Creating multiple nodes using Script Tools

# **Using Docker-machine**

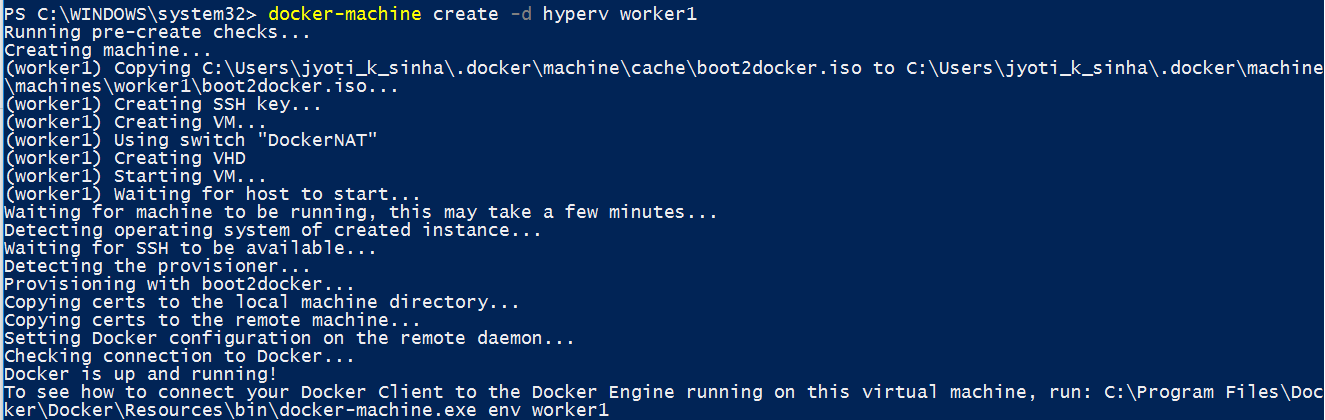
## **On Hyper-V:**

Always use Powershell in “administrator” mode.

### To create a VM named as “manager1” use following command line:

docker-machine create –d hyperv manager1

You can create other nodes or workers using same command by changing the name of VM:

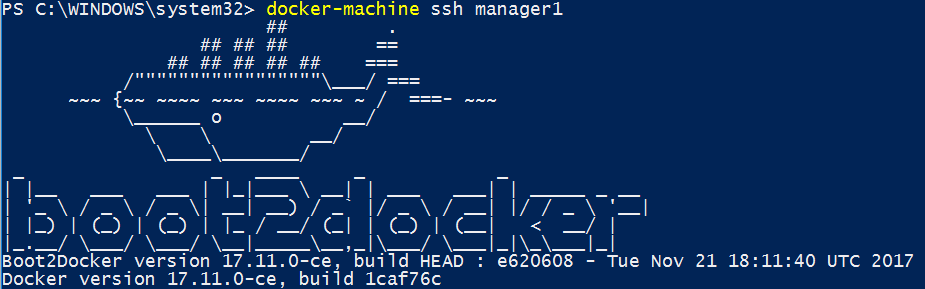


### To connect to the VM created above, use the following command:

docker-machine env worker1

### Using a node

To enter a VM, use the command docker-machine ssh <vm name>



### Removing a VM

To remove a VM use the following command:

docker-machine rm <vm name>

## **On VirtualBox:**

Use virtualbox as the –d parameter value

The command should look like following:

docker-machine create –d virtualbox worker1

rest all the commands remain same as using on hyperV.

# **Using Vagrant**

## Install the Vagrant

Follow the link to get installation instructions for Vagrant setup:

<https://github.com/sinhajyoti/DevOps-Artefacts/blob/master/Steps%20to%20Install%20Vagrant%20on%20Ubuntu.docx>

## Script the VM creation code

Create a text file and name it as *Vagrantfile*.

Following is a sample vagrantfile script for provisioning multiple VMs:

*# -\*- mode: ruby -\*-*

*# vi: set ft=ruby :*

*Vagrant.configure("2") do |config|*

*config.vm.box = "hashicorp/precise64"*

*(1..3).each do |number|*

*config.vm.define "m#{number}" do |node|*

*node.vm.network "private\_network", ip: "172.164.10.1#{number}"*

*node.vm.hostname = "m#{number}"*

*end*

*end*

*config.vm.provider "hyperv" do |v|*

*v.memory = 2048*

*v.cpus = 1*

*end*

*end*

On Powershell prompt, just run following command to spin up 3 ubuntu 16.10 VMs named as m1, m2 and m3 respectively.

vagrant up

In case only two VM(say m2 and m3) is required to be spinned up, use the following command:

vagrant up m2 m3

## Provisioning and configuring Docker using Vagrant

Vagrant provides options for provisioning the add-on commands as well. Let’s use that to embed some additional settings to automate the Docker container creation with specific settings.

### **Create the Docker Provisioning file**

Create a text file and save it as *docker.sh* (shell command) file. This file should also be in same folder as *vagrantfile*.

Docker.sh should look like following:

*#!/bin/bash*

*# Install test version of docker engine, also shell completions*

*curl -fsSL https://test.docker.com/ | sh*

*# Add the vagrant user to the docker group*

*usermod -aG docker vagrant*

*# Configure the docker engine*

*# Daemon options: https://docs.docker.com/engine/reference/commandline/dockerd/*

*# Set both unix socket and tcp to make it easy to connect both locally and remote*

*# You can add TLS for added security (docker-machine does this automagically)*

*cat > /etc/docker/daemon.json <<END*

*{*

*"hosts": [*

*"unix://",*

*"tcp://0.0.0.0:2375"*

*],*

*"experimental": true,*

*"debug": true,*

*"metrics-addr": "0.0.0.0:9323"*

*}*

*END*

*# You can't pass both CLI args and use the daemon.json for parameters,*

*# so I'm using the RPM systemd unit file because it doesn't pass any args*

*# This version changes the following as of 17.03:*

*# - Removes Requires=docker.socket*

*# - Removes docker.socket from After*

*# - Sets LimitNOFILE=infinity*

*# - Removes -H fd:// from ExecStart*

*wget -O /lib/systemd/system/docker.service https://raw.githubusercontent.com/docker/docker/v17.03.0-ce/contrib/init/systemd/docker.service.rpm*

*systemctl daemon-reload*

*systemctl restart docker*

*# optional tools for learning*

*apt-get install -y -q ipvsadm tree*

*# lsns is helpful from util-linux, this is installed already*

### **Create the VMs with Docker configuration**

Let’s create a vagrant script that allows to create 2 Docker Swarm managers and 3 Docker swarm worker nodes. *Vagrantfile* shall look like following:

*# -\*- mode: ruby -\*-*

*# vi: set ft=ruby :*

*Vagrant.configure("2") do |config|*

*config.vm.box = "hashicorp/precise64"*

*config.vm.provision "shell", path: "docker.sh", privileged: true*

*# Managers*

*(1..2).each do |number|*

*config.vm.define "m#{number}" do |node|*

*node.vm.network "private\_network", ip: "162.164.80.1#{number}"*

*node.vm.hostname = "m#{number}"*

*end*

*end*

*# Workers*

*(1..3).each do |number|*

*config.vm.define "w#{number}" do |node|*

*node.vm.network "private\_network", ip: "172.164.80.2#{number}"*

*node.vm.hostname = "w#{number}"*

*end*

*end*

*config.vm.provider "hyperv" do |v|*

*v.memory = 2048*

*v.cpus = 1*

*end*

*end*