Docker Machine, Compose, ar They Work Together

Last updated: 2017-02-17

by Jonas Rosland | 9 Comments

Development

During the past year, Docker has been hard at work creating simple t (Machine), manage multiple containers linked together (Compose), a (Swarm).

Even though they are meant to be simple, these tools are very power you run off and deploy tons of containers on top of your favorite Infra light on why they're great to have in your toolbelt, where they should

> Docker Machine, Compose, and Swar simple, but they do require some plar

The three tools are now neatly packaged into what's called the Docke before you continue further.

Docker Machine

The first tool we'll look at from the toolbox is Docker Machine, which of the most popular Infrastructure-as-a-Service platforms. Of course virtualization platforms: VMware Fusion and VirtualBox, but it also su DigitalOcean, Exoscale, Google Compute Engine, OpenStack, RackSp Air.

Let's start with getting a container host up and running on your local Toolbox you also got VirtualBox installed, so let's use that to create a create a container host, just run the following command:

```
$ docker-machine create --driver virtualbox
```

☎ 855.790.8079

EBOOKS

BLOG

PRESS

DOCUMENTATIONS

CONTACT SALES

LOGIN

WRITE FOR CODESHIP

FREE RESOURCES

DOCKER PLATFORM

FEATURES

PRICING

LOGIN

FREE SIGNUP

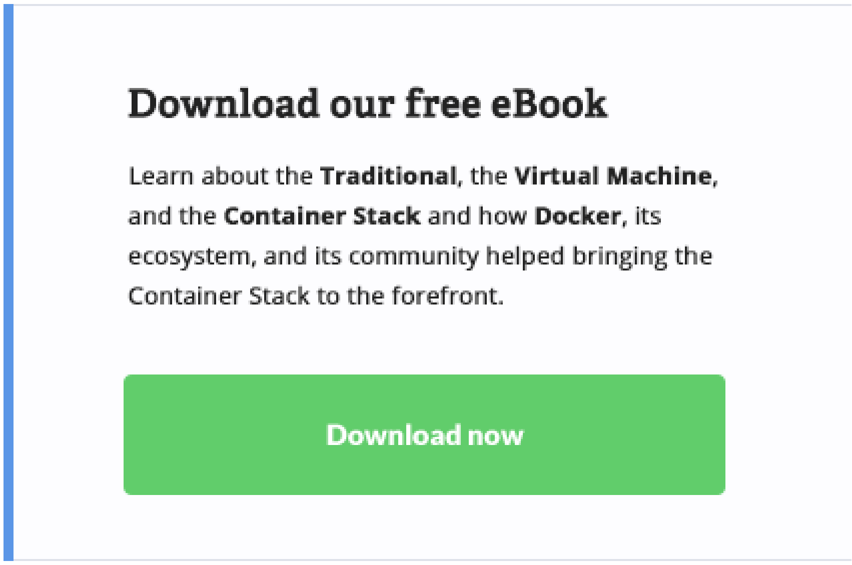
```
$ docker-machine create --driver virtualbox
Creating VirtualBox VM...
Creating SSH key...
Starting VirtualBox VM...
Starting VM...
To see how to connect Docker to this machin
```

This command will tell Docker Machine to use the VirtualBox driver t
“containerhost.” We now have a place to run our containers! Let’s con
the following:

```
eval "$(docker-machine env containerhost)"
```

If you’re used to working with Boot2Docker, the predecessor of Docke
the familiar `boot2docker shellinit` .

After the eval of your Docker Machine env variables, you can now rur
run, pull, ps, rm, etc. Try it out!



Download our free eBook

Learn about the **Traditional**, the **Virtual Machine**, and the **Container Stack** and how **Docker**, its ecosystem, and its community helped bringing the Container Stack to the forefront.

Download now

Docker Compose

Now that we have a container host up and running, we’ll focus more
it. We’ll use Docker Compose for this. It’s actually built on Docker’s fir
company called Orchard that had created a multi-container manage
joined team from Orchard continued the great work that Orchard had
Compose late last year.

☎ 855.790.8079

EBOOKS
BLOG
PRESS
DOCUMENTATIONS
CONTACT SALES
LOGIN

WRITE FOR CODESHIP
FREE RESOURCES
DOCKER PLATFORM
FEATURES
PRICING
LOGIN
FREE SIGNUP

Docker Compose has a simple way of describing an application as se should be linked, and what ports should be exposed to the end user. defined in a "docker-compose.yml" file; let's look at an example and b

```
web:
  build: .
  ports:
    - "5000:5000"
  volumes:
    - .:/code
  links:
    - redis
redis:
  image: redis
```

Here we have one application built using two containers. The first cor Dockerfile that we have in the current working directory. This is great container image but haven't pushed it up to a registry yet.

