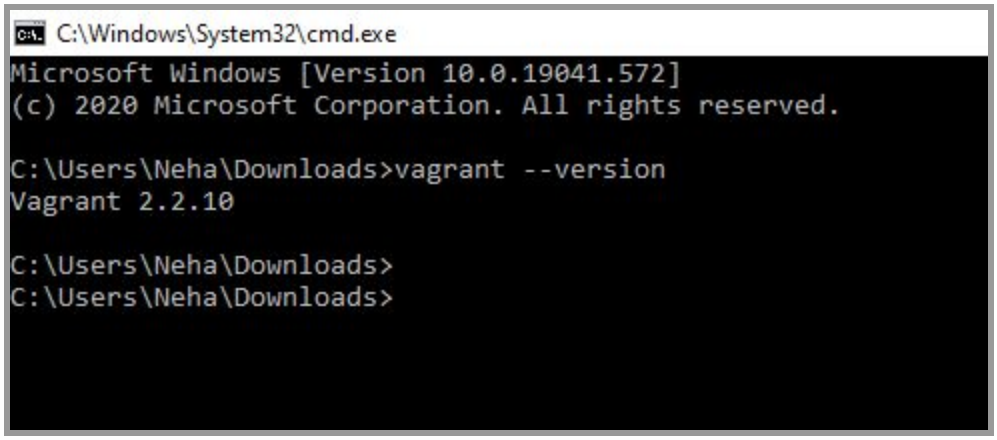
Vagrant & Docker
Experiment 11
~Neha Singh

Vagrant: **Vagrant is an open-source software product for building and maintaining portable virtual software development environments; e.g., for VirtualBox, KVM, Hyper-V, Docker containers, VMware, and AWS.**

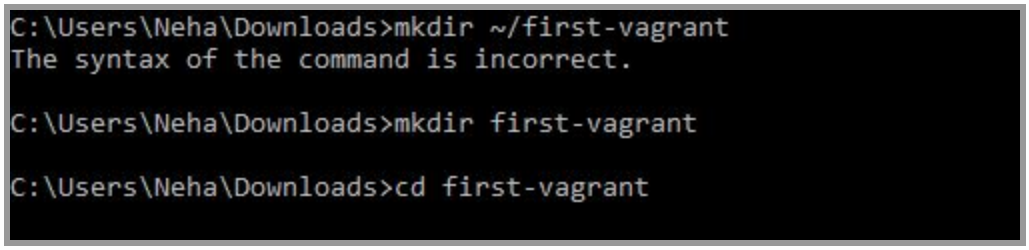
Docker: **Docker is a set of the platform as service products that use OS-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries, and configuration files; they can communicate with each other through well-defined channels.**

Download Vagrant from here.
https://www.vagrantup.com/downloads

Check if Vagrant is downloaded or not by running
Vagrant --version



Make a directory and move to that directory

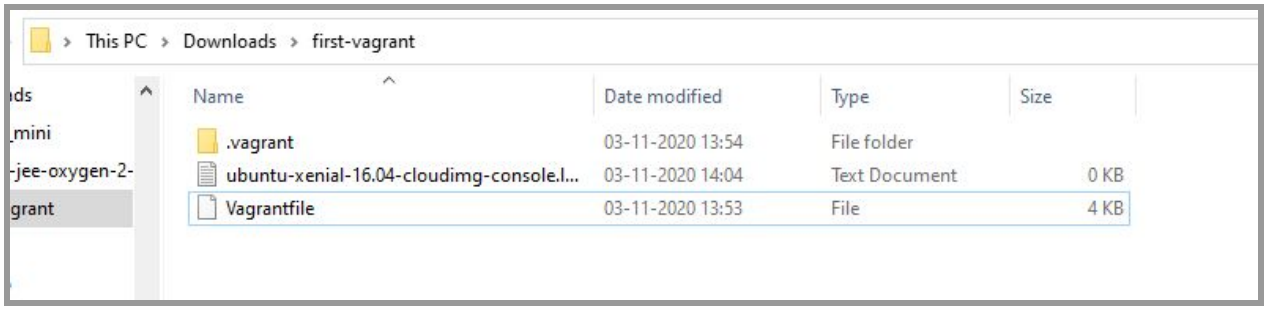


Now, edit vagrantfile and mention machine name ubuntu/xenial64
Config.vm.box = “ubuntu/xenial64”

