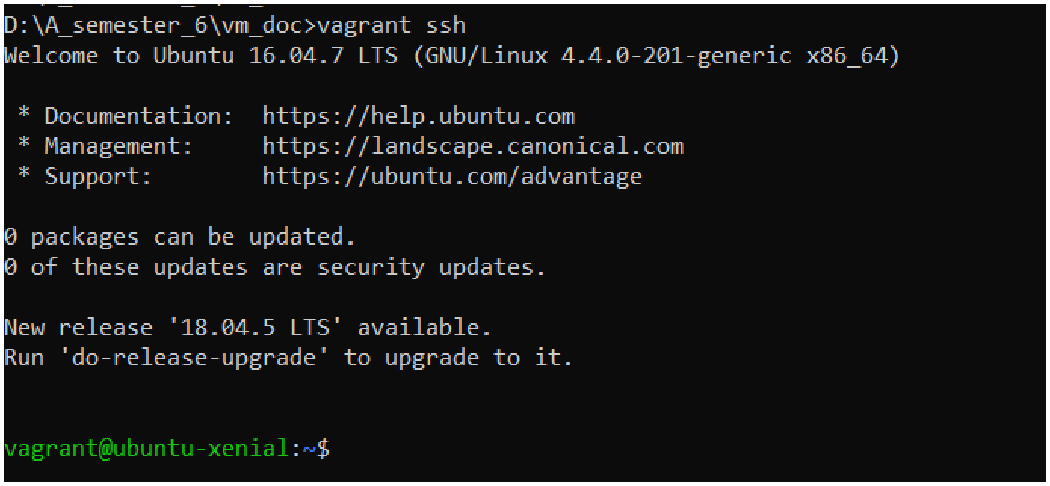
Application Containerization Lab File

Roll no: R171218009

Experiment 2

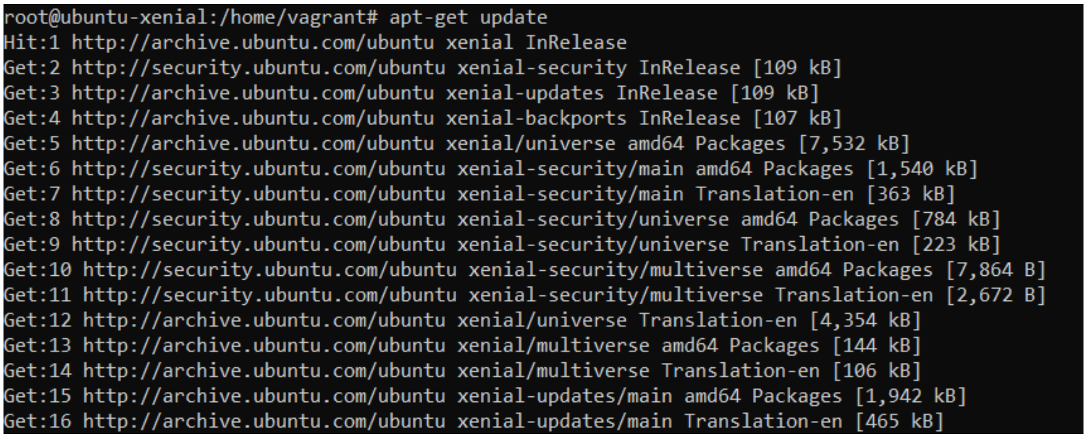
AIM: Install docker in VM, create docker volumes and share data between containers.

Run vagrant up and vagrant SSH commands.

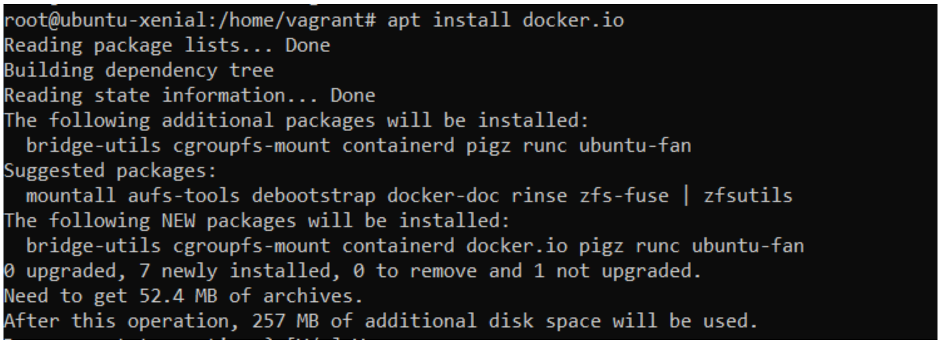


```
D:\A_semester_6\vm_doc>vagrant ssh
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-201-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 of these updates are security updates.

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


vagrant@ubuntu-xenial:~$
```

