Web Runbook

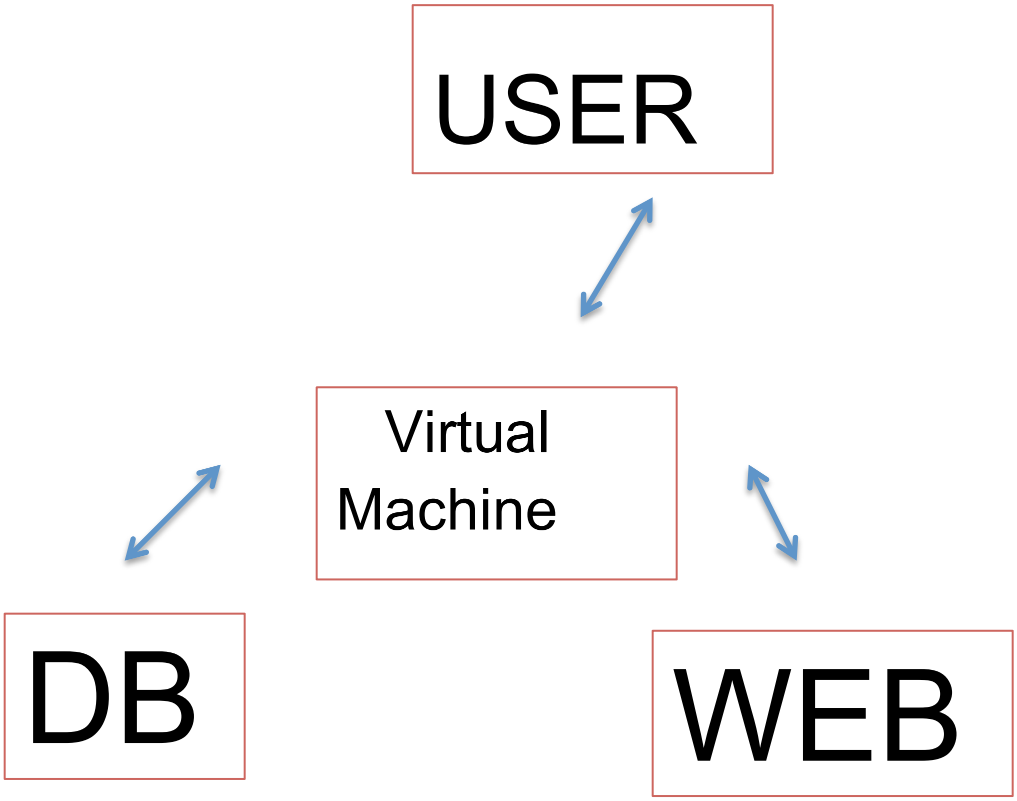
Short Description

Installs php, and nginx services while booting them properly

Required Software

SELinux, epel-release, nginx, php, php-fpm, php-ldap, php-mbstring,php-mcrypt, php-mysql, php-phpunit-PHPUnit, composer

Architecture Diagram



Deployment

```
ansible-playbook -i hosts.ini web.yml --ask-vault-pass
Vault password: secret

PLAY [web]
***********************************************************
**********

TASK [setup]
***********************************************************
********
ok: [localhost]

TASK [Disabling SELinux]
*******************************************************
become: yes
      lineinfile: dest=/etc/sysconfig/selinux
regexp="^SELINUX=" line="SELINUX=permissive"
ok: [localhost]

TASK [Setting SELinux in permissive mode]
**************************************
become: yes
      command: setenforce 0
changed: [localhost]

TASK [Installing epel-release and updates them using yum]
**********************
yum: name=epel-release update_cache=yes state=present
ok: [localhost]

TASK [Installing nginx, php modules and updates them using
yum] ****************

become: yes
      yum: name=nginx,php,php-fpm,php-ldap,php-
```

```
mbstring,php-mcrypt,php-mysql,php-phpunit-PHPUnit
update_cache=yes state=present
      register: nginx_install
ok: [localhost]

TASK [Downloading composer]
****************************************************
become: yes
      get_url:
        url: https://getcomposer.org/installer
        dest: /tmp/installer
ok: [localhost]

TASK [Installing composer]
*****************************************************
shell: cat /tmp/installer | php -- --install-
dir=/usr/local/bin
changed: [localhost]

TASK [Copying nginx's configuration file to destination
location] **************
become: yes
      copy: src=web/nginx.conf dest=/etc/nginx/nginx.conf
mode=0000
      register: nginx_conf
ok: [localhost]

TASK [Templates vhost configuration files]
*************************************
become: yes
      template: src=web/vhost.conf.j2
dest=/etc/nginx/conf.d/{{ service_name }}.conf mode=0644
      register: vhost_conf
ok: [localhost]

TASK [Templates php.ini file]
**************************************************
```

```
template: src=web/php.ini.j2 dest=/etc/php.ini mode=0644
      register: php_conf
ok: [localhost]

TASK [Copying php configuration file into destined file]
**********************
become: yes
      copy: src=web/php-fpm-web.conf dest=/etc/php-
fpm.d/www.conf mode=0644
      register: php_fpm_conf
ok: [localhost]

TASK [Starts nginx on boot]
***************************************************
become: yes
      service: name=nginx state=restarted enabled=yes
      when: nginx_install|changed or nginx_conf|changed or
vhost_conf|changed
skipping: [localhost]

TASK [Restarts FastCGI Process Manager when nginx is
changed] *****************
service: name=php-fpm state=restarted enabled=yes
      when: nginx_install|changed or php_conf|changed or
php_fpm_conf|changed
skipping: [localhost]

TASK [Sets service_dir as a directory]
****************************************
become: yes
      file: path="{{ service_dir }}" state=directory
ok: [localhost]

TASK [Unpacking service_name.tgz to service_dir directory]
********************
   become: yes
      unarchive: src=web/{{ service_name }}.tgz dest="{{
```

```
service_dir }}" mode=0755
ok: [localhost]

TASK [Creates service_dir with composer]
***************************************
become: yes
      command: /usr/local/bin/composer.phar update chdir={{
service_dir }}
changed: [localhost]

TASK [Creates service_directory]
***********************************************
     become: yes
      shell: chdir="{{ service_dir }}" mode=0777
changed: [localhost]

TASK [Creates ngnix as a group]
************************************************
group: name=nginx state=present
ok: [localhost]

TASK [Creates nginx as a user]
*************************************************
user: name=ngnix state=present
ok: [localhost]

TASK [Creates nginx directory]
*************************************************
shell: chdir="{{ service_dir }}" state=directory
group=nginx owner=nginx
changed: [localhost]

TASK [Templates env.j2 to service_dir]
*****************************************


become: yes
      template: src=web/env.j2 dest="{{ service_dir
```

```
}}/.env" mode=0644 owner=nginx

ok: [localhost]
```

Issues

Title: Disk Space

Description: The user doesn’t have enough allocated space to download and install certain software.

Remediation Steps: Make more space by deleting unnecessary files that doesn’t hinder the OS.

Title: Process Speed

Description: The user doesn’t have enough processing speed to correctly install the software in the playbook.

Remediation Steps: Increase the VM processing speed in settings and add more base memory ram to increase speed.

Title: Unprotected Network

Description: Unprotected network connections are very unreliable and risky. Users might be connected to an unprotected network and have their computer compromised.

Remediation Steps: Make sure to have a VPN connection through the VM to hide your IP address. Even though you’re already using a VM it’s not unnecessary to be anonymous in an unprotected network.