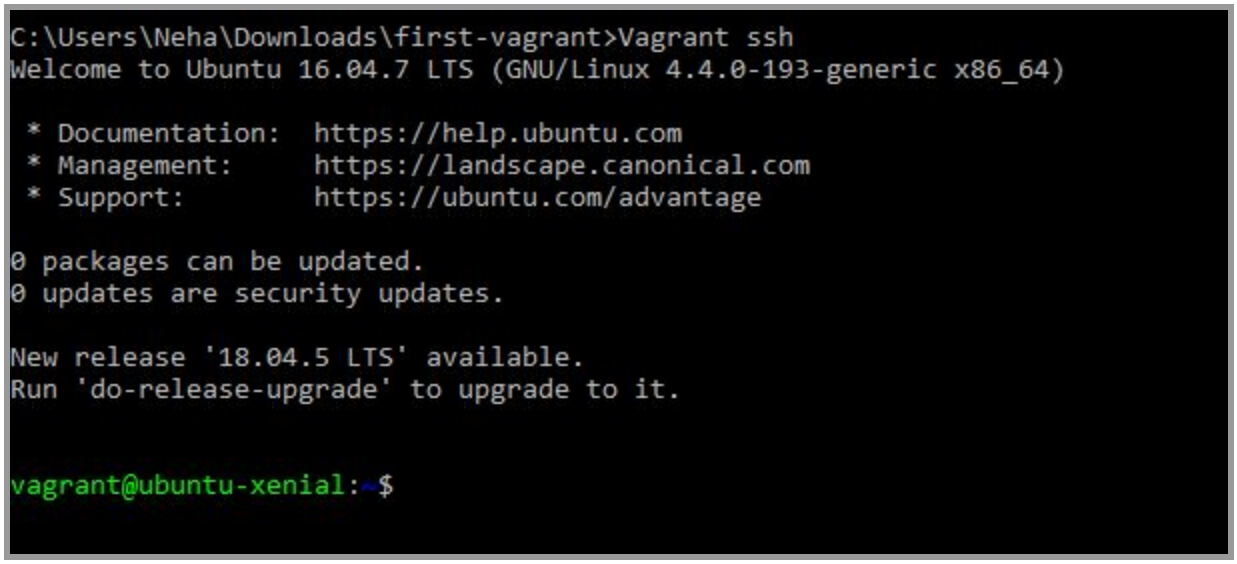
Now, run the vagrant up command

```
C:\Users\Neha\Downloads\first-vagrant>vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'ubuntu/xenial64' could not be found. Attempting to find and install...
    default: Box Provider: virtualbox
    default: Box Version: >= 0
==> default: Loading metadata for box 'ubuntu/xenial64'
    default: URL: https://vagrantcloud.com/ubuntu/xenial64
==> default: Adding box 'ubuntu/xenial64' (v20201102.0.0) for provider: virtualbox
    default: Downloading: https://vagrantcloud.com/ubuntu/boxes/xenial64/versions/20201102.0.0/providers/virtualbox.box
Download redirected to host: cloud-images.ubuntu.com
    default:
==> default: Successfully added box 'ubuntu/xenial64' (v20201102.0.0) for 'virtualbox'!
==> default: Importing base box 'ubuntu/xenial64'...
==> default: Matching MAC address for NAT networking...
==> default: Checking if box 'ubuntu/xenial64' version '20201102.0.0' is up to date...
==> default: Setting the name of the VM: first-vagrant_default_1604392409606_26789
==> default: Clearing any previously set network interfaces...
==> default: Preparing network interfaces based on configuration...
    default: Adapter 1: nat
==> default: Forwarding ports...
    default: 22 (guest) => 2222 (host) (adapter 1)
==> default: Running 'pre-boot' VM customizations...
==> default: Booting VM...
==> default: Waiting for machine to boot. This may take a few minutes...
    default: SSH address: 127.0.0.1:2222
    default: SSH username: vagrant
    default: SSH auth method: private key
    default: Warning: Connection aborted. Retrying...
    default: Warning: Remote connection disconnect. Retrying...
    default: Warning: Connection reset. Retrying...
    default: Warning: Connection aborted. Retrying...
    default: Warning: Connection reset. Retrying...
    default: Warning: Connection aborted. Retrying...
    default:
    default: Vagrant insecure key detected. Vagrant will automatically replace
    default: this with a newly generated keypair for better security.
    default:
    default: Inserting generated public key within guest...
    default: Removing insecure key from the guest if it's present...
    default: Key inserted! Disconnecting and reconnecting using new SSH key...
```

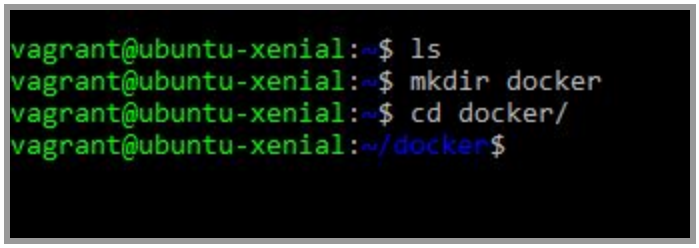
Folder after vagrant up command



Virtual Machine started



Virtual Machine running



First, in order to ensure the downloads are valid, add the GPG key for the official Docker repository to your system

