

Lab Setup



Lab Setup

To set up the virtual machine (VM) needed for the PHP course labs, follow these general steps:

1. Install Oracle VirtualBox
2. Install Hashicorp Vagrant
3. Use the *Vagrantfile* to build the course VirtualBox
4. Login to the new VM and update the existing software

The next several slides provide you with more detail on the lab VM setup.



Lab Setup

Install VirtualBox and Vagrant

Install Oracle VirtualBox using the instructions provided here: <https://www.virtualbox.org/manual/ch02.html>

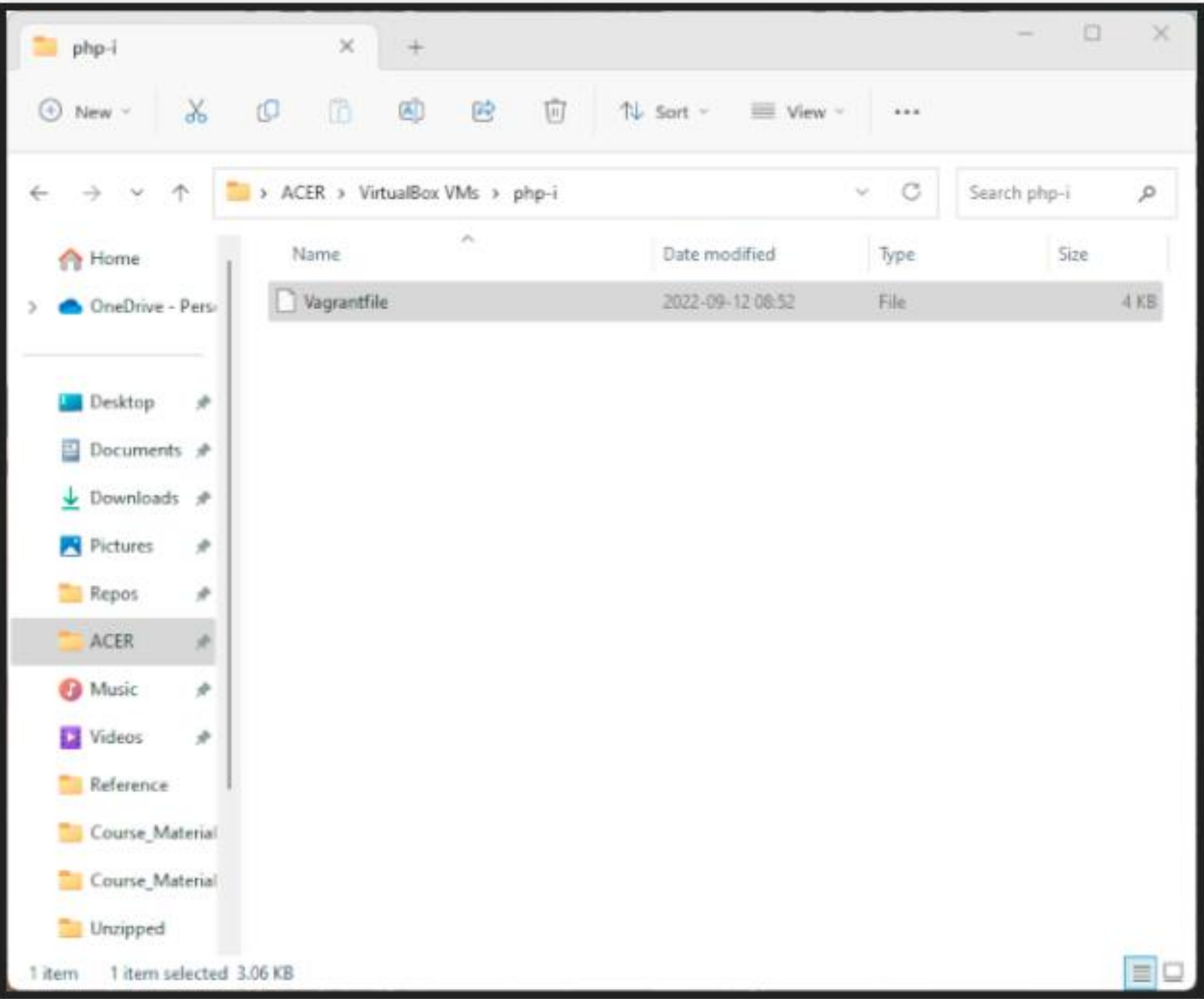


Install Hashicorp Vagrant using the instructions provided here: <https://developer.hashicorp.com/vagrant/downloads>



