## Docker + Vagrant

\*Refer to appendix for diagram

**1.** Create a Dockerfile (include configuration) for the docker container (needed throughout the whole process)

**vim Dockerfile**



**2.** Type in the file (created in step 1)

**type in the following…**



FROM ubuntu:14.04 is the base image that we are building on. There are more base images that are available on docker hub.

The following lines describe modifications that will be made on top of the base image:

* Install development tools using apt-get

1. openjdk
2. git
3. wget

* Git Clone and install vertx
* Add vertx bin folder to the specified path
* Make a new folder /usr/local/src and use it as the default working directory

**3.** Create a Vagrantfile for the vagrant automation (setup and install docker container)

**vim Vagrantfile**



**4.** Type in the file (created in step 3)

vertxreceiver = get request from vertxsender (react server)

vertxsend = send request to vertxreceiver (client)

**type in the following…**



This file configures the DEFAULT\_PROVIDER to “docker”, HOST\_NAME of docker to “dockerhost”, HOST\_VAGRANTFILE to “./DockerHostVagrantfile” and define the containers settings and commands to run.

Example of some of the settings and commands…

* Set the build directory to “.” (use current working directory to build Vagrantfile)
* Build args for Docker is set to “-t=vertxreceiver” (set name of container)
* Set container to remain running to “true” (do not exit unless specified to)
* Specify a command to point the machine as “receiver” or “sender”
* Set volumes to “/src/verts/:/usr/local/src” (pointing guest directory to host directory to get files)
* Set vagrant machine and vagrantfile as “DOCKER\_HOST\_NAME and DOCKER\_HOST\_VAGRANTFILE”
* Specify “end” as last line for that specific config definition

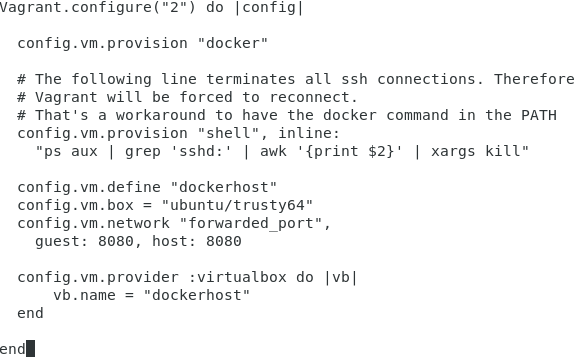
**5.** Create a DockerHostVagrantfile for specifying custom settings for the docker host

**vim DockerHostVagrantfile**



**6.** Type in the file (created in step 5)

**type in the following…**



This file contains the configurations of custom settings for the individual containers.

Example of some of the settings and commands…

* Kill all ssh connections
* Set provision to “docker” (run Docker as Vagrant provisioner)
* Define name to “dockerhost”
* Get base box of “ubuntu/trusty64”
* Configure port-forwarding of “guest: 8080 to host: 8080”
* Run “dockerhost” in VirtualBox

**7.** Start vagrant and let it run

**vagrant up**

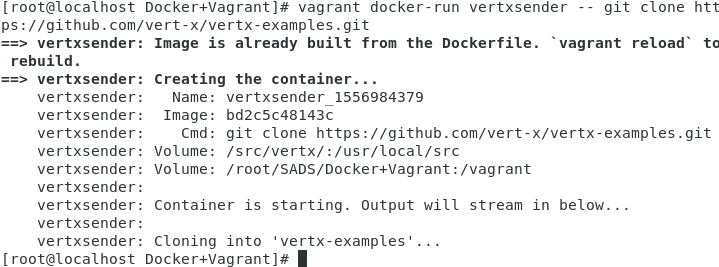




**8.** Download vertx example (vertx is a react toolkit) from GitHub

vertxsender is the machine name that was defined (step 4)

**vagrant docker-run vertxsender – git clone** [**https://github.com/vert-x/vertx-examples.git**](https://github.com/vert-x/vertx-examples.git)