Installing docker

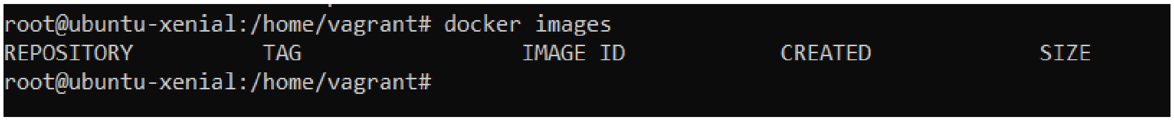


```
root@ubuntu-xenial:/home/vagrant# apt-get update
Hit:1 http://archive.ubuntu.com/ubuntu xenial InRelease
Get:2 http://security.ubuntu.com/ubuntu xenial-security InRelease [109 kB]
Get:3 http://archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]
Get:4 http://archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]
Get:5 http://archive.ubuntu.com/ubuntu xenial/universe amd64 Packages [7,532 kB]
Get:6 http://security.ubuntu.com/ubuntu xenial-security/main amd64 Packages [1,540 kB]
Get:7 http://security.ubuntu.com/ubuntu xenial-security/main Translation-en [363 kB]
Get:8 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 Packages [784 kB]
Get:9 http://security.ubuntu.com/ubuntu xenial-security/universe Translation-en [223 kB]
Get:10 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 Packages [7,864 B]
Get:11 http://security.ubuntu.com/ubuntu xenial-security/multiverse Translation-en [2,672 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/main amd64 Packages [1,942 kB]
Get:16 http://archive.ubuntu.com/ubuntu xenial-updates/main Translation-en [465 kB]
```



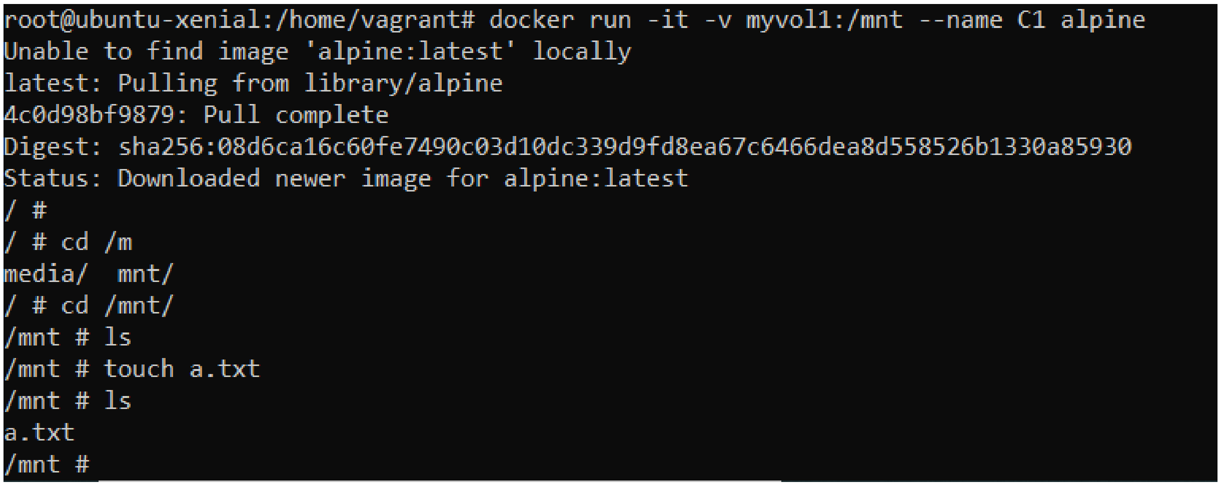
```
root@ubuntu-xenial:/home/vagrant# apt install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  bridge-utils cgroupfs-mount containerd pigz runc ubuntu-fan
Suggested packages:
  mountall aufs-tools debootstrap docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils cgroupfs-mount containerd docker.io pigz runc ubuntu-fan
0 upgraded, 7 newly installed, 0 to remove and 1 not upgraded.
Need to get 52.4 MB of archives.
After this operation, 257 MB of additional disk space will be used.
```

