[Deploying Wordpress micro-services with Docker containers on Vagrant box via Ansible](http://www.bogotobogo.com/DevOps/Ansible/Docker-WordPress-Microservices-with-Nginx-reverse-proxy-Varnish-Mysql-Deployed-via-Ansible.php)

Note

This tutorial is based on [Wordpress on Docker the CORRECT way - part 1 locally with Vagrant](https://www.youtube.com/watch?v=vFPOv3zn7eY&index=1&list=PLCxhJ1rikyTpo4Ji_fhWuidxuiB5dywNk%22" \t "_blank).

So, it's not definitely my work, and I just wanted to do it by myself!

Since it is tightly linked to a specific blog app, it's awfully complicated (well, that's because there are lots of things going on).

However, I think writing this tutorial will help me to understand how Ansible orchestrates Docker containers at least for Dev environment.

As the first try, we'll use Vagrant as a local server where containers with micro services are deployed.

Picture source : [WordPress and Docker the correct way](https://ejosh.co/de/2015/08/wordpress-and-docker-the-correct-way/" \t "_blank)

Source code

Get the source from [here](https://github.com/johanan/Dockerized-Wordpress-the-Correct-way.git).

**.env**:

export MYSQL\_ROOT\_PASSWORD=mysql-root-password

export MYSQL\_USER=mysql-user

export MYSQL\_PASSWORD=mysql-password

export MYSQL\_DATABASE=mysql-database

export NEWUSER=newuser

export SSH\_KEY=~/.ssh/id\_rsa.pub

**vagrant/Vagrantfile**:

Vagrant.configure(2) do |config|

config.vm.box = "ubuntu/trusty64"

config.vm.network "forwarded\_port", guest: 80, host: 8888

config.vm.network "forwarded\_port", guest: 443, host: 8443

config.vm.synced\_folder "./vmwordpress", "/var/www/html"

config.vm.provision "ansible" do |ansible|

ansible.playbook = "../ansible/server\_setup.yml"

ansible.sudo = true

ansible.extra\_vars = { ansible\_ssh\_user: 'vagrant' }

ansible.host\_key\_checking = false

end

end

We'll use Ubuntu 14 for the Vagrant OS, and we'll use port forwarding (8888(local)-80(Vagrant)).

Also, we will use **/var/www/html** of Vagrant to store **./vmwordpress**.

**vagrant/development\_build.sh**:

#!/bin/bash

source ../.env

rm -rf ./vmwordpress/\*

vagrant up

export ANSIBLE\_HOST\_KEY\_CHECKING=False

ansible-playbook --user=vagrant --inventory-file=./.vagrant/provisioners/ansible/inventory/vagrant\_ansible\_inventory ../ansible/new\_wordpress\_setup.yml --extra-vars "site\_name=$1 wordpress\_version=$2 edit=True"

sudo ssh -p 2222 -gNfL 80:localhost:80 vagrant@localhost -i .vagrant/machines/default/virtualbox/private\_key -o UserKnownHostsFile=/dev/null -o StrictHostKeyChecking=no

The core of this tutorial is the line of "ansible-playbook":

1. The user is **vagrant**.
2. The inventory file will be created after the vagrant box is created.
3. Then, we'll run **ansible/new\_wordpress\_setup.yml**.
4. We're going to pass in two variables via command line: FQDN and WP version (4.5.3).

The next line is to ssh into the vagrant box.

Playbook

**ansible/new\_wordpress\_setup.yml>**:

- hosts: all

remote\_user: "{{ newuser }}"

sudo: yes

vars\_files:

- vars.yml

roles:

- docker

- swap

- { role: docker\_build, service: mysql, tags: [], exec\_files: ['backup.sh', 'load\_db.sh'] }

- { role: docker\_build, service: php, tags: [], exec\_files: [] }

- { role: docker\_build, service: backend, tags: ['backend'], exec\_files: ['site-normal', 'site-upgrade', 'site-upgrade', 'start-nginx.sh'] }

- { role: docker\_build, service: varnish, tags: [], exec\_files: ['start-varnish.sh'] }

- { role: docker\_build, service: frontend, tags: [], exec\_files: ['start-nginx.sh'] }

- wordpress

- new\_wordpress

tasks:

- include: tasks/docker-compose\_rebuild.yml service=data tags=[]

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=[]

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=['backend']

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=['backend']

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=[]

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=[]

#remove the workpress install not in the container

#bring up the containers and remove any unused containers

- { include: tasks/docker\_up.yml }

- name: Set permissions in container

shell: "{{ work\_dir }}/site-normal"

We used **all** for hosts and for variables it will go to **ansible/vars.yml**:

newuser: "{{ lookup('env', 'NEWUSER') }}"

work\_dir: /var/lib/{{ site\_name }}

ssh\_key: "{{ lookup('env', 'SSH\_KEY') }}"

It gets the 'NEWUSER' and 'SSH\_KEY' from **.env** file. The "site\_name" comes from the args in the command line when we run **vagrant/development\_build.sh** in the next section.