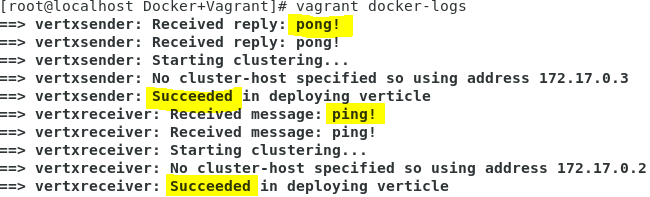


**9.** Reload vagrant to update changes

**vagrant reload**

**10.** Issue log command to check the status and events of container (ping & pong prove that it is successful)

**vagrant docker-logs**



**11.** Create a shell script (provision.sh) that will house the commands that is going to be used in step 13

**vim provision.sh**



**12.** Stop Vagrant

**vagrant halt**

**13.** Use vim to enter Vagrantfile

**vim Vagrantfile**

**14.** Type in the Vagrantfile (step 13) before the last “end”

**type in the following…**



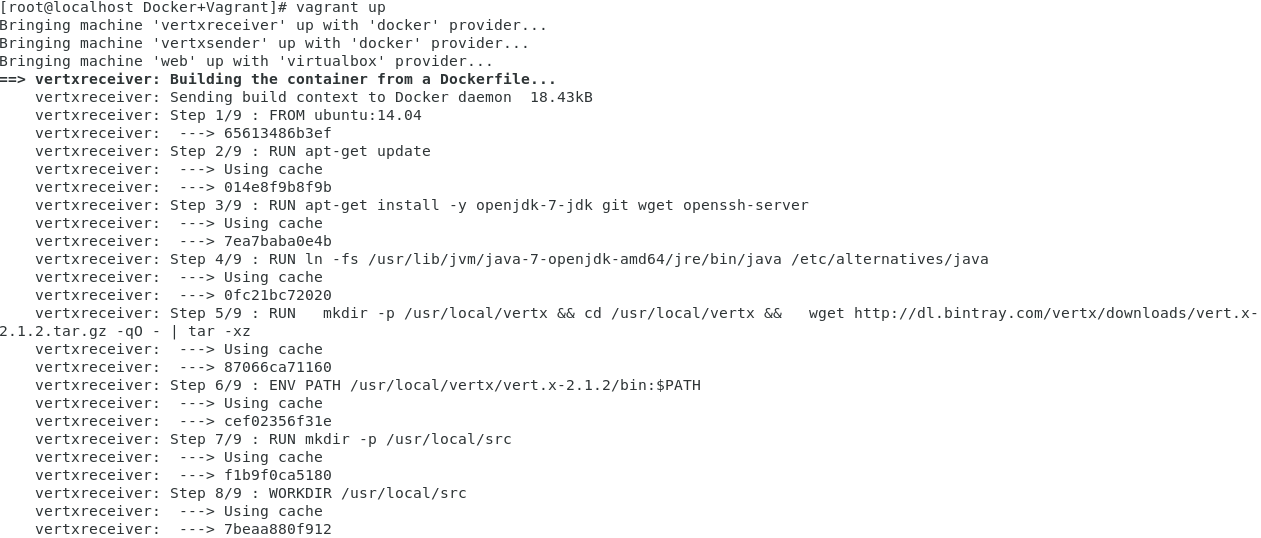
These configurations add a new container through “VirtualBox” provider.