The next row shows which ports will be exposed on the host and whi container. The third part shows that we will mount a Docker volume i code. Then lastly we will link to another container that we call "redis," image from the Docker Hub.

Now to run this, we issue the following command:

```
$ docker-compose up
```

This will read the docker-compose.yml and create the application en the web application from the Dockerfile as well as pulling down the r web container to the redis container. It will look something like this:

```
$ docker-compose up
Pulling redis (redis:latest)...
<snip>
Creating compose_redis_1...
Building web...
Step 0 : FROM python:2.7
2.7: Pulling from python
<snip>
Successfully built b88dd767cf97
Creating compose_web_1...
Attaching to compose_redis_1, compose_web_1
<snip>
redis_1 | 1:M 11 Sep 21:21:56.463 # Server
<snip>
```

☎ 855.790.8079

EBOOKS

BLOG

PRESS

DOCUMENTATIONS

CONTACT SALES

LOGIN

WRITE FOR CODESHIP

FREE RESOURCES

DOCKER PLATFORM

FEATURES

PRICING

LOGIN

FREE SIGNUP

```
web_1 | * Running on http://0.0.0.0:5000/ (
web_1 | * Restarting with stat
```

☎ 855.790.8079

EBOOKS

BLOG

PRESS

DOCUMENTATIONS

CONTACT SALES

LOGIN

WRITE FOR CODESHIP

FREE RESOURCES

DOCKER PLATFORM

FEATURES

PRICING

LOGIN

FREE SIGNUP

There you have it! You have successfully used Docker Compose to cr services: one an outward-facing web service and the other a persiste page to verify the application is running.

To find the IP of your container host, run the following command:

```
$ docker-machine ip containerhost
192.168.99.101
```

Since we have also made sure we open up port 5000 on the contair now be able to connect to http://192.168.99.101:5000. You should se seen 1 times." You are now hitting a website that's storing a hit counte store, retrieving that value on each new hit on the website and preser

All right, now that we've explained the basics of Docker Compose, it's

Docker Swarm

Finally, let's look at the most interesting tool in the current Docker Toc is work with one container host and run a container or two, which is g Docker Swarm we're now going to turn that small test environment ir that can be used to scale your operations into something even more involve things like service discovery, clustering, and remote managen

Let's start by cleaning up the environment we have so we don't run ir running. Stop and remove the current local container host by running

```
$ docker-machine stop containerhost
exit status 1
$ docker-machine rm containerhost
Successfully removed containerhost
```

Now let's begin by creating a new fresh container host that we will us

```
$ docker-machine create -d virtualbox local
```

This creates a new container host for us called "local." Get the right c running:

```
$ eval "$(docker-machine env local)"
```

```
eval "$(docker-machine env local)"
```

Now we need to generate what's called a "discovery token" that will b
nodes are part of the correct cluster:

```
$ docker run swarm create
<snip>
Status: Downloaded newer image for swarm:la
8d7dc66346a3e0d999ed38dd29ed0d38
```

That last line is your discovery token and will be different from mine.
Docker's public discovery service. You can find the information it keep
https://discovery.hub.docker.com/v1/clusters/YOURTOKENHERE. Yo
members including the Swarm Master that we create like this:

```
$ docker-machine create -d virtualbox --swa
```

Now we'll create our two first Swarm nodes:

```
er-machine create -d virtualbox --swarm --swarm-c
er-machine create -d virtualbox --swarm --swarm-c
```

You can now also shut down and remove the "local" container host; w

Now let's make sure your shell is pointing to the Swarm Master:

```
eval $(docker-machine env --swarm swarm-mas
```

You now have a Swarm Master and two Swarm Nodes running locall

```
$ docker info
Containers: 4
Images: 3
Role: primary
Strategy: spread
Filters: affinity, health, constraint, port
Nodes: 3
  swarm-agent-00: 192.168.99.104:2376
    └ Containers: 1
    └ Reserved CPUs: 0 / 1
    └ Reserved Memory: 0 B / 1.022 GiB
    └ Labels: executiondriver=native-0.2, k
  swarm-agent-01: 192.168.99.105:2376
    └ Containers: 1
    └ Reserved CPUs: 0 / 1
```

☎ 855.790.8079

EBOOKS

BLOG

PRESS

DOCUMENTATIONS

CONTACT SALES

LOGIN

WRITE FOR CODESHIP

FREE RESOURCES

DOCKER PLATFORM

FEATURES

PRICING

LOGIN

FREE SIGNUP

```
      └ Reserved Memory: 0 B / 1.022 GiB
      └ Labels: executiondriver=native-0.2, k
  swarm-master: 192.168.99.103:2376
      └ Containers: 2
      └ Reserved CPUs: 0 / 1
      └ Reserved Memory: 0 B / 1.022 GiB
      └ Labels: executiondriver=native-0.2, k
CPUs: 3
Total Memory: 3.065 GiB
Name: 054cb8519400
```

That's really cool! You now have full control over a cluster of containe think that's really fantastic!

