

EXPERIMENT 2

Starting VM

\$ vagrant up

```
F:\first-vagrant>vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Checking if box 'ubuntu/xenial64' version '20201104.0.0' is up to date...
==> default: Clearing any previously set forwarded ports...
==> default: Fixed port collision for 22 => 2222. Now on port 2200.
==> default: Clearing any previously set network interfaces...
==> default: Preparing network interfaces based on configuration...
    default: Adapter 1: nat
==> default: Forwarding ports...
    default: 22 (guest) => 2200 (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: 127.0.0.1:2200
    default: SSH username: vagrant
    default: SSH auth method: private key
    default: Warning: Connection aborted. Retrying...
    default: Warning: Remote connection disconnect. Retrying...
==> default: Machine booted and ready!
==> default: Checking for guest additions in VM...
    default: The guest additions on this VM do not match the installed version of
    default: VirtualBox! In most cases this is fine, but in rare cases it can
```

Listing all the volumes in docker and inspecting a volume

\$ docker volume ls

```
vagrant@ubuntu-xenial:~$ ls
vagrant@ubuntu-xenial:~$ docker --version
Docker version 19.03.13, build 4484c46d9d
vagrant@ubuntu-xenial:~$ docker volume ls
DRIVER      VOLUME NAME
local       84b27a886a4606b3ae662f1128af362e3ba98e6f7c69ecd95d2616511c7e3275
local       7159a0eaca4ace48a606f9b4caa09fb32a4ddb4b5a2621d1696b24275387414c
local       614531430a43274c4e256347bdadf770d2a858ac7faca8094a84613dc7f5c7fe
local       d8774dcbbc276d80b3ac096f9220b6c79c03336f9c040ac77421875dd8d42f04
```

```
vagrant@ubuntu-xenial:~$ docker volume inspect 84b27a886a4606b3ae662f1128af362e3ba98e6f7c69ecd95d2616511c7e3275
[
  {
    "CreatedAt": "2020-11-20T06:38:18Z",
    "Driver": "local",
    "Labels": null,
    "Mountpoint": "/var/lib/docker/volumes/84b27a886a4606b3ae662f1128af362e3ba98e6f7c69ecd95d2616511c7e3275/_data",
    "Name": "84b27a886a4606b3ae662f1128af362e3ba98e6f7c69ecd95d2616511c7e3275",
    "Options": null,
    "Scope": "local"
  }
]
vagrant@ubuntu-xenial:~$ docker run alpine
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
4c0d98bf9879: Pull complete
Digest: sha256:08d6ca16c60fe7490c03d10dc339d9fd8ea67c6466dea8d558526b1330a85930
```

Running an Alpine Image

\$ docker run alpine

```
vagrant@ubuntu-xenial:~$ docker run alpine
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
4c0d98bf9879: Pull complete
Digest: sha256:08d6ca16c69fe7490c03d10dc339d9fd8ea67c6466dea8d558526b1330a85930
Status: Downloaded newer image for alpine:latest
vagrant@ubuntu-xenial:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
vagrant@ubuntu-xenial:~$ docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
a7ca31cdaebc       alpine             "/bin/sh"          17 seconds ago     Exited (0) 16 seconds ago              pedantic_tereshkova
fc5214159b1b       redis              "docker-entrypoint.s..." 2 months ago       Exited (0) 2 months ago              test
dd09f62b7568       redis              "docker-entrypoint.s..." 2 months ago       Exited (0) 2 months ago              sharp_keller
ace4825d6132       redis              "docker-entrypoint.s..." 2 months ago       Exited (255) 20 minutes ago          gifted_murdock
989295354bd4       redis              "docker-entrypoint.s..." 2 months ago       Exited (0) 2 months ago              infallible_swartz
2e6d810bd42a       ubuntu             "/bin/bash"        2 months ago       Exited (0) 2 months ago              trusting_blackwell
6fd9d26742ee       ubuntu             "/bin/bash"        2 months ago       Exited (0) 2 months ago              trusting_chandrasekhar
```

```
vagrant@ubuntu-xenial:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
vagrant@ubuntu-xenial:~$ docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
91a9d32caea6       alpine             "/bin/sh"          About a minute ago  Exited (0) 14 seconds ago              compassionate_colden
a7ca31cdaebc       alpine             "/bin/sh"          8 minutes ago      Exited (0) 8 minutes ago              pedantic_tereshkova
fc5214159b1b       redis              "docker-entrypoint.s..." 2 months ago       Exited (0) 2 months ago              test
dd09f62b7568       redis              "docker-entrypoint.s..." 2 months ago       Exited (0) 2 months ago              sharp_keller
ace4825d6132       redis              "docker-entrypoint.s..." 2 months ago       Exited (255) 29 minutes ago          gifted_murdock
989295354bd4       redis              "docker-entrypoint.s..." 2 months ago       Exited (0) 2 months ago              infallible_swartz
2e6d810bd42a       ubuntu             "/bin/bash"        2 months ago       Exited (0) 2 months ago              trusting_blackwell
6fd9d26742ee       ubuntu             "/bin/bash"        2 months ago       Exited (0) 2 months ago              trusting_chandrasekhar
vagrant@ubuntu-xenial:~$ docker run -it alpine
/ # cd mnt
/ mnt # ls
/ mnt # exit
vagrant@ubuntu-xenial:~$ docker run -it -v myvol:/mnt alpine
/ # ls
```