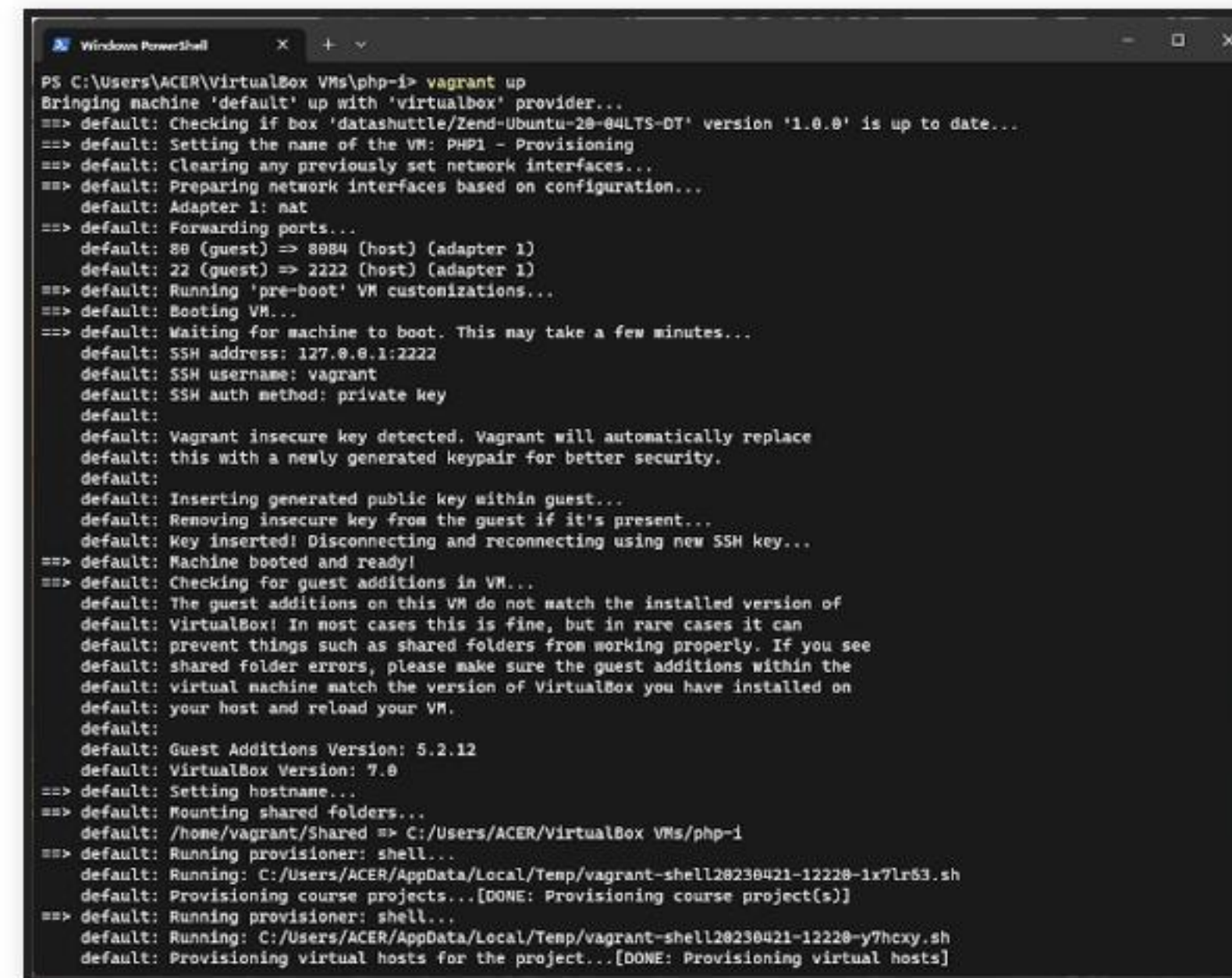