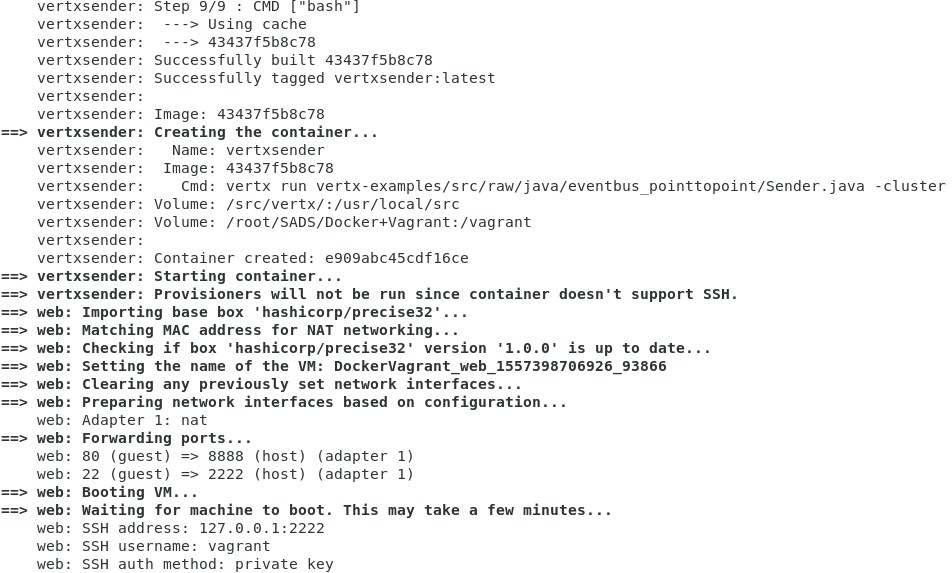
Example of some of the configurations…

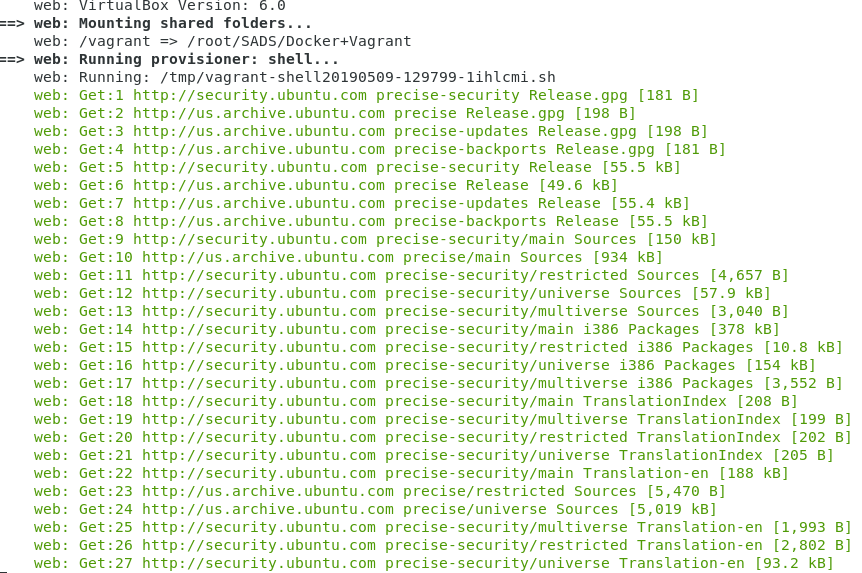
* Create a new container with the name “web”
* Use base image of “hashicorp/precise32”
* Run “provision.sh” (created in step 11) in container
* Set port mapping of guest: 80 to host: 8080 with the id of “nginx”

**15.** Start the automation through Vagrant

**vagrant up**



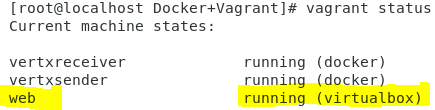






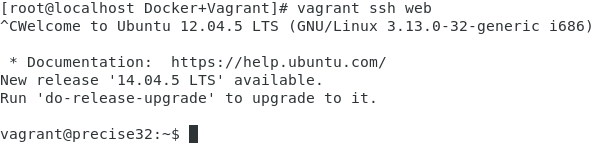
**16.** Check the status of the Vagrant events

**vagrant status**



**17.** Test nginx server through ssh

**vagrant ssh web**



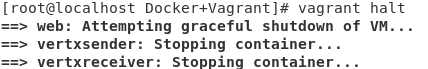
**18.** Test nginx server through host’s firefox

**open mozilla firefox on host machine and go to localhost:8888**



**19.** Turn off the all containers

**vagrant halt**



# Appendices

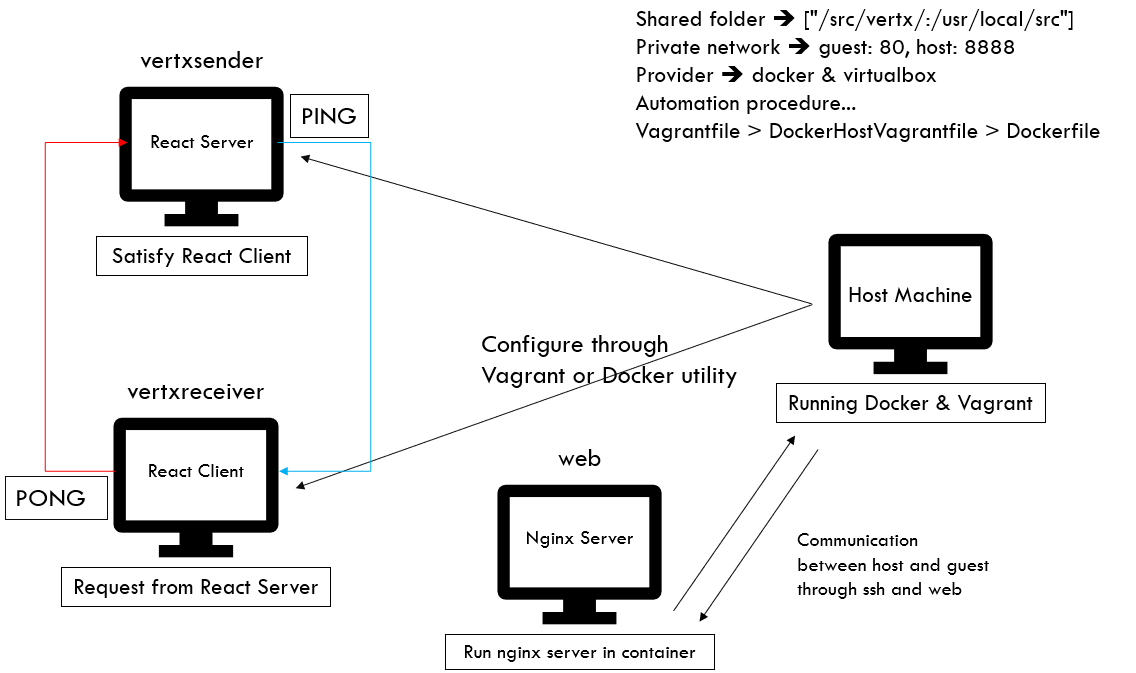


Diagram to show how Vagrant works with Docker