Let's take a look at the "tasks" in **ansible/roles/docker/tasks/main.yml**:

---

- name: Add repo from Docker

shell: curl -sSL https://get.docker.com/ | sudo sh creates=/usr/bin/docker

- name: Install docker

apt: name=docker-engine update-cache=yes

- name: get setuptools

apt: name=python-setuptools state=present

- name: make sure pip works

shell: easy\_install -U pip creates=/usr/local/bin/pip

- name: install docker-py

pip: name=docker-py state=present

- name: install docker-compose

pip: name=docker-compose state=present

- name: create the logrotate conf for docker

copy: src=logrotate\_docker dest=/etc/logrotate.d/docker

We want to make sure **docker** installed, **docker-compose** is there and **docker logrotate** in ready. Basically, this setup gets docker in our box.

Also, we can see we have **swap** tasks are there as well.

Note that we set a link to the **docker** folder:

ansible/roles/docker\_build$ ls -la

total 12

drwxrwxr-x 3 k k 4096 Aug 14 09:50 .

drwxrwxr-x 9 k k 4096 Aug 14 09:50 ..

lrwxrwxrwx 1 k k 16 Aug 14 09:50 files -> ../../../docker/

drwxrwxr-x 2 k k 4096 Aug 14 09:50 tasks

In fact, we want to get all things in the **docker** into our box.

Here is the **ansible/roles/docker\_build/tasks/main.yml**:

---

- name: Create {{ service }} directory

file: path={{ work\_dir }} state=directory owner=root group=root

- name: Load {{ service }}

copy: src={{ service }} dest={{ work\_dir }}

- name: Make files executable

file: path={{ work\_dir }}/{{ service }}/{{ item }} mode=755

with\_items:

- "{{ exec\_files }}"

The **main.yml** get the directory passed as a **service**. For example, with **mysql** service, it looks for **mysql** folder (**docker/mysql**) and put it onto the box.

The **ansible/roles/wordpress** does preparation and checks if we're ready before we actually get download wordpress stuff in **ansible/roles/new\_wordpress**.

Run

At this point, we can run the script (**vagrant/development\_build.sh**). For future reference, I'll list most of the output from the run:

(venv)vagrant$ ./development\_build.sh bogo-wordpress.com 4.5.3

Bringing machine 'default' up with 'virtualbox' provider...

==> default: Checking if box 'ubuntu/trusty64' is up to date...

==> default: A newer version of the box 'ubuntu/trusty64' is available! You currently

==> default: have version '20151008.0.0'. The latest is version '20160802.0.1'. Run

==> default: `vagrant box update` to update.

==> default: Machine already provisioned. Run `vagrant provision` or use the `--provision`

...

PLAY [all] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TASK [setup] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker : Add repo from Docker] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker : Install docker] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker : get setuptools] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker : make sure pip works] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker : install docker-py] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker : install docker-compose] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker : create the logrotate conf for docker] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [swap : register swap var] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [swap : create the file to be used for swap] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

skipping: [default]

TASK [swap : format the file for swap] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

skipping: [default]

TASK [swap : change swap file permissions] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

skipping: [default]

TASK [swap : add the file to the system as a swap file] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

skipping: [default]

TASK [docker\_build : Create mysql directory] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Load mysql] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Make files executable] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default] => (item=backup.sh)

ok: [default] => (item=load\_db.sh)

TASK [docker\_build : Create php directory] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Load php] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Make files executable] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TASK [docker\_build : Create backend directory] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Load backend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Make files executable] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default] => (item=site-normal)

ok: [default] => (item=site-upgrade)

ok: [default] => (item=site-upgrade)

ok: [default] => (item=start-nginx.sh)

TASK [docker\_build : Create varnish directory] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Load varnish] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Make files executable] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default] => (item=start-varnish.sh)