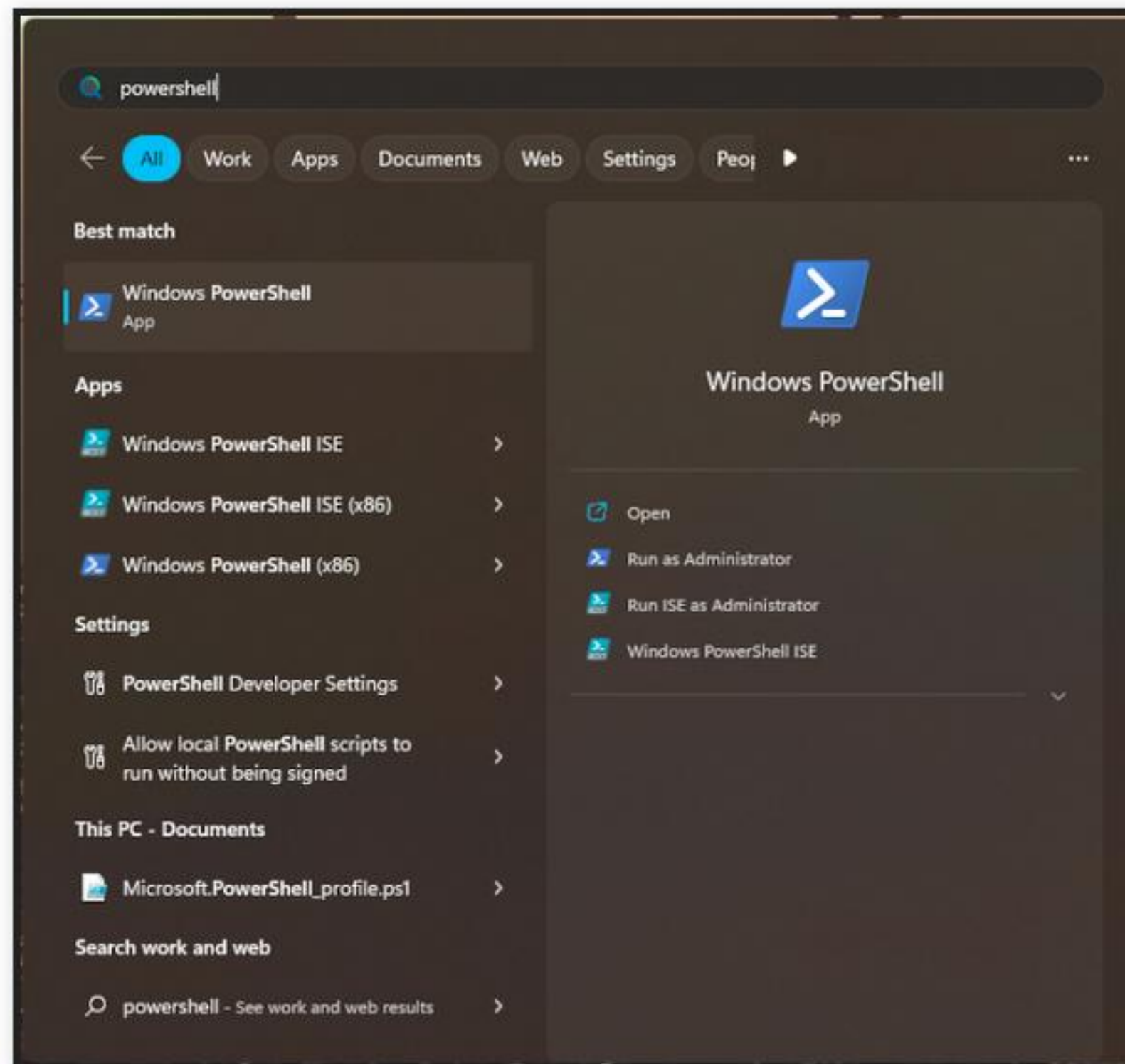
Download the Vagrantfile