Run “docker images” command



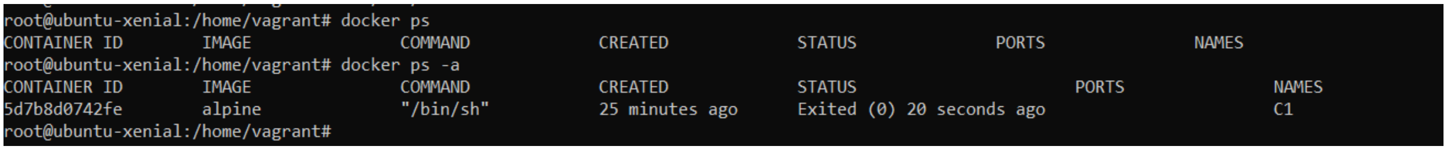
```
root@ubuntu-xenial:/home/vagrant# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
root@ubuntu-xenial:/home/vagrant#
```

Launch docker container c1 in interactive terminal mode, attach volume ‘myvol1’ and save some data in this volume.



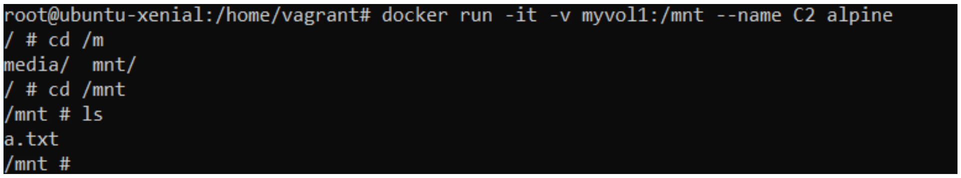
```
root@ubuntu-xenial:/home/vagrant# docker run -it -v myvol1:/mnt --name C1 alpine
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
4c0d98bf9879: Pull complete
Digest: sha256:08d6ca16c60fe7490c03d10dc339d9fd8ea67c6466dea8d558526b1330a85930
Status: Downloaded newer image for alpine:latest
/ #
/ # cd /m
media/  mnt/
/ # cd /mnt/
/mnt # ls
/mnt # touch a.txt
/mnt # ls
a.txt
/mnt #
```

Run “docker ps” command



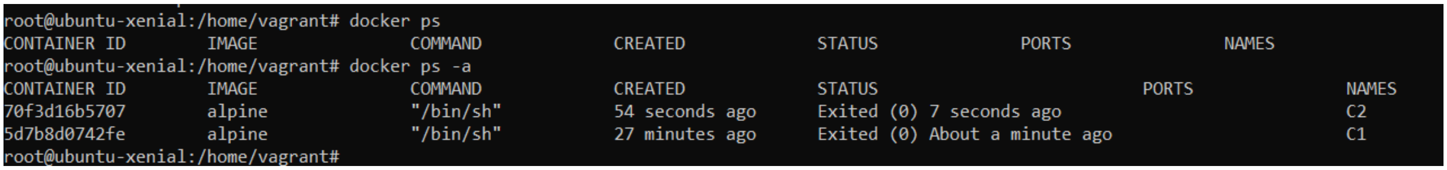
```
root@ubuntu-xenial:/home/vagrant# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
root@ubuntu-xenial:/home/vagrant# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS                      PORTS               NAMES
5d7b8d0742fe        alpine              "/bin/sh"           25 minutes ago      Exited (0) 20 seconds ago                       C1
root@ubuntu-xenial:/home/vagrant#
```

Launch another docker container c2 in interactive terminal mode, attach volume ‘myvol1’, the data saved in myvol1 by container c1 is visible in this container c2 as well.