Creating some files in that alpine cont.

\$ docker run -it -v myvol:/mnt alpine

```
vagrant@ubuntu-xenial:~$ docker run -it -v myvol:/mnt alpine
/ # ls
bin    dev    etc    home   lib    media  mnt    opt    proc   root   run    sbin   srv    sys    tmp    usr    var
/ # ls
bin    dev    etc    home   lib    media  mnt    opt    proc   root   run    sbin   srv    sys    tmp    usr    var
/ # cd mnt
/ mnt # ls
/ mnt # touch a.txt
/ mnt # touch b.txt
/ mnt # ls
a.txt  b.txt
/ mnt # exit
vagrant@ubuntu-xenial:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
vagrant@ubuntu-xenial:~$ docker run -it -v myvol:/mnt alpine
/ # ls
bin    dev    etc    home   lib    media  mnt    opt    proc   root   run    sbin   srv    sys    tmp    usr    var
/ # cd mnt
/ mnt # ls
a.txt  b.txt
/ mnt # exit
vagrant@ubuntu-xenial:~$ docker run -it alpine
/ # cd mnt
/ mnt # ls
/ mnt # exit
vagrant@ubuntu-xenial:~$ docker volume ls
DRIVER      VOLUME NAME
local      84b27a886a4606b3ae662f1128af362e3ba98e6f7c69ecd95d2616511c7e3275
local      7159a0eaca4ace48a606f9b4caa09fb32a4ddb4b5a2621d1696b24275387414c
local      614531430a43274c4e256347bdad770d2a858ac7faca8094a84613dc7f5c7fe
local      d8774dcbbc276d80b3ac096f9220b6c79c0336f9c040ac77421875dd8d42f04
local      myvol
vagrant@ubuntu-xenial:~$ docker volume inspect myvol
[
  {
    "CreatedAt": "2021-01-29T06:20:39Z",
    "Driver": "local",
    "Labels": null,
    "Mountpoint": "/var/lib/docker/volumes/myvol/_data",
    "Name": "myvol",
    "Options": null,
    "Scope": "local"
  }
]
```

Creating Container with Volume

```
$ docker run -it -v myvol:/mnt --name c1 alpine
```

```
vagrant@ubuntu-xenial:~$ docker run -it -v myvol:/mnt --name c1 alpine
/ # cd /mnt
/mnt # ls
a.txt b.txt
/mnt # ls
a.txt b.txt c.txt
/mnt # ls
a.txt b.txt c.txt d.txt
/mnt # exit
```

In another instance of terminal, starting a VM and running docker in that

```
$ docker run -it -v myvol:/mnt --name c2 alpine
```

```
vagrant@ubuntu-xenial:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS   NAMES
8d261bbd152e   alpine    "/bin/sh"               3 minutes ago Up 3 minutes             c1
vagrant@ubuntu-xenial:~$ docker run -it -v myvol:/mnt --name c2 alpine
/ # ls
bin  dev  etc  home  lib  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
/ # cd /mnt
/mnt # ls
a.txt b.txt
/mnt # touch c.txt
/mnt # ls
a.txt b.txt c.txt
```

All the files show synced up

```
$ docker run -it -v myvol:/var ubuntu
```