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo
apt-key add -
```

Add the Docker repository to APT sources

```
$ sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
```

Next, update the package database with the Docker packages from the newly added repo:

```
$ sudo apt-get update
```

```
vagrant@ubuntu-xenial:~/docker$ sudo apt-get update
Hit:1 http://archive.ubuntu.com/ubuntu xenial InRelease
Get:2 http://archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]
Get:3 https://download.docker.com/linux/ubuntu xenial InRelease [66.2 kB]
Get:4 http://security.ubuntu.com/ubuntu xenial-security InRelease [109 kB]
Get:5 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages [15.0 kB]
Get:6 http://archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]
Get:7 http://archive.ubuntu.com/ubuntu xenial/universe amd64 Packages [7,532 kB]
Get:8 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 Packages [772 kB]
Get:9 http://security.ubuntu.com/ubuntu xenial-security/universe Translation-en [218 kB]
Get:10 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 Packages [8,236 B]
Get:11 http://security.ubuntu.com/ubuntu xenial-security/multiverse Translation-en [2,888 B]
Get:12 http://archive.ubuntu.com/ubuntu xenial/universe Translation-en [4,354 kB]
Get:13 http://archive.ubuntu.com/ubuntu xenial/multiverse amd64 Packages [144 kB]
Get:14 http://archive.ubuntu.com/ubuntu xenial/multiverse Translation-en [106 kB]
Get:15 http://archive.ubuntu.com/ubuntu xenial-updates/universe amd64 Packages [1,194 kB]
Get:16 http://archive.ubuntu.com/ubuntu xenial-updates/universe Translation-en [348 kB]
Get:17 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 Packages [23.0 kB]
Get:18 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse Translation-en [8,632 B]
Get:19 http://archive.ubuntu.com/ubuntu xenial-backports/main amd64 Packages [9,812 B]
Get:20 http://archive.ubuntu.com/ubuntu xenial-backports/main Translation-en [4,456 B]
Get:21 http://archive.ubuntu.com/ubuntu xenial-backports/universe amd64 Packages [11.3 kB]
Get:22 http://archive.ubuntu.com/ubuntu xenial-backports/universe Translation-en [4,476 B]
Fetched 15.1 MB in 40s (374 kB/s)
Reading package lists... Done
```

Make sure you are about to install from the Docker repo instead of the default Ubuntu 16.04 repo:

$ apt-cache policy docker-ce

```
vagrant@ubuntu-xenial:~/docker$ apt-cache policy docker-ce
docker-ce:
  Installed: (none)
  Candidate: 5:19.03.13~3-0~ubuntu-xenial
  Version table:
     5:19.03.13~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.12~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.11~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.10~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.9~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.8~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.7~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.6~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.5~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.4~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.3~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.2~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.1~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:19.03.0~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:18.09.9~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:18.09.8~3-0~ubuntu-xenial 500
        500 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages
     5:18.09.7~3-0~ubuntu-xenial 500
```

Finally, install Docker:

`$ sudo apt-get install -y docker-ce`

```
vagrant@ubuntu-xenial:~/docker$  sudo apt-get install -y docker-ce
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  aufs-tools cgroupfs-mount containerd.io docker-ce-cli libltdl7 pigz
Suggested packages:
  mountall
The following NEW packages will be installed:
  aufs-tools cgroupfs-mount containerd.io docker-ce docker-ce-cli libltdl7 pigz
0 upgraded, 7 newly installed, 0 to remove and 0 not upgraded.
Need to get 91.2 MB of archives.
After this operation, 410 MB of additional disk space will be used.
Get:1 https://download.docker.com/linux/ubuntu xenial/stable amd64 containerd.io amd64 1.3.7-1 [24.3 MB]
Get:2 http://archive.ubuntu.com/ubuntu xenial/universe amd64 pigz amd64 2.3.1-2 [61.1 kB]
Get:3 http://archive.ubuntu.com/ubuntu xenial/universe amd64 aufs-tools amd64 1:3.2+20130722-1.1ubuntu1 [92.9 kB]
Get:4 http://archive.ubuntu.com/ubuntu xenial/universe amd64 cgroupfs-mount all 1.2 [4,970 B]
Get:5 http://archive.ubuntu.com/ubuntu xenial/main amd64 libltdl7 amd64 2.4.6-0.1 [38.3 kB]
Get:6 https://download.docker.com/linux/ubuntu xenial/stable amd64 docker-ce-cli amd64 5:19.03.13~3-0~ubuntu-xenial [44.2 MB]
Get:7 https://download.docker.com/linux/ubuntu xenial/stable amd64 docker-ce amd64 5:19.03.13~3-0~ubuntu-xenial [22.5 MB]
Fetched 91.2 MB in 29s (3,082 kB/s)
Selecting previously unselected package pigz.
(Reading database ... 54328 files and directories currently installed.)
Preparing to unpack .../pigz_2.3.1-2_amd64.deb ...
Unpacking pigz (2.3.1-2) ...
Selecting previously unselected package aufs-tools.
Preparing to unpack .../aufs-tools_1%3a3.2+20130722-1.1ubuntu1_amd64.deb ...
Unpacking aufs-tools (1:3.2+20130722-1.1ubuntu1) ...
Selecting previously unselected package cgroupfs-mount.
Preparing to unpack .../cgroupfs-mount_1.2_all.deb ...
Unpacking cgroupfs-mount (1.2) ...
Selecting previously unselected package containerd.io.
Preparing to unpack .../containerd.io_1.3.7-1_amd64.deb ...
Unpacking containerd.io (1.3.7-1) ...
Selecting previously unselected package docker-ce-cli.
Preparing to unpack .../docker-ce-cli_5%3a19.03.13~3-0~ubuntu-xenial_amd64.deb ...
Unpacking docker-ce-cli (5:19.03.13~3-0~ubuntu-xenial) ...
Selecting previously unselected package docker-ce.
Preparing to unpack .../docker-ce_5%3a19.03.13~3-0~ubuntu-xenial_amd64.deb ...
Unpacking docker-ce (5:19.03.13~3-0~ubuntu-xenial) ...
Selecting previously unselected package libltdl7:amd64.
```

Check that it's running:

`$ sudo systemctl status docker`