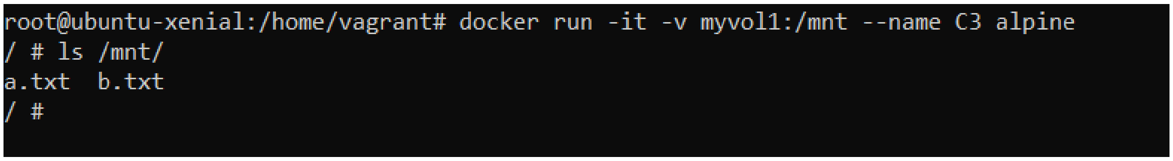
```
root@ubuntu-xenial:/home/vagrant# docker run -it -v myvol1:/mnt --name C2 alpine
/ # cd /m
media/  mnt/
/ # cd /mnt
/mnt # ls
a.txt
/mnt #
```

Run “docker ps” command

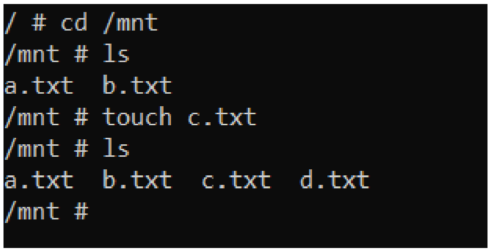


```
root@ubuntu-xenial:/home/vagrant# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
root@ubuntu-xenial:/home/vagrant# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS                        PORTS               NAMES
70f3d16b5707        alpine              "/bin/sh"           54 seconds ago      Exited (0) 7 seconds ago                          C2
5d7b8d0742fe        alpine              "/bin/sh"           27 minutes ago      Exited (0) About a minute ago                     C1
root@ubuntu-xenial:/home/vagrant#
```

In cmd window 1

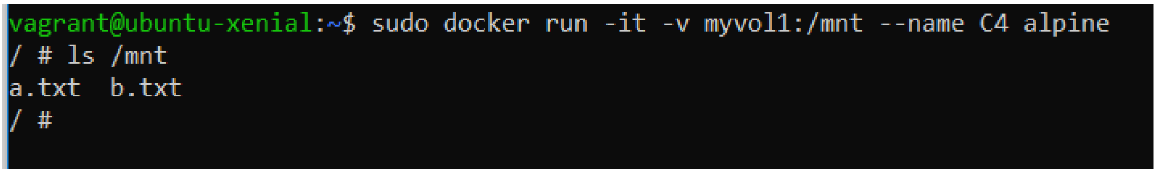


```
root@ubuntu-xenial:/home/vagrant# docker run -it -v myvol1:/mnt --name C3 alpine
/ # ls /mnt/
a.txt  b.txt
/ #
```

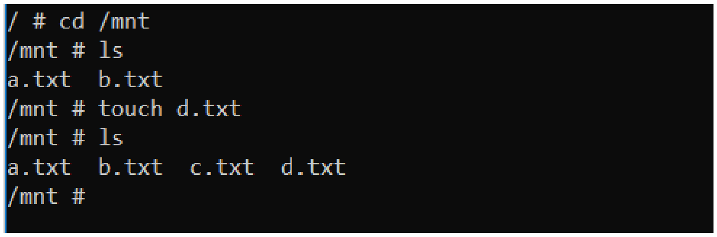


```
/ # cd /mnt
/mnt # ls
a.txt  b.txt
/mnt # touch c.txt
/mnt # ls
a.txt  b.txt  c.txt  d.txt
/mnt #
```

In cmd window 2

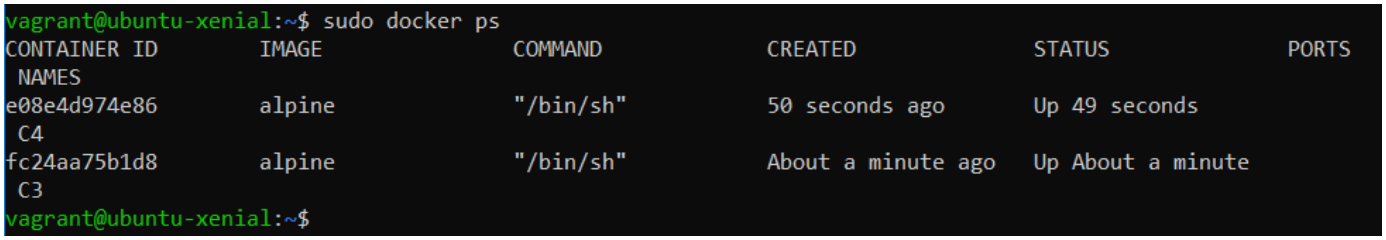


```
vagrant@ubuntu-xenial:~$ sudo docker run -it -v myvol1:/mnt --name C4 alpine
/ # ls /mnt
a.txt  b.txt
/ #
```