You can now try to run containers just like normal. Sometimes it take unique ID of the container, so just wait until it comes back:

```
$ docker run -d redis 0d7af2492be35cc9c7593
$ docker run -d nginx 0babf055abf9b487b6baf
```

And now list the containers to see that they're being scheduled on dif given names:

```
$ docker ps
CONTAINER ID    IMAGE     COMMAND
0babf055abf9    nginx     "nginx -g 'daemon
0d7af2492be3    redis     "/entrypoint.sh r
```

Awesome job! You now have a cluster of container hosts that you cor across.

Unfortunately, the integration between Docker Compose and Docker being done to have them properly compatible in the near future; you

Until then, have fun coming up with interesting applications that can

PS: If you liked this article you can also download it as an eBook PDF Compose and Swarm

> “Docker Machine, Compose, and Swai
> Together” – via @codeship

☎ 855.790.8079

EBOOKS

BLOG

PRESS

DOCUMENTATIONS

CONTACT SALES

LOGIN

WRITE FOR CODESHIP

FREE RESOURCES

DOCKER PLATFORM

FEATURES

PRICING

LOGIN

FREE SIGNUP

Subscribe via Email

Over 60,000 people from companies like Netflix, Apple, Spotify and O'Reilly are reading our articles.
Subscribe to receive a weekly newsletter with articles around Continuous Integration, Docker, and software development best practices.

We promise that we won't spam you. You can unsubscribe any time.

First Nan

Last Nam

Email*

Join the Discus

Leave us some comments on what you
or if you like to add some

Thomas

Great Article!!! Do you know how i set a static ip to boc
VBox because Hyper-V hasn't the ability to set ip only
Thank's

nanekratzke

Really nice post. I added it to valuable docker links.
http://www.nkode.io/2014/08/24/valuable-docker-lir

Manuel Weiss @codeship

Very cool, thanks!

Nissan Dookeran

Thanks for this. The Dockerfile link in the section that
Compose leads to a dead link though.

Brandon Stiles

Brilliant. Thanks for the info!

Pingback: DevOps: OpenStack + Docker Swarm | 1 2

thusspokez

Thanks, the best post I have read about Docker tools a
the practical examples.

Bernard Łabno

Well done, this article helped me a lot! Thank you Jona
You could mention about IP problems when you restar
may be advertising addresses from previous machine
different IP).
This problem has not been resolved yet nicely apart fr
containers of swarm agents and masters.

Davide Bertola

Something i don't understand. If I do `docker-machine
get "Error response from daemon: This node is not a s
number of other commands (docker swarm *, docker
deploy swarm nodes using –swarm –swarm-master c

☎ 855.790.8079

EBOOKS

BLOG

PRESS

DOCUMENTATIONS

CONTACT SALES

LOGIN

WRITE FOR CODESHIP

FREE RESOURCES

DOCKER PLATFORM

FEATURES

PRICING

LOGIN

FREE SIGNUP

☎ 855.790.8079

PLATFORM

Docker Continuous Integration

Features

ParallelCI

COMPANY

Team

Customers

Careers

Press

GETTING STARTED

Pricing

Contact sales

Request a Codeship Docker trial

Get Codeship Swag

HELP

Documentation

Community

Contact Support

CASE STUDIES

Product Hunt

Bannerman

Lending Crowd

LEGAL

EBOOKS

BLOG

PRESS

DOCUMENTATIONS

CONTACT SALES

LOGIN

WRITE FOR CODESHIP

FREE RESOURCES

DOCKER PLATFORM

FEATURES

PRICING

LOGIN

FREE SIGNUP

Terms of Service

Privacy

Security

Imprint

☎ 855.790.8079

☎ 855.790.8079 Twitter

EBOOKS

BLOG

PRESS

DOCUMENTATIONS

CONTACT SALES

LOGIN

WRITE FOR CODESHIP

FREE RESOURCES

DOCKER PLATFORM

FEATURES

PRICING

LOGIN

FREE SIGNUP