

A simple bash script to provision a virtual host for a wordpress site, called within a Vagrant build.

```
#!/bin/bash

# Install wp-cli
curl -O
https://raw.githubusercontent.com/wp-cli/builds/gh-pages/phar/wp-cli.p
har
php wp-cli.phar --info
chmod +x wp-cli.phar
sudo mv wp-cli.phar /usr/local/bin/wp
wp --info

# Setup Apache to properly rewrite the WordPress www subdomain
sudo cp /var/www/myapp/provisioning/myapp.conf.vagrant
/etc/apache2/sites-available/myapp.conf

sudo wp core config --dbname=myapp --dbuser=root --allow-root

if ! mysql -u root -e 'use myapp'; then
    sudo wp db create --prompt --quiet --allow-root
fi

sudo wp core install --url="www.myapp.dev" --title="Invent Value" -
-admin_user="admin" --admin_password="myapp" --
admin_email="admin@myapp.dev" --allow-root

# Install the default theme (Sage)
# @see: https://github.com/roots/sage
#wp theme install https://github.com/roots/sage/archive/master.zip --
```

```
activate
wp theme enable myapp --activate
```

```
# Restart Apache and enable the vhost
sudo a2ensite myapp
sudo service apache2 restart
```

Converting (and expanding) this to a playbook:

```
# vim:ft=ansible:
#
```

```
---
```

```
- hosts: all
  become: yes
```

```
tasks:
```

```
  - name: install wp-cli
    shell: curl -0
```

```
https://raw.githubusercontent.com/wp-cli/builds/gh-pages/phar/wp-cli.p
har
```

```
    args:
```

```
      creates: /usr/local/bin/wp
```

```
  - name: config wp-cli
```

```
    shell: php wp-cli.phar --info --allow-root
```

```
    args:
```

```
      creates: /usr/local/bin/wp
```

```
  - name: make wp-cli writable
```

```
shell: chmod +x wp-cli.phar
args:
  creates: /usr/local/bin/wp

- name: move wp-cli to local bin
  shell: sudo mv wp-cli.phar /usr/local/bin/wp
  args:
    creates: /usr/local/bin/wp

- name: setup wordpress vhost config
  copy:
    src=myapp.conf
    dest=/etc/apache2/sites-available/myapp.conf
    owner=root
    mode=0644
    force=no

- name: set ServerName for wordpress vhost
  replace:
    dest=/etc/apache2/sites-available/myapp.conf
    regexp="{HOSTNAME}"
    replace="www.{{hostname}}"
    backup=no

- name: set webroot for apache vhost
  replace:
    dest=/etc/apache2/sites-available/myapp.conf
    regexp="{WEBROOT}"
    replace="/var/www/myapp/www"
    backup=no
```

```

- name: enable the myapp virtual host
  shell: sudo a2ensite wordpress

- name: create 'myapp' database if it doesn't exist
  shell: mysql -uroot -e "CREATE DATABASE IF NOT EXISTS myapp"

- name: setup wp-config.php
  shell: sudo wp core config --path=/var/www/myapp/www --
dbname=myapp --dbuser=root --allow-root
  args:
    creates: /var/www/myapp/www/wp-config.php

- name: add WP_ENV and APP_URL to wp-config
  lineinfile:
    dest=/var/www/myapp/www/wp-config.php
    regexp='^WP_ENV'
    line='\n\ndefine("WP_ENV",
"development");\ndefine("APP_URL","{{hostname}}");'
    insertafter='<\?php'
    state=absent
  notify: restart apache

- name: install wp build
  shell: sudo wp core install --path=/var/www/myapp/www --
url="http://www.myapp.dev" --title="Invent Value" --admin_user="admin"
--admin_password="myapp" --admin_email="admin@myapp.dev" --allow-root

- name: enable and activate myapp theme
  shell: sudo wp theme activate myapp --path=/var/www/myapp/www --
allow-root
  notify: restart apache

```

And finally, the call in Vagrant:

```
hostname = 'myapp.dev'
```

```
config.vm.provision "ansible" do |ansible|  
  # Provision WordPress submodule as www.{{hostname}}  
  ansible.playbook = "/path/to/playbook.yml"  
  ansible.extra_vars = {  
    hostname: hostname  
  }  
end
```