```
vagrant@ubuntu-xenial:~/docker$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2020-11-03 08:56:57 UTC; 1min 45s ago
     Docs: https://docs.docker.com
 Main PID: 3858 (dockerd)
   CGroup: /system.slice/docker.service
           └─3858 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Nov 03 08:56:56 ubuntu-xenial dockerd[3858]: time="2020-11-03T08:56:56.058149873Z" level=warning msg="Your kernel does not support swap memory limit"
Nov 03 08:56:56 ubuntu-xenial dockerd[3858]: time="2020-11-03T08:56:56.059179722Z" level=warning msg="Your kernel does not support cgroup rt period"
Nov 03 08:56:56 ubuntu-xenial dockerd[3858]: time="2020-11-03T08:56:56.060391525Z" level=warning msg="Your kernel does not support cgroup rt runtime"
Nov 03 08:56:56 ubuntu-xenial dockerd[3858]: time="2020-11-03T08:56:56.061091372Z" level=info msg="Loading containers: start."
Nov 03 08:56:56 ubuntu-xenial dockerd[3858]: time="2020-11-03T08:56:56.819039649Z" level=info msg="Default bridge (docker0) is assigned with an IP address 172.17.0.0/16
Nov 03 08:56:57 ubuntu-xenial dockerd[3858]: time="2020-11-03T08:56:57.017023039Z" level=info msg="Loading containers: done."
Nov 03 08:56:57 ubuntu-xenial dockerd[3858]: time="2020-11-03T08:56:57.663889935Z" level=info msg="Docker daemon" commit=4484c46d9d graphdriver(s)=overlay2 version=19.0
Nov 03 08:56:57 ubuntu-xenial dockerd[3858]: time="2020-11-03T08:56:57.665325377Z" level=info msg="Daemon has completed initialization"
Nov 03 08:56:57 ubuntu-xenial systemd[1]: Started Docker Application Container Engine.
Nov 03 08:56:57 ubuntu-xenial dockerd[3858]: time="2020-11-03T08:56:57.725007958Z" level=info msg="API listen on /var/run/docker.sock"

vagrant@ubuntu-xenial:~/docker$
```

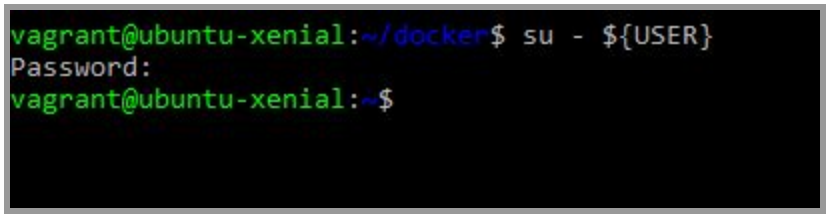
- **Configuration**

Executing the Docker Command Without Sudo
If you want to avoid typing sudo whenever you run the docker command, add your username to the docker group:

```
$ sudo usermod -aG docker ${USER}
```

To apply the new group membership, you can log out of the server and back in, or you can type the following:

```
$ su - ${USER}
```



You will be prompted to enter your user's password to continue. Afterwards, you can confirm that your user is now added to the docker group by typing:

If you need to add a user to the docker group that you're not logged in as, declare that username explicitly using:

```
$ sudo usermod -aG docker username
```