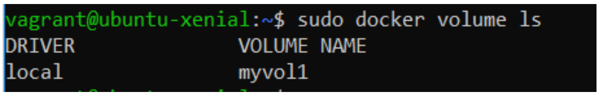
```
/ # cd /mnt
/mnt # ls
a.txt  b.txt
/mnt # touch d.txt
/mnt # ls
a.txt  b.txt  c.txt  d.txt
/mnt #
```

In cmd window 3

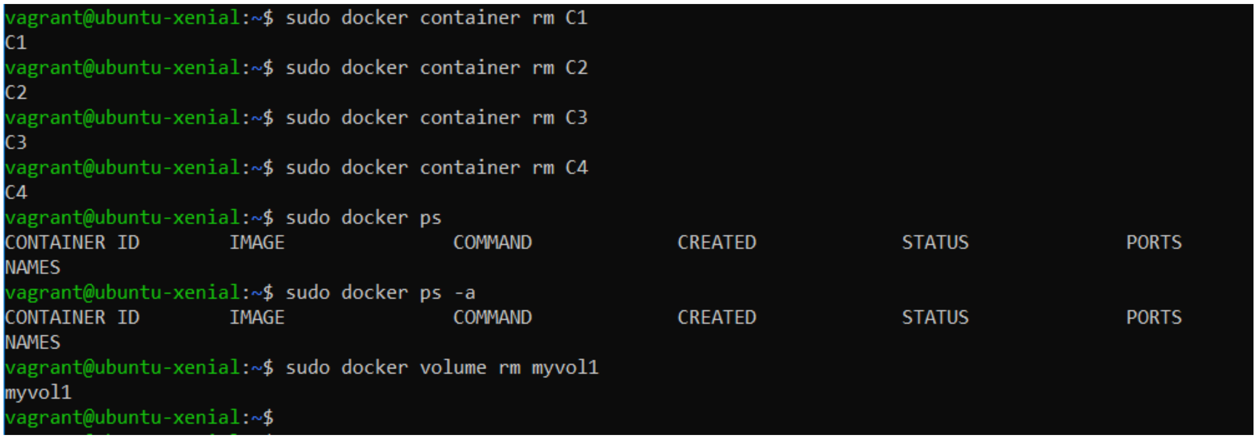


```
vagrant@ubuntu-xenial:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
 NAMES
e08e4d974e86        alpine              "/bin/sh"           50 seconds ago      Up 49 seconds
 C4
fc24aa75b1d8        alpine              "/bin/sh"           About a minute ago  Up About a minute
 C3
vagrant@ubuntu-xenial:~$
```

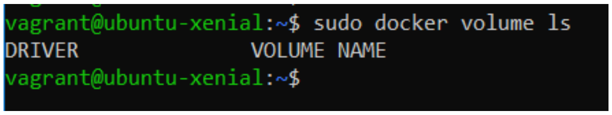
Removing volume “myvol1” (before that delete all containers using this volume)



```
vagrant@ubuntu-xenial:~$ sudo docker volume ls
DRIVER              VOLUME NAME
local               myvol1
```



```
vagrant@ubuntu-xenial:~$ sudo docker container rm C1
C1
vagrant@ubuntu-xenial:~$ sudo docker container rm C2
C2
vagrant@ubuntu-xenial:~$ sudo docker container rm C3
C3
vagrant@ubuntu-xenial:~$ sudo docker container rm C4
C4
vagrant@ubuntu-xenial:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
NAMES
vagrant@ubuntu-xenial:~$ sudo docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
NAMES
vagrant@ubuntu-xenial:~$ sudo docker volume rm myvol1
myvol1
vagrant@ubuntu-xenial:~$
```



```
vagrant@ubuntu-xenial:~$ sudo docker volume ls
DRIVER              VOLUME NAME
vagrant@ubuntu-xenial:~$
```