Create an empty directory on the computer you plan to use for the labs. Using the link provided to you when you signed up for the course, download the *Vagrantfile*, and move it to the new directory.



Use Vagrant to Build the VM

Open a command terminal (or Powershell), and run the command *vagrant up*. The "provisioning" process can take up to one to two hours to complete depending on the speed of your computer and network connection.



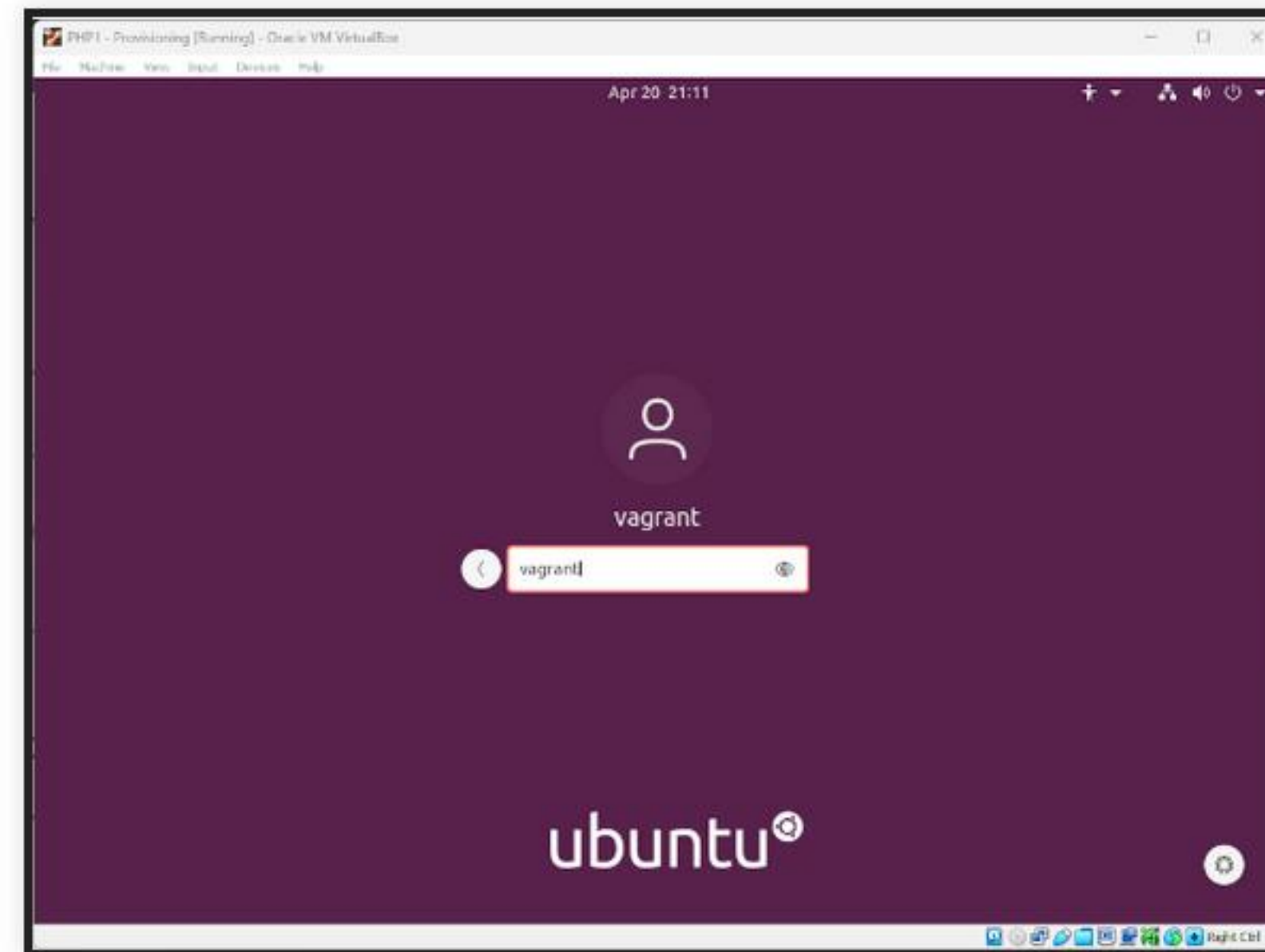
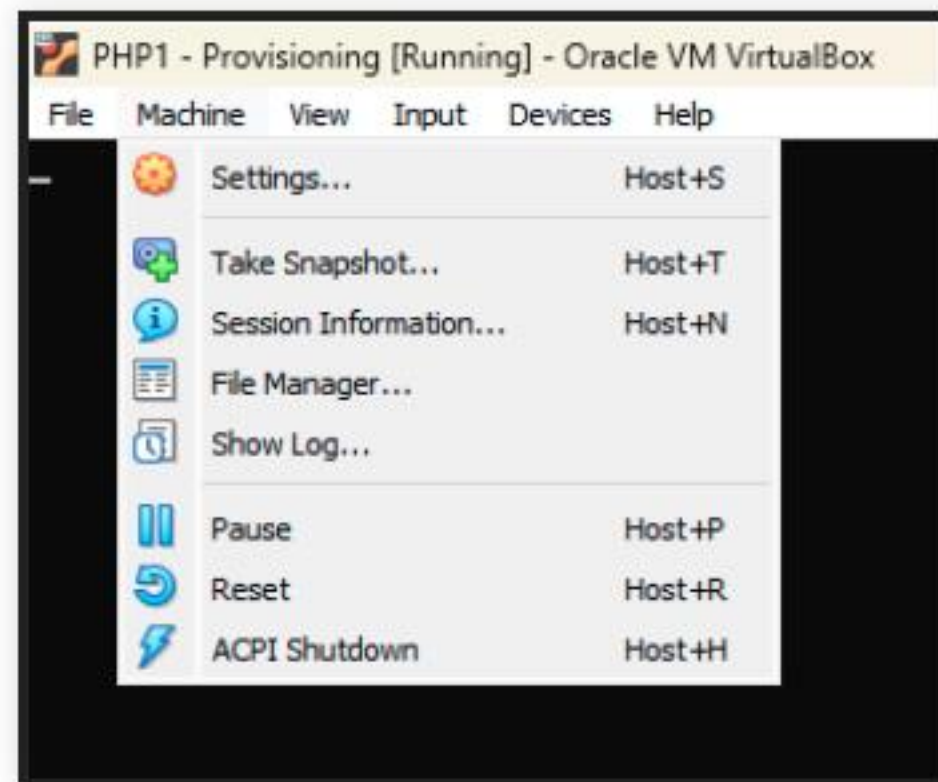
Lab Setup