TASK [docker\_build : Create frontend directory] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Load frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [docker\_build : Make files executable] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default] => (item=start-nginx.sh)

TASK [wordpress : Create Docker base directory] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Copy docker-compose file] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Setup ufw] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Open up SSL ufw] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Create backup script] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Create correct default.templ for nginx] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Add site-upgrade] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Add site-normal] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Make sure data directory exists] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Copy over the data Dockerfile] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : schedule backup to run weekly] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [wordpress : Copy move SQL] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [new\_wordpress : Make sure httplib2 is installed] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [new\_wordpress : Get salt values] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [new\_wordpress : Set salts] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [new\_wordpress : Make sure data directory exists] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [default]

TASK [new\_wordpress : Get wordpress] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [new\_wordpress : Unarchive wordpress] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [new\_wordpress : Remove wordpress zip] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [new\_wordpress : Create a wp-config] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [new\_wordpress : Set permissions] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [include] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[DEPRECATION WARNING]: You should not specify tags in the include parameters. All tags should be specified using the task-

level option.

This feature will be removed in a future release. Deprecation warnings can be disabled by setting

deprecation\_warnings=False in ansible.cfg.

included: /home/k/TEST/VAGRANT/Dockerized-Wordpress-the-Correct-way/ansible/tasks/docker-compose\_rebuild.yml for default

TASK [Stop data] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Remove data] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Build data] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [include] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

included: /home/k/TEST/VAGRANT/Dockerized-Wordpress-the-Correct-way/ansible/tasks/docker-compose\_rebuild.yml for default

TASK [Stop frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Remove frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Build frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [include] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

included: /home/k/TEST/VAGRANT/Dockerized-Wordpress-the-Correct-way/ansible/tasks/docker-compose\_rebuild.yml for default

TASK [Stop frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Remove frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Build frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [include] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

included: /home/k/TEST/VAGRANT/Dockerized-Wordpress-the-Correct-way/ansible/tasks/docker-compose\_rebuild.yml for default

TASK [Stop frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Remove frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Build frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [include] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

included: /home/k/TEST/VAGRANT/Dockerized-Wordpress-the-Correct-way/ansible/tasks/docker-compose\_rebuild.yml for default

TASK [Stop frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Remove frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Build frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [include] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

included: /home/k/TEST/VAGRANT/Dockerized-Wordpress-the-Correct-way/ansible/tasks/docker-compose\_rebuild.yml for default

TASK [Stop frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Remove frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Build frontend] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [include] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

included: /home/k/TEST/VAGRANT/Dockerized-Wordpress-the-Correct-way/ansible/tasks/docker\_up.yml for default

TASK [Copy wordpress for editing] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Remove Wordpress] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [bring it all up] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [default]

TASK [Clean all unused images] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

fatal: [default]: FAILED! => {"changed": true, "cmd": ["docker", "images", "-q", "-f", "dangling=true", "|", "xargs", "docker", "rmi"], "delta": "0:00:03.505673", "end": "2016-08-15 01:08:16.974570", "failed": true, "failed\_when\_result": true, "rc": 1, "start": "2016-08-15 01:08:13.468897", "stderr": "\"docker images\" requires at most 1 argument(s).\nSee 'docker images --help'.\n\nUsage: docker images [OPTIONS] [REPOSITORY[:TAG]]\n\nList images", "stdout": "", "stdout\_lines": [], "warnings": []}

to retry, use: --limit @../ansible/new\_wordpress\_setup.retry

PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

default : ok=72 changed=26 unreachable=0 failed=1

[sudo] password for k:

Warning: Permanently added '[localhost]:2222' (ECDSA) to the list of known hosts.

To make the run successful, I had to slightly modify the **tasks** portion of our playbook, **ansible/new\_wordpress\_setup.yml**:

...

tasks:

- include: tasks/docker-compose\_rebuild.yml service=data tags=[]

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=[]

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=['backend']

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=['backend']

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=[]

- include: tasks/docker-compose\_rebuild.yml service=frontend tags=[]

...

All other scripts are working fine.

As we can see from the output, most of the Ansible tasks were successful except the "Unused image clean-up" and "PLAY RECAP".

To make the dns work, we should add a line to **/etc/hosts** of our local machine which hosts Vagrant like this:

127.0.0.1 bogo-wordpress.com

On a local machine, we can access the WordPress site: