dev should be outta here!

**# Restore your cluster from the Velero backup stored in AWS**

* Velero backs up the kubernetes objects in the cluster. You need an empty cluster in order to restore the backup(???)
* You can also delete an object in an existing cluster and use "velero restore" to restore the specific object
* Create an empty cluster
  + % kind create cluster --name=velero-dev
* Install the Velero backup controller image in the cluster
  + % velero install ….
* Restore the cluster contents from backup
  + % Velero restore create --from-backup <xxxx>
* % Kind get clusters

***## Now let's GO dig into the development***

**# Velero plugins and code are developed in Go Lang**

* Install Go Lang
* <https://golang.org/doc/install>

**# Grab the Velero source code**

* % git clone <https://github.com/vmware-tanzu/velero.git>

**# Compile and build the Velero client code (Darwin/AMD64)**

* cd ~/go/src/velero
* % make local
* This creates a “velero” executable in the current directory
* Replace the installed version
  + % sudo mv ./velero /usr/local/bin/velero
* When you run the velero command line, you should now be using your version of the velero client

**[Note: do we need to do this? What is the output?]**

**# Compile and build the Velero binary targeting linux/amd64 within a build container on your local machine**

* Cd ~/go/src/velero
* % make build

**# Build the Velero container image**

* % make container
* *[Note] You should not need Go Lang installed in order to create the container - Check this*
* You should see something like this….

A screenshot of a social media post

Description automatically generated

* Make sure the image has been built
* % docker image ls



**# Push the new Velero image to your Dockerhub registry**

* First, go into docker hub and create a repository for the image

A screenshot of a cell phone

Description automatically generated

* Create a tag using the image id from “docker image ls”
  + % docker tag <image id> <dockerhubid><image name>:<version>
  + % docker tag f67d7a9378bd bikeskinh/velero:v1
* Now, push the tagged image to docker hub
  + % docker push bikeskinh/velero
* Go into docker hub and check that the tag has been pushed

A screenshot of a social media post

Description automatically generated

**# Use your new Velero image in place of the original Velero image you installed in the KIND cluster earlier**

In this example, velero install adds the "--image" flag which uses the tagged image in dockerhub and uses your aws credentials to configure the backup controller

export VERSION=dev-dave-0914

export BUCKET=test-velero-migration

export REGION=us-east-2

# bring your own credentials - see credentials-velero.example for an example

export SECRETFILE=credentials-velero

export IMAGE=hub.docker.com/bikeskinh/velero:$VERSION

velero install \

--provider aws \

--plugins velero/velero-plugin-for-aws:latest \

--bucket $BUCKET \

--prefix $PREFIX \

--backup-location-config region=$REGION \

--snapshot-location-config region=$REGION \

--secret-file $SECRETFILE \

--image $IMAGE

***## Hopefully, this gets you started…enjoy Velero and thank you for your contribution!***