Reset VM and Login

When provisioning completes successfully, Ubuntu will boot up in a separate window.

In many cases you'll need to reset the VM using the menu at the top left.

When the VM is ready, login as the user *vagrant* using the password *vagrant*.



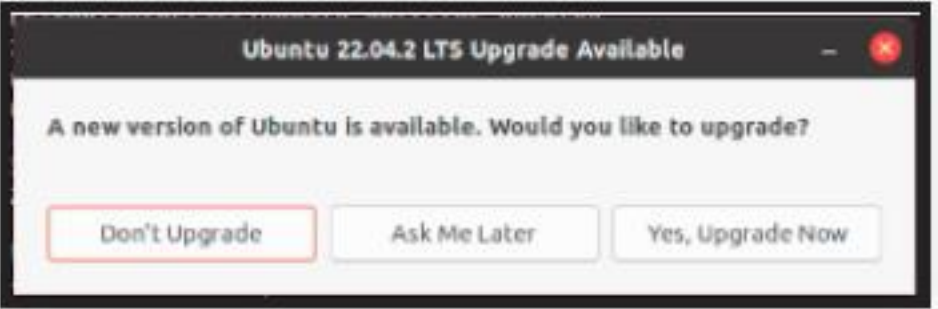
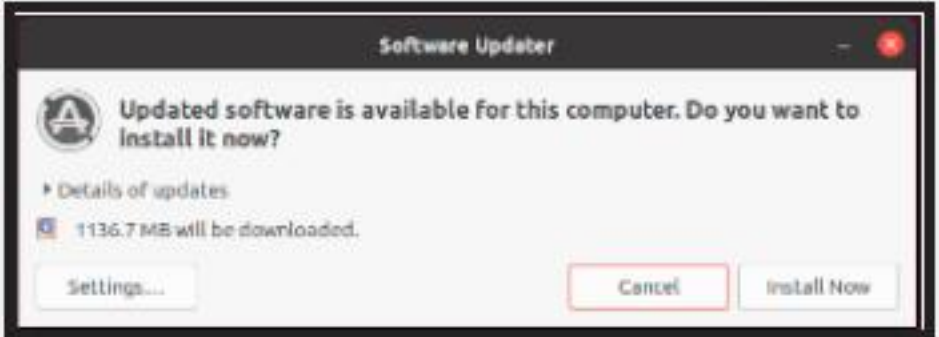
Lab Setup

Do Not Accept the Update or Upgrade Prompts

Once you login it's important to wait a few seconds for the system to come fully up.

At this point you'll see two prompts: one to update, one to upgrade.

Be sure to decline both of these options!



Open a command terminal and run these commands. It will take several hours to complete so it's best to let it run overnight.

```
$ sudo dpkg --configure -a
$ sudo apt -y update && sudo apt -f -y install && sudo apt -y full-upgrade
```



Lab Setup

Accept New Configuration

At some point you will be asked if you wish to retain the original *php.ini* configuration or accept the new. Go ahead and **accept the new configuration**.

```
Modified configuration file
php.ini: A new version (/usr/lib/php/8.0/php.ini-production) of configuration file /etc/php/8.0/apache2/php.ini is available, but
the version installed currently has been locally modified.

What do you want to do about modified configuration file php.ini?

Install the package maintainer's version
keep the local version currently installed
show the differences between the versions
show a side-by-side difference between the versions
start a new shell to examine the situation

<ok>
```

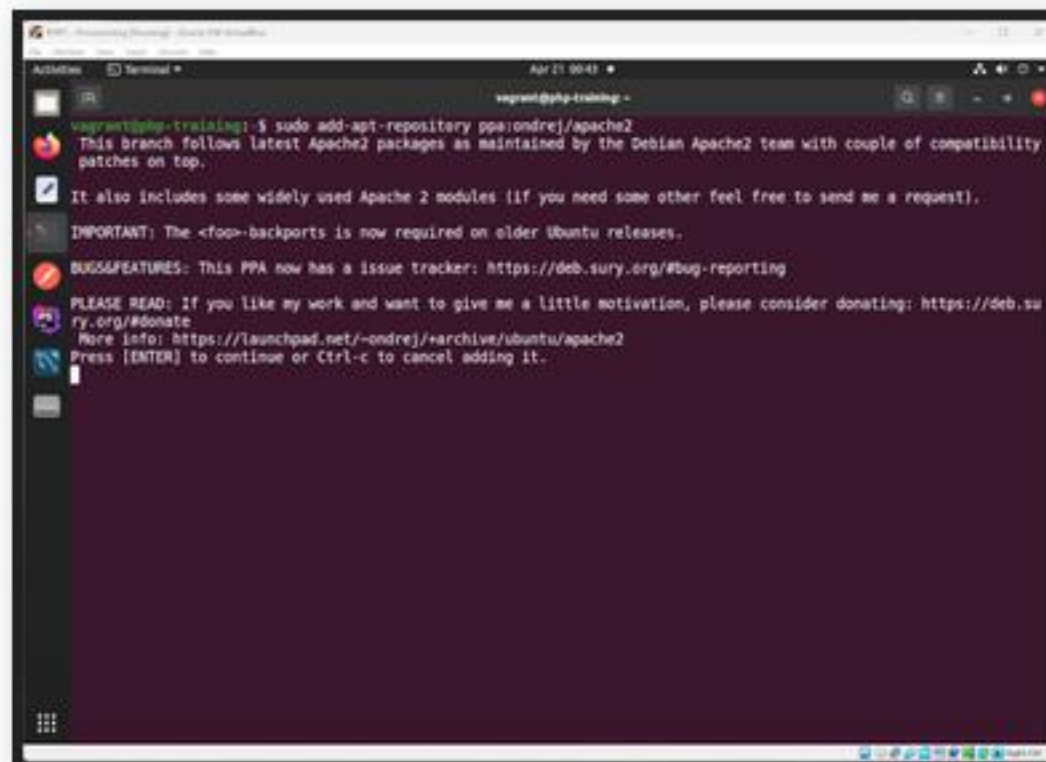


Lab Setup

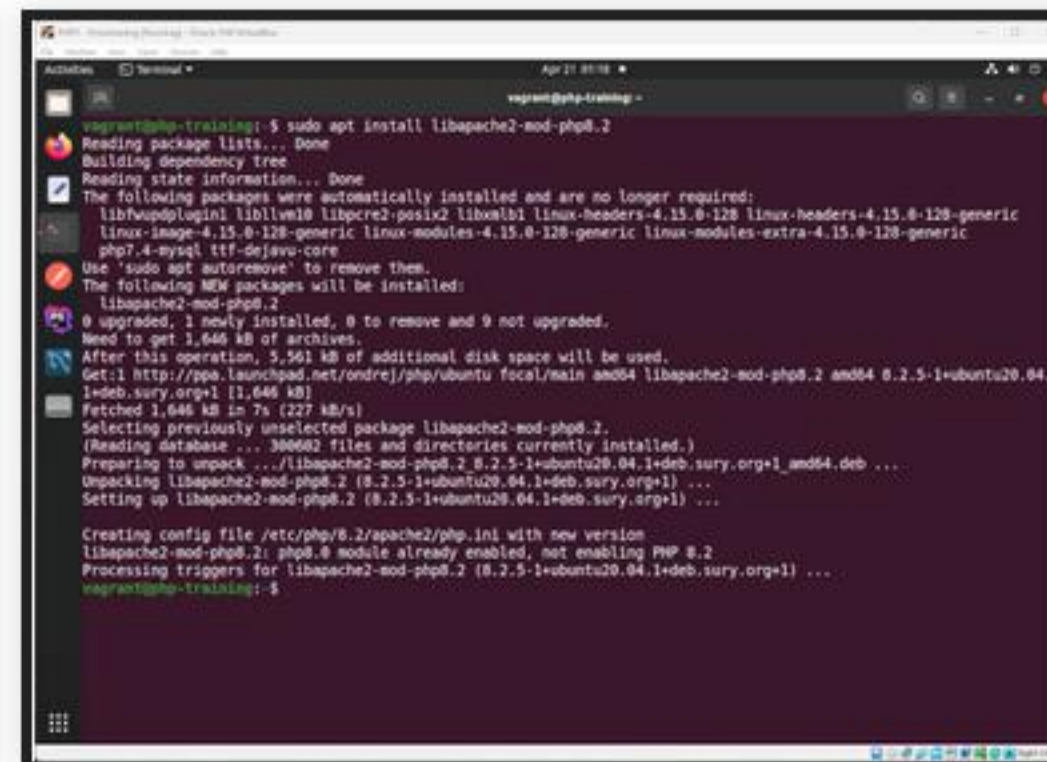
Update Apache PHP Module

So far PHP from the command line (PHP-CLI) has been updated. You'll still need to update the PHP Apache module using these commands. Please note that "8.0" is the old version, and "8.2" is the new version. You may have to change these two values as more recent versions become available.

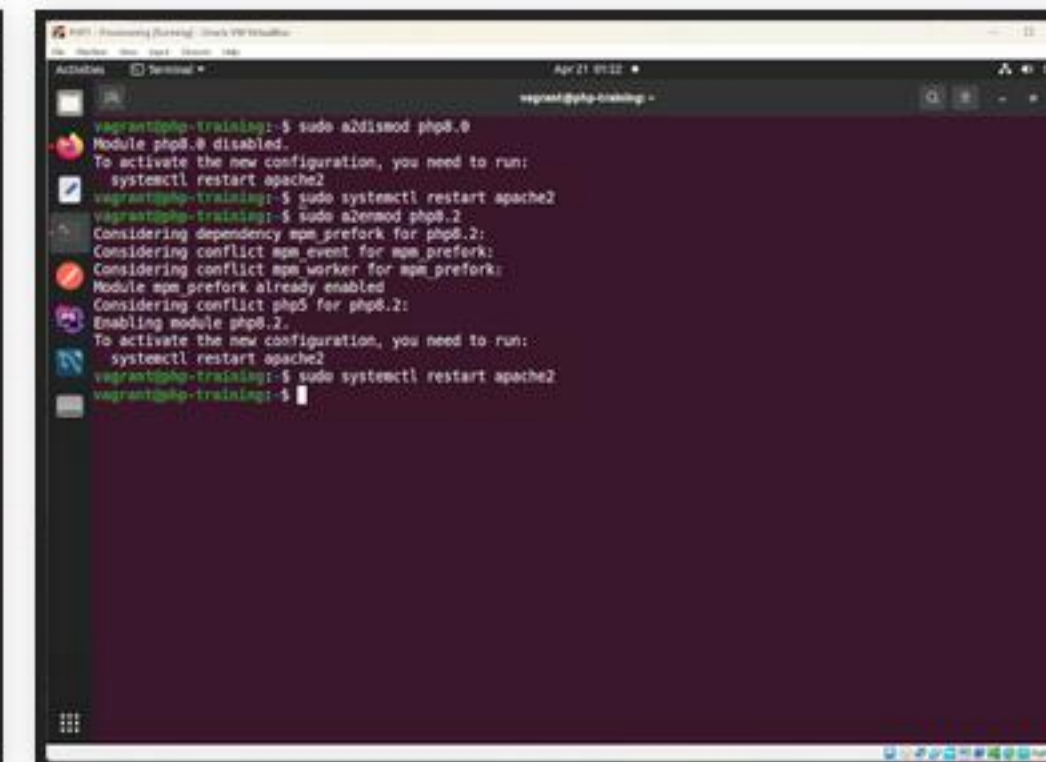
```
sudo apt-add-repository ppa:ondrej/apache2
sudo apt install libapache2-mod-php8.2
sudo a2dismod php8.0
sudo systemctl restart apache2
sudo a2enmod php8.2
sudo systemctl restart apache2
```



```
vagrant@php-training:~$ sudo apt-add-repository ppa:ondrej/apache2
Adding repository.
This branch follows latest Apache2 packages as maintained by the Debian Apache2 team with couple of compatibility patches on top.
It also includes some widely used Apache 2 modules (if you need some other feel free to send me a request).
IMPORTANT: The <foo>-backports is now required on older Ubuntu releases.
BUGS/FEATURES: This PPA now has a issue tracker: https://deb.sury.org/#bug-reporting
PLEASE READ: If you like my work and want to give me a little motivation, please consider donating: https://deb.sury.org/#donate
More info: https://launchpad.net/~ondrej/+archive/ubuntu/apache2
Press [ENTER] to continue or Ctrl-C to cancel adding it.
vagrant@php-training:~$
```



```
vagrant@php-training:~$ sudo apt install libapache2-mod-php8.2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libfswapdplugin liblvm2 libpcre2-posix2 libxmlb1 linux-headers-4.15.0-128 linux-headers-4.15.0-128-generic
  linux-image-4.15.0-128-generic linux-modules-4.15.0-128-generic linux-modules-extra-4.15.0-128-generic
  php7.4-mysql ttf-dejavu-core
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  libapache2-mod-php8.2
0 upgraded, 1 newly installed, 0 to remove and 9 not upgraded.
Need to get 1,646 kB of archives.
After this operation, 5,561 kB of additional disk space will be used.
Get:1 http://ppa.launchpad.net/ondrej/php/ubuntu focal/main amd64 libapache2-mod-php8.2 amd64 8.2.5-1ubuntu20.04.1+deb.sury.org~1 [1,646 kB]
Fetched 1,646 kB in 7s (227 kB/s)
Selecting previously unselected package libapache2-mod-php8.2.
(Reading database ... 380662 files and directories currently installed.)
Preparing to unpack .../libapache2-mod-php8.2_8.2.5-1ubuntu20.04.1+deb.sury.org~1_amd64.deb ...
Unpacking libapache2-mod-php8.2 (8.2.5-1ubuntu20.04.1+deb.sury.org~1) ...
Setting up libapache2-mod-php8.2 (8.2.5-1ubuntu20.04.1+deb.sury.org~1) ...
Creating config file /etc/php/8.2/apache2/php.ini with new version
libapache2-mod-php8.2: php8.0 module already enabled, not enabling PHP 8.2
Processing triggers for libapache2-mod-php8.2 (8.2.5-1ubuntu20.04.1+deb.sury.org~1) ...
vagrant@php-training:~$
```



```
vagrant@php-training:~$ sudo a2dismod php8.0
Module php8.0 disabled.
To activate the new configuration, you need to run:
  systemctl restart apache2
vagrant@php-training:~$ sudo systemctl restart apache2
vagrant@php-training:~$ sudo a2enmod php8.2
Considering dependency mpm_prefork for php8.2:
Considering conflict mpm_event for mpm_prefork:
Considering conflict mpm_worker for mpm_prefork:
Module mpm_prefork already enabled
Considering conflict php5 for php8.2:
Enabling module php8.2.
To activate the new configuration, you need to run:
  systemctl restart apache2
vagrant@php-training:~$ sudo systemctl restart apache2
vagrant@php-training:~$
```



Lab Setup

Confirm PHP Installation

Confirm the PHP installation by opening the VM's browser, and entering *http://